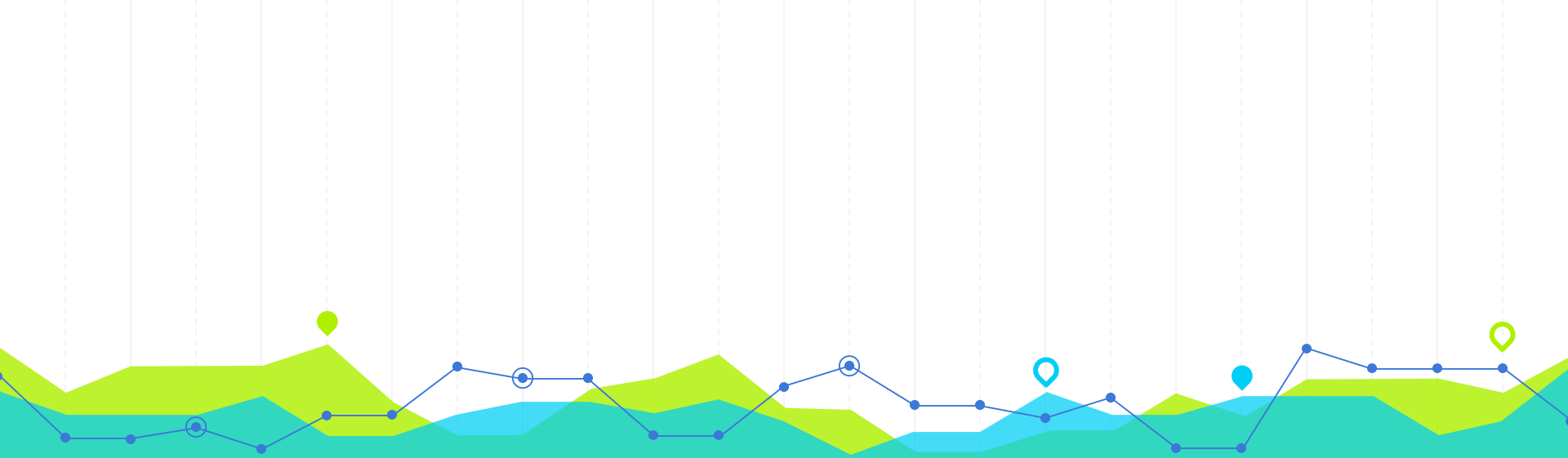


Which questions should we spend more time on? Any questions from past tutorials please feel free to clarify =)



# EE2211 Introduction to Machine Learning

T14 & T22, Chua Dingjuan [elechua@nus.edu.sg](mailto:elechua@nus.edu.sg)  
Slides @ [tiny.cc/ee2211tut](https://tiny.cc/ee2211tut)

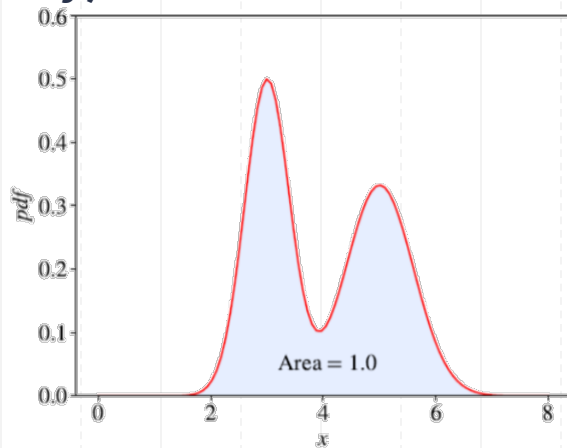
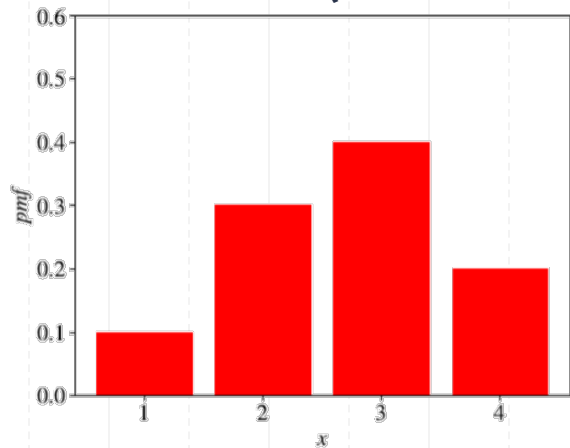


# Tutorial 3

Let's start with a brief summary

# Brief Summary of Key Points

## PMF vs PDF (mass vs density)



## Python

### Dictionary

Scipy  
`stats.norm.cdf`

## Bayes' Rule

$$\Pr(y|x) = \frac{\Pr(y) \Pr(x|y)}{\Pr(x)} = \frac{\Pr(y) \Pr(x|y)}{\sum_y \Pr(y) \Pr(x|y)}$$



# Discussion of Solutions

# 3

Q1, 2, 3, 10, **4, 5**, 6(self reading!), 7-13

# Question1

The random variable  $N$  has probability mass function (PMF)

$$P_N(n) = \begin{cases} c(1/2)^n, & n = 0, 1, 2 \\ 0, & \text{otherwise} \end{cases}$$

(a) What is the value of the constant  $c$ ?

(b) What is  $\Pr[N \leq 1]$ ?

$$P_N(0) = c, P_N(1) = \frac{1}{2}c, P_N(2) = \frac{1}{4}c$$



## Question1

The random variable  $N$  has probability mass function (PMF)

$$P_N(n) = \begin{cases} c(1/2)^n, & n = 0, 1, 2 \\ 0, & \text{otherwise} \end{cases}$$

(a) What is the value of the constant  $c$ ?

(b) What is  $\Pr[N \leq 1]$ ?

$$P_N(0) = c, P_N(1) = \frac{1}{2}c, P_N(2) = \frac{1}{4}c$$

## Solution

(a) Sum of all probabilities = 1 (PMF sum up to one)

$$\sum_{n=0}^2 P_X(n) = c + \frac{c}{2} + \frac{c}{4} = 1 \quad \text{implying } c = \frac{4}{7}$$

$$(b) \Pr[N \leq 1] = P_N(n = 0) + P_N(n = 1) = c + \frac{c}{2} = \frac{6}{7}$$

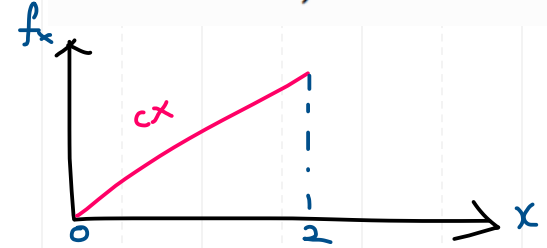
## Question2

The random variable  $X$  has probability density function (PDF)

Use the PDF to find

- (a) the constant  $c$ ,
- (b)  $\Pr[0 \leq X \leq 1]$ ,
- (c)  $\Pr[-1/2 \leq X \leq 1/2]$ .

$$f_X(x) = \begin{cases} cx, & 0 \leq x \leq 2 \\ 0, & \text{otherwise.} \end{cases}$$



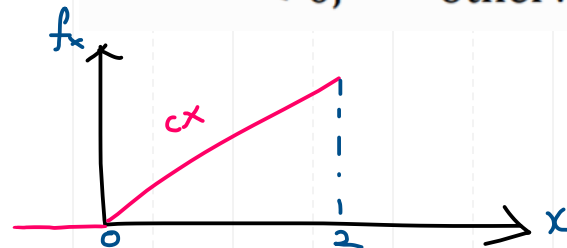
## Question2

The random variable  $X$  has probability density function (PDF)

Use the PDF to find

- (a) the constant  $c$ ,
- (b)  $\Pr[0 \leq X \leq 1]$ ,
- (c)  $\Pr[-1/2 \leq X \leq 1/2]$ .

$$f_X(x) = \begin{cases} cx, & 0 \leq x \leq 2 \\ 0, & \text{otherwise.} \end{cases}$$



## Solution

- (a) Sum of all probabilities = 1 (PDF area under the curve sum up to one)

$$\int_0^2 cx \, dx = c \left[ \frac{x^2}{2} \right]_0^2 = c \left[ \frac{4}{2} \right] = 2c = 1$$

$$c = 1/2.$$



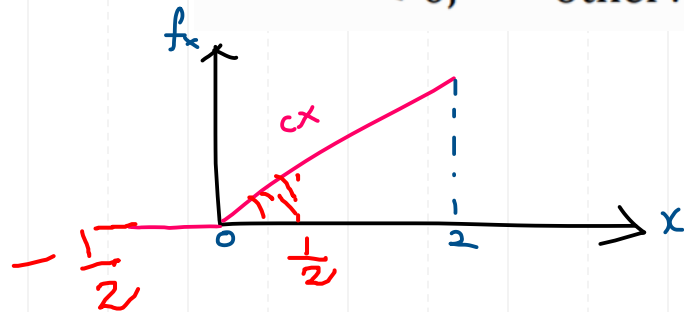
## Question2

The random variable  $X$  has probability density function (PDF)

$$f_X(x) = \begin{cases} cx, & 0 \leq x \leq 2 \\ 0, & \text{otherwise.} \end{cases}$$

Use the PDF to find

- (a) the constant  $c$ ,
- (b)  $\Pr[0 \leq X \leq 1]$ ,
- (c)  $\Pr[-1/2 \leq X \leq 1/2]$ .



## Solution

- (b)  $\Pr[0 \leq X \leq 1]$

$$\Pr[0 \leq X \leq 1] = \int_0^1 x/2 \, dx = 1/4$$

- (c)  $\Pr[-1/2 \leq X \leq 1/2]$

$$\Pr[-1/2 \leq X \leq 1/2] = \int_{-1/2}^{1/2} x/2 \, dx = 1/16$$

## Question3

Let  $A = \{\text{resistor is within } 50\Omega \text{ of the nominal value}\}$ .

The probability that a resistor is from machine B is  $\Pr[B] = 0.3$ . The probability that a resistor is acceptable, i.e., within  $50\Omega$  of the nominal value, is  $\Pr[A] = 0.78$ . Given that a resistor is from machine B, the conditional probability that it is acceptable is  $\Pr[A|B] = 0.6$ .

What is the probability that an acceptable resistor comes from machine B?

## Solution

$\Pr[A]$  = resistor is  $\pm 50\Omega$  of nominal value = 0.78

$\Pr[B]$  = resistor from machine B = 0.3

$\Pr[A|B]$  = P (resistor is  $\pm 50\Omega$  of nominal value, given that it comes from B)  
= 0.6

$\Pr[B|A]$  = P (resistor is from B, given that it is  $\pm 50\Omega$  of nominal value) = ?

acceptable

$$\begin{aligned}\Pr(B|A) &= \frac{\Pr(A|B)\Pr(B)}{\Pr(A)} \\ &= (0.6)(0.3)/(0.78) \approx 0.23\end{aligned}$$

## Question4

Consider tossing a fair six-sided die. There are only six outcomes possible,  $\Omega = \{1, 2, 3, 4, 5, 6\}$ . Suppose we toss two dice and assume that each throw is **independent**. *outcome of one does not influence the other*

(a) What is the probability that the sum of the dice equals seven?

i. List out all pairs of possible outcomes together with their sums from the two throws.  
(hint: enumerate all the items in `range(1,7)`)

ii. Collect all of the (a, b) pairs that sum to each of the possible values from two to twelve (including the sum equals seven).

(hint: use dictionary from `collections` import `defaultdict` to collect all of the (a, b) pairs that sum to each of the possible values from two to twelve)

(b) What is the probability that half the product of three dice will exceed their sum?



# Question4 Solution

(a) What is the probability that the sum of the dice equals seven?

## Python Dictionary CS1010E

- You search for the **key** in the dictionary
- Then look for its **value**



- Each key has a **correspondent** value

```
>>> students = {'A100000X': 'John', 'A123456X': 'Peter',  
                'A999999X': 'Paul'}  
>>> students['A123456X']  
'Peter'
```

key : value pair

Tuples use '(' and ')'  
Lists use '[' and ']'  
Dictionary use '{' and '}'

## Question4 Solution

- (a) What is the probability that the sum of the dice equals seven?
- List out all pairs of possible outcomes together with their sums from the two throws.
  - Collect all of the (a, b) pairs that sum to each of the possible values from two to twelve (including the sum equals seven).

```
#(i)
d={(i,j):i+j for i in range(1,7) for j in range(1,7)}

#(ii) collect all of the (a, b) pairs that sum to
#each of the possible values from two to twelve
|
from collections import defaultdict
dinv = defaultdict(list)

for i,j in d.items(): dinv[j].append(i)

#      Compute the probability measured for each sum
#      including the sum equals seven

X={i:len(j)/36. for i,j in dinv.items() }

print(X)
```

## Question4 Solution

(a) What is the probability that the sum of the dice equals seven?

- List out all pairs of possible outcomes together with their sums from the two throws.
- Collect all of the (a, b) pairs that sum to each of the possible values from two to twelve (including the sum equals seven).

```
#(i) tuple  
d={(i,j):i+j for i in range(1,7) for j in range(1,7)}
```

```
>>> print([i for i in range(1,7) for j in range(1,7)])  
[1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 5, 5, 5,  
, 5, 5, 5, 6, 6, 6, 6, 6, 6]  
>>> print([j for i in range(1,7) for j in range(1,7)])  
[1, 2, 3, 4, 5, 6, 1, 2, 3, 4, 5, 6, 1, 2, 3, 4, 5, 6, 1, 2, 3, 4, 5, 6, 1, 2, 3,  
, 4, 5, 6, 1, 2, 3, 4, 5, 6]
```

```
>>> d={(i,j):i+j for i in range(1,7) for j in range(1,7)}  
  
>>> d  
{(1, 1): 2, (1, 2): 3, (1, 3): 4, (1, 4): 5, (1, 5): 6, (1, 6): 7, (2, 1): 3, (2,  
2): 4, (2, 3): 5, (2, 4): 6, (2, 5): 7, (2, 6): 8, (3, 1): 4, (3, 2): 5, (3, 3):  
6, (3, 4): 7, (3, 5): 8, (3, 6): 9, (4, 1): 5, (4, 2): 6, (4, 3): 7, (4, 4): 8, (4,  
5): 9, (4, 6): 10, (5, 1): 6, (5, 2): 7, (5, 3): 8, (5, 4): 9, (5, 5): 10, (5,  
6): 11, (6, 1): 7, (6, 2): 8, (6, 3): 9, (6, 4): 10, (6, 5): 11, (6, 6): 12}
```

## Question4 Solution

ii. Collect all of the (a, b) pairs that sum to each of the possible values from two to twelve (including the sum equals seven).

```
#(i)
d={(i,j):i+j for i in range(1,7) for j in range(1,7)}

#(ii) collect all of the (a, b) pairs that sum to
#each of the possible values from two to twelve

from collections import defaultdict
dinv = defaultdict(list)

for i,j in d.items(): dinv[j].append(i)

#      Compute the probability measured for each sum
#      including the sum equals seven

X={i:len(j)/36. for i,j in dinv.items() }

print(X)
```

The main difference between defaultdict and dict is that when you try to access or modify a key that's not present in the dictionary, a default value is automatically given to that key.

```
>>> from collections import defaultdict
>>> dinv = defaultdict(list)
>>> dinv
defaultdict(<class 'list'>, {})
```

Call list() to create a new empty list  
dinv ['key'] . append( value )

*dinv['key'].append(value)*  
*(1,1)*  
*(1,2)*



## Question4 Solution

ii. Collect all of the (a, b) pairs that sum to each of the possible values from two to twelve (including the sum equals seven).

```
#(i)
d={(i,j):i+j for i in range(1,7) for j in range(1,7)}
```

```
#(ii) collect all of the (a, b) pairs that sum to
#each of the possible values from two to twelve
```

```
from collections import defaultdict
dinv = defaultdict(list)
```

```
for i,j in d.items(): dinv[j].append(i)
# Compute the probability measured for each sum
# including the sum equals seven
```

```
defaultdict(<class 'list'>, {2: [(1, 1)], 3: [(1, 2), (2, 1)], 4: [(1, 3), (2, 2), (3, 1)], 5: [(1, 4), (2, 3), (3, 2), (4, 1)], 6: [(1, 5), (2, 4), (3, 3), (4, 2), (5, 1)], 7: [(1, 6), (2, 5), (3, 4), (4, 3), (5, 2), (6, 1)], 8: [(2, 6), (3, 5), (4, 4), (5, 3), (6, 2)], 9: [(3, 6), (4, 5), (5, 4), (6, 3)], 10: [(4, 6), (5, 5), (6, 4)], 11: [(5, 6), (6, 5)], 12: [(6, 6)]})
```

```
>>> d.items()
dict_items([(1, 1), 2), ((1, 2), 3), ((1, 3), 4), ((1, 4), 5), ((1, 5), 6), ((1, 6), 7), ((2, 1), 3), ((2, 2), 4), ((2, 3), 5), ((2, 4), 6), ((2, 5), 7), ((2, 6), 8), ((3, 1), 4), ((3, 2), 5), ((3, 3), 6), ((3, 4), 7), ((3, 5), 8), ((3, 6), 9), ((4, 1), 5), ((4, 2), 6), ((4, 3), 7), ((4, 4), 8), ((4, 5), 9), ((4, 6), 10), ((5, 1), 6), ((5, 2), 7), ((5, 3), 8), ((5, 4), 9), ((5, 5), 10), ((5, 6), 11), ((6, 1), 7), ((6, 2), 8), ((6, 3), 9), ((6, 4), 10), ((6, 5), 11), ((6, 6), 12)])
```

```
>>> for i,j in d.items(): print (i)
```

```
(1, 1)
(1, 2)
(1, 3)
(1, 4)
(1, 5)
(1, 6)
(2, 1)
```

```
>>> for i,j in d.items(): print (j)
```

```
2
3
4
5
6
7
3
```



## Question4 Solution

ii. Collect all of the (a, b) pairs that sum to each of the possible values from two to twelve (including the sum equals seven).

```
#(i)
d={(i,j):i+j for i in range(1,7) for j in range(1,7)}

#(ii) collect all of the (a, b) pairs that sum to
#each of the possible values from two to twelve

from collections import defaultdict
dinv = defaultdict(list)

for i,j in d.items():
    di = i+j
    dinv[di].append((i,j))

# Compute the probability of each sum
# including the sum equals seven

X={i:len(dinv[i])/36. for i,j in dinv.items() }

print(X)
```

*sum*

*dice pair values*

*1*

```
{2: 0.027777777777777776, 3: 0.05555555555555555, 4: 0.08333333333333333, 5: 0.11111111111111111, 6: 0.13888888888888889, 7: 0.16666666666666666, 8: 0.13888888888888889, 9: 0.11111111111111111, 10: 0.08333333333333333, 11: 0.05555555555555555, 12: 0.027777777777777776}
```

## Question4

Consider tossing a fair six-sided die. There are only six outcomes possible,  $\Omega = \{1, 2, 3, 4, 5, 6\}$ . Suppose we toss two dice and assume that each throw is **independent**.

(b) What is the probability that half the product of three dice will exceed their sum?

```
# (b) What is the probability that half the product of three dice will exceed their sum?
d={(i,j,k):((i*j*k)/2>i+j+k) for i in range(1,7)
                                for j in range(1,7)
                                for k in range(1,7)}

dinv = defaultdict(list)
for i,j in d.items(): dinv[j].append(i)
X={i:len(j)/6.0**3 for i,j in dinv.items() }
print(X)
```

## Question4

Consider tossing a fair six-sided die. There are only six outcomes possible,  $\Omega = \{1, 2, 3, 4, 5, 6\}$ . Suppose we toss two dice and assume that each throw is **independent**.

(b) What is the probability that half the product of three dice will exceed their sum?

```
# (b) What is the probability that half the product of three dice will exceed their sum?  
d={(i,j,k):((i*j*k)/2>i+j+k) for i in range(1,7)  
    for j in range(1,7)  
    for k in range(1,7)}
```

```
>>> d={(i,j,k):((i*j*k)/2>i+j+k) for i in range(1,7)  
    for j in range(1,7)  
    for k in range(1,7)}
```

```
>>> d  
{(1, 1, 1): False, (1, 1, 2): False, (1, 1, 3): False, (1, 1, 4): False, (1, 1, 5): False, (1, 1, 6): False, (1, 2, 1): False, (1, 2, 2): False, (1, 2, 3): False, (1, 2, 4): False, (1, 2, 5): False, (1, 2, 6): False, (1, 3, 1): False, (1, 3, 2): False, (1, 3, 3): False, (1, 3, 4): False, (1, 3, 5): False, (1, 3, 6): False, (1, 4, 1): False, (1, 4, 2): False, (1, 4, 3): False, (1, 4, 4): False, (1, 4, 5): False, (1, 4, 6): True, (1, 5, 1): False, (1, 5, 2): False, (1, 5, 3): False, (1, 5, 4): False, (1, 5, 5): True, (1, 5, 6): True, (1, 6, 1): False, (1, 6, 2): False, (1, 6, 3): False, (1, 6, 4): False, (1, 6, 5): True, (1, 6, 6): True, (2, 1, 1): False, (2, 1, 2): False, (2, 1, 3): False, (2, 1, 4): False, (2, 1, 5): False, (2, 1, 6): False, (2, 2, 1): False, (2, 2, 2): False, (2, 2, 3): False, (2, 2, 4): False, (2, 2, 5): False, (2, 2, 6): False, (2, 3, 1): False, (2, 3, 2): False, (2, 3, 3): False, (2, 3, 4): False, (2, 3, 5): False, (2, 3, 6): False, (2, 4, 1): False, (2, 4, 2): False, (2, 4, 3): False, (2, 4, 4): False, (2, 4, 5): False, (2, 4, 6): False, (2, 5, 1): False, (2, 5, 2): False, (2, 5, 3): False, (2, 5, 4): False, (2, 5, 5): False, (2, 5, 6): False, (2, 6, 1): False, (2, 6, 2): False, (2, 6, 3): False, (2, 6, 4): False, (2, 6, 5): False, (2, 6, 6): False, (3, 1, 1): False, (3, 1, 2): False, (3, 1, 3): False, (3, 1, 4): False, (3, 1, 5): False, (3, 1, 6): False, (3, 2, 1): False, (3, 2, 2): False, (3, 2, 3): False, (3, 2, 4): False, (3, 2, 5): False, (3, 2, 6): False, (3, 3, 1): False, (3, 3, 2): False, (3, 3, 3): False, (3, 3, 4): False, (3, 3, 5): False, (3, 3, 6): False, (3, 4, 1): False, (3, 4, 2): False, (3, 4, 3): False, (3, 4, 4): False, (3, 4, 5): False, (3, 4, 6): False, (3, 5, 1): False, (3, 5, 2): False, (3, 5, 3): False, (3, 5, 4): False, (3, 5, 5): False, (3, 5, 6): False, (3, 6, 1): False, (3, 6, 2): False, (3, 6, 3): False, (3, 6, 4): False, (3, 6, 5): False, (3, 6, 6): False, (4, 1, 1): False, (4, 1, 2): False, (4, 1, 3): False, (4, 1, 4): False, (4, 1, 5): False, (4, 1, 6): False, (4, 2, 1): False, (4, 2, 2): False, (4, 2, 3): False, (4, 2, 4): False, (4, 2, 5): False, (4, 2, 6): False, (4, 3, 1): False, (4, 3, 2): False, (4, 3, 3): False, (4, 3, 4): False, (4, 3, 5): False, (4, 3, 6): False, (4, 4, 1): False, (4, 4, 2): False, (4, 4, 3): False, (4, 4, 4): False, (4, 4, 5): False, (4, 4, 6): False, (4, 5, 1): False, (4, 5, 2): False, (4, 5, 3): False, (4, 5, 4): False, (4, 5, 5): False, (4, 5, 6): False, (4, 6, 1): False, (4, 6, 2): False, (4, 6, 3): False, (4, 6, 4): False, (4, 6, 5): False, (4, 6, 6): False, (5, 1, 1): False, (5, 1, 2): False, (5, 1, 3): False, (5, 1, 4): False, (5, 1, 5): False, (5, 1, 6): False, (5, 2, 1): False, (5, 2, 2): False, (5, 2, 3): False, (5, 2, 4): False, (5, 2, 5): False, (5, 2, 6): False, (5, 3, 1): False, (5, 3, 2): False, (5, 3, 3): False, (5, 3, 4): False, (5, 3, 5): False, (5, 3, 6): False, (5, 4, 1): False, (5, 4, 2): False, (5, 4, 3): False, (5, 4, 4): False, (5, 4, 5): False, (5, 4, 6): False, (5, 5, 1): False, (5, 5, 2): False, (5, 5, 3): False, (5, 5, 4): False, (5, 5, 5): False, (5, 5, 6): False, (5, 6, 1): False, (5, 6, 2): False, (5, 6, 3): False, (5, 6, 4): False, (5, 6, 5): False, (5, 6, 6): False, (6, 1, 1): False, (6, 1, 2): False, (6, 1, 3): False, (6, 1, 4): False, (6, 1, 5): False, (6, 1, 6): False, (6, 2, 1): False, (6, 2, 2): False, (6, 2, 3): False, (6, 2, 4): False, (6, 2, 5): False, (6, 2, 6): False, (6, 3, 1): False, (6, 3, 2): False, (6, 3, 3): False, (6, 3, 4): False, (6, 3, 5): False, (6, 3, 6): False, (6, 4, 1): False, (6, 4, 2): False, (6, 4, 3): False, (6, 4, 4): False, (6, 4, 5): False, (6, 4, 6): False, (6, 5, 1): False, (6, 5, 2): False, (6, 5, 3): False, (6, 5, 4): False, (6, 5, 5): False, (6, 5, 6): False, (6, 6, 1): False, (6, 6, 2): False, (6, 6, 3): False, (6, 6, 4): False, (6, 6, 5): False, (6, 6, 6): False}
```

## Question4

Consider tossing a fair six-sided die. There are only six outcomes possible,  $\Omega = \{1, 2, 3, 4, 5, 6\}$ . Suppose we toss two dice and assume that each throw is **independent**.

(b) What is the probability that half the product of three dice will exceed their sum?

```
# (b) What is the probability that half the product of three dice will exceed their sum?
d={(i,j,k):((i*j*k)/2>i+j+k) for i in range(1,7)
                                for j in range(1,7)
                                for k in range(1,7)}
```

```
dinv = defaultdict(list)
for i,j in d.items(): dinv[j].append(i)
```

```
>>> dinv = defaultdict(list)
>>> for i,j in d.items(): dinv[j].append(i)
```

```
>>> dinv
defaultdict(<class 'list'>, {False: [(1, 1, 1), (1, 1, 2), (1, 1, 3), (1, 1, 4), (1, 1, 5), (1, 1, 6), (1, 2, 1), (1, 2, 2), (1, 2, 3), (1, 2, 4), (1, 2, 5), (1, 2, 6), (1, 3, 1), (1, 3, 2), (1, 3, 3), (1, 3, 4), (1, 3, 5), (1, 3, 6), (1, 4, 1), (1, 4, 2), (1, 4, 3), (1, 4, 4), (1, 4, 5), (1, 4, 6), (1, 5, 1), (1, 5, 2), (1, 5, 3), (1, 5, 4), (1, 5, 5), (1, 5, 6), (1, 6, 1), (1, 6, 2), (1, 6, 3), (1, 6, 4), (1, 6, 5), (1, 6, 6), (2, 1, 1), (2, 1, 2), (2, 1, 3), (2, 1, 4), (2, 1, 5), (2, 1, 6), (2, 2, 1), (2, 2, 2), (2, 2, 3), (2, 2, 4), (2, 2, 5), (2, 2, 6), (2, 3, 1), (2, 3, 2), (2, 3, 3), (2, 3, 4), (2, 3, 5), (2, 3, 6), (2, 4, 1), (2, 4, 2), (2, 4, 3), (2, 4, 4), (2, 4, 5), (2, 4, 6), (2, 5, 1), (2, 5, 2), (2, 5, 3), (2, 5, 4), (2, 5, 5), (2, 5, 6), (2, 6, 1), (2, 6, 2), (2, 6, 3), (2, 6, 4), (2, 6, 5), (2, 6, 6), (3, 1, 1), (3, 1, 2), (3, 1, 3), (3, 1, 4), (3, 1, 5), (3, 1, 6), (3, 2, 1), (3, 2, 2), (3, 2, 3), (3, 2, 4), (3, 2, 5), (3, 2, 6), (3, 3, 1), (3, 3, 2), (3, 3, 3), (3, 3, 4), (3, 3, 5), (3, 3, 6), (3, 4, 1), (3, 4, 2), (3, 4, 3), (3, 4, 4), (3, 4, 5), (3, 4, 6), (3, 5, 1), (3, 5, 2), (3, 5, 3), (3, 5, 4), (3, 5, 5), (3, 5, 6), (3, 6, 1), (3, 6, 2), (3, 6, 3), (3, 6, 4), (3, 6, 5), (3, 6, 6)], True: [(1, 4, 6), (1, 5, 5), (1, 5, 6), (1, 6, 4), (1, 6, 5), (1, 6, 6), (2, 2, 5), (2, 2, 6), (2, 3, 3), (2, 3, 4), (2, 3, 5), (2, 3, 6), (2, 4, 3), (2, 4, 4), (2, 4, 5), (2, 4, 6), (2, 5, 2), (2, 5, 3), (2, 5, 4), (2, 5, 5), (2, 5, 6), (2, 6, 2), (2, 6, 3), (2, 6, 4), (2, 6, 5), (2, 6, 6), (3, 2, 3), (3, 2, 4), (3, 2, 5), (3, 2, 6), (3, 3, 2), (3, 3, 3), (3, 3, 4), (3, 3, 5), (3, 3, 6), (3, 4, 2), (3, 4, 3), (3, 4, 4), (3, 4, 5), (3, 4, 6), (3, 5, 2), (3, 5, 3), (3, 5, 4), (3, 5, 5), (3, 5, 6), (3, 6, 2), (3, 6, 3), (3, 6, 4), (3, 6, 5), (3, 6, 6)]})
```

## Question4

Consider tossing a fair six-sided die. There are only six outcomes possible,  $\Omega = \{1, 2, 3, 4, 5, 6\}$ . Suppose we toss two dice and assume that each throw is **independent**.

(b) What is the probability that half the product of three dice will exceed their sum?

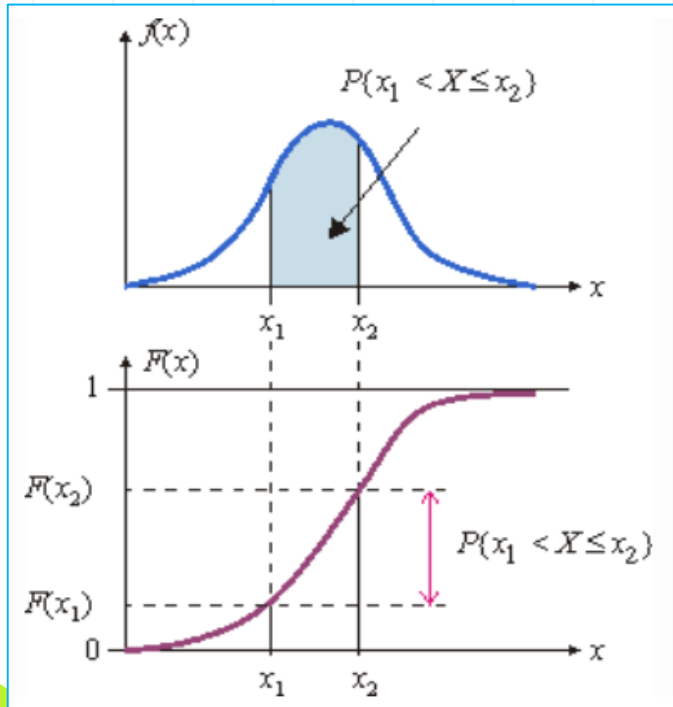
```
# (b) What is the probability that half the product of three dice will exceed their sum?
d={(i,j,k):((i*j*k)/2>i+j+k) for i in range(1,7)
                                for j in range(1,7)
                                for k in range(1,7)}

dinv = defaultdict(list)
for i,j in d.items(): dinv[j].append(i)
X={i:len(j)/6.0**3 for i,j in dinv.items() }
print(X)
```

```
>>> X
{False: 0.37037037037037035, True: 0.6296296296296297}
```

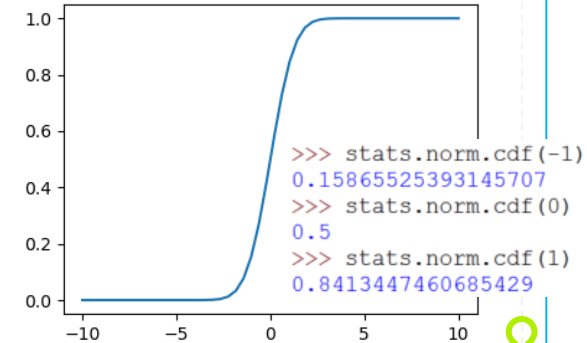
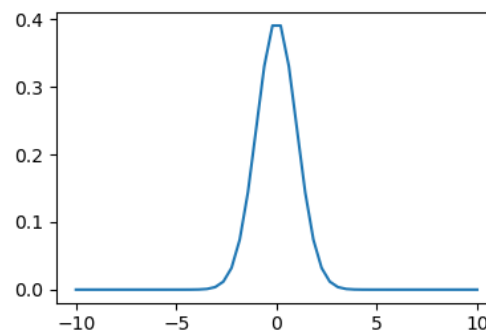
## Question5

Assuming a normal (Gaussian) distribution with mean  $30\ \Omega$  and standard deviation of  $1.8\ \Omega$ , determine the probability that a resistor coming off the production line will be within the range of  $28\ \Omega$  to  $33\ \Omega$ . (Hint: use `stats.norm.cdf` function from `scipy` import `stats`)



`stats.norm.cdf`:

- Creates a normal variable
- Returns the cumulative distribution function value
- `stats.norm.cdf(cdfvalue, mu, sigma)`





## Question5

Assuming a normal (Gaussian) distribution with mean  $30\ \Omega$  and standard deviation of  $1.8\ \Omega$ , determine the probability that a resistor coming off the production line will be within the range of  $28\ \Omega$  to  $33\ \Omega$ . (Hint: use `stats.norm.cdf` function from `scipy` import `stats`)

```
mu = 30 # mean = 30Ω
sigma = 1.8 # standard deviation = 1.8Ω
x1 = 28 # lower bound = 28Ω
x2 = 33 # upper bound = 33Ω

## calculate probabilities
# probability from Z=0 to lower bound
p_lower = stats.norm.cdf(x1, mu, sigma)
# probability from Z=0 to upper bound
p_upper = stats.norm.cdf(x2, mu, sigma)
# probability of the interval
Prob = (p_upper) - (p_lower)
```

```
>>> p_lower
0.13326026290250537
```

```
>>> p_upper
0.9522096477271853
```

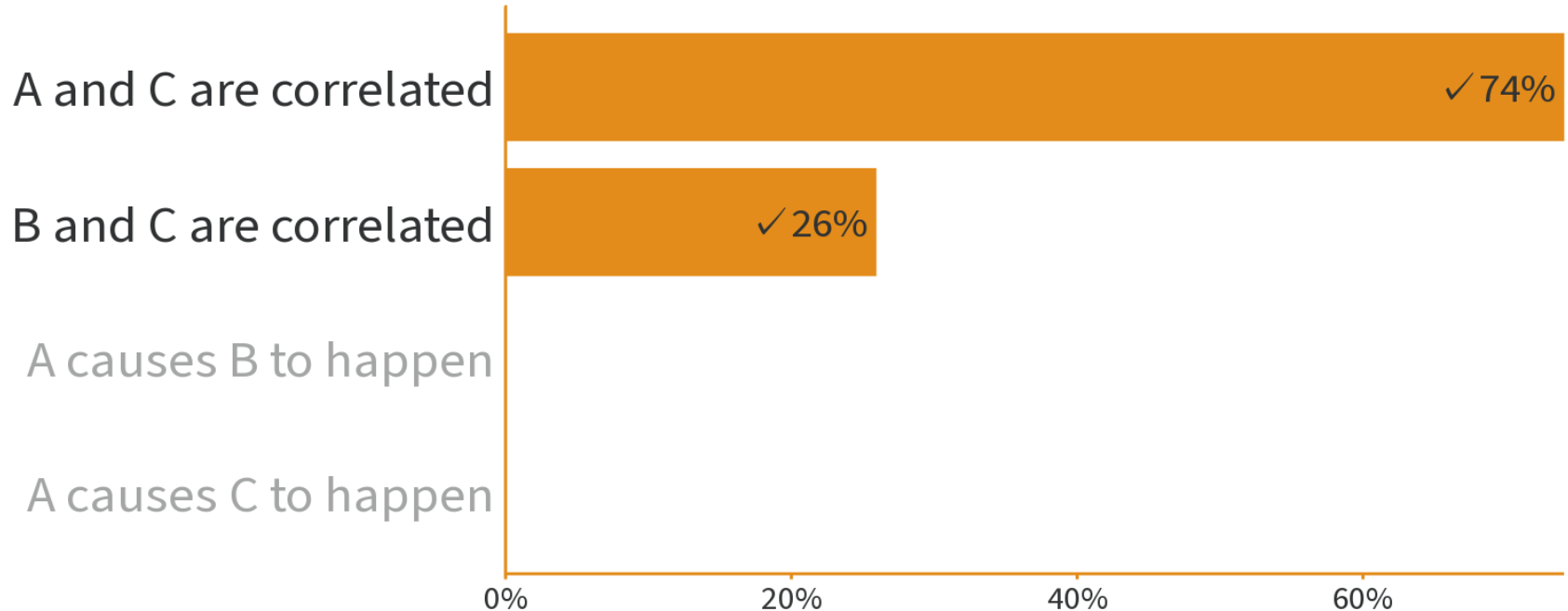
```
>>> Prob
0.8189493848246799
```

$P(R \leq 28)$

$P(R \leq 33)$

$P(28 \leq R \leq 33)$

## Q7 If A and B are correlated, but they're actually caused by C, which of the following statements are correct?





When poll is active, respond at [pollev.com/cdj](https://pollev.com/cdj)

Text **CDJ** to **+61 480 025 509** once to join



**Q9 We toss a coin and observe which side is facing up.  
Which of the following statements represent valid  
probability assignments for observing head  $P['H']$  and tail  
 $P['T']$ ?**

$$P['H']=0.2, P['T']=0.9$$

$$P['H']=0.0, P['T']=1.0$$

$$P['H']=-0.1, P['T']=1.1$$

$$P['H']=P['T']=0.5$$

### Question9

Are the two vectors  $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$  linearly dependent?

### Question10

The rank of the matrix  $\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$  is :

### Question11

The rank of the matrix  $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$  is :

## Question9

Are the two vectors  $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix}$  linearly dependent?

No. They are not multiples of each other.  $\begin{bmatrix} 4 \\ 5 \\ 6 \end{bmatrix} \neq n \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

## Question10

The rank of the matrix  $\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$  is :

Using row-echelon form :  $\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 3 \\ 0 & -2 \end{bmatrix}$

# of non-zero rows / columns = 2

row2 = row2 - 2\*row1

Therefore rank is 2.

## Question11

The rank of the matrix  $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$  is :

Using row-echelon form, rank is 2.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -12 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 0 \end{bmatrix}$$

row2 = row2 - 4\*row1

row3 = row3 - 7\*row1

row3 = row3 - 2\*row2