HW_1 ¶

Задача_1

Имеется колода в 52 карты. Найти число возможностей вытянуть из неё 4 карты так, чтобы среди них был хотя бы один туз.

Решение:

Разобъем колоду на две части: в одной тузы (4 карты), в остальной - оставштеся 48 карт.

1) Количество способов вытащить **ОДНОГО** туза => (C_4^1) и при этом вытащить ТРИ "не туза" => (C_{48}^3)
Таких комбинаций: $A_1 = C_4^1 \cdot C_{48}^3$.

2) Количество способов вытащить **ДВУХ** тузов => (C_4^2) и при этом вытащить ДВА "не туза" => (C_{48}^2)
Такое событие наступит с вероятносТаких комбинацийтью : $A_2 = C_4^2 \cdot C_{48}^2$.

3) Количество способов вытащить **ТРЁХ** тузов => (C_4^3) и при этом вытащить ОДНОГО "не туза" => (C_{48}^1)
Таких комбинаций : $A_3 = C_4^3 \cdot C_{48}^1$.

4) Количество способов вытащить **ЧЕТЫРЁХ** тузов => (C_4^4) Таких комбинаций : $A_4 = C_4^4 \cdot C_{48}^0$.

A1...A4 комбинации не повторяются, поэтому для итогового значения можем сложить $A_1 \ldots A_4$

Для того, чтобы найти искомое число, нужно сложить количество этих событий:
$A = A_1 + A_2 + A_3 + A_4$

Сочетания считаем по формуле: $C_n^k = \dfrac{n!}{k! \cdot (n-k)!}$

In [57]:

```
import numpy as np
import math
```

In [58]:

```
def get_combination(n, k):
    "сочетание"
    c = np.math.factorial(n) // (np.math.factorial(k) * np.math.factorial(n - k))
    return c
```

In [59]:

```
a1 = get_combination(4,1) * get_combination(48,3)
a2 = get_combination(4,2) * get_combination(48,2)
a3 = get_combination(4,3) * get_combination(48,1)
a4 = get_combination(4,4) * get_combination(48,0)
a = a1 + a2 + a3 + a4
print(f'ОТВЕТ: число возможностей вытянуть из 52х-карточной колоды 4 карты так, чтобы с
реди них был хотя бы один туз равно {a}')
```

```
ОТВЕТ: число возможностей вытянуть из 52х-карточной колоды 4 карты так, чт
обы среди них был хотя бы один туз равно 76145
```

Задача_2

Из 60 вопросов, входящих в экзаменационные билеты, студент знает 50. Случайным образом студент вытягивает 3 вопроса. Какова вероятность, что все выбранные вопросы знакомы студенту?

Решение:

Наше событие А - все выбранные вопросы знакомы студенту.
Вероятность найдем по формуле классической вероятности: $P(A) = \frac{m}{n}$

Где: n — общее число исходов (3 из 60) ($n = C_{60}^3$) , а m — число исходов, которые влекут за собой событие А (3 из 50) ($m = C_{50}^3$).

Сочетания считаем по формуле: $C_n^k = \frac{n!}{k! \cdot (n-k)!}$

In [60]:

```
def get_combination(n, k):
    "сочетание"
    c = np.math.factorial(n) // (np.math.factorial(k) * np.math.factorial(n - k))
    return c
```

In [61]:

```
 p_1 = get_combination(50,3) / get_combination(60,3)

print(f'ОТВЕТ: вероятность, что все выбранные вопросы знакомы студенту равна {round(p_1
,5)}')
```

```
ОТВЕТ: вероятность, что все выбранные вопросы знакомы студенту равна 0.572
76
```

Задача_3

Из 60 вопросов, входящих в экзаменационные билеты, студент знает 50. Случайным образом студент вытягивает 3 вопроса. Какова вероятность что два из трёх вопросов знакомы студенту?

Решение:

Наше событие A - все выбранные вопросы знакомы студенту.
Вероятность найдем по формуле классической вероятности: $P(A) = \frac{m}{n}$

Где: n — общее число исходов (3 из 60) ($n = C_{60}^{3}$) , а m — число исходов, которые складываются из двух событий: A1 (2 из 50) ($m = C_{50}^{2}$) и A2 (1 из 10) ($m = C_{10}^{1}$), которые перемножаютмя (потому что независимы)

Сочетания считаем по формуле: $C_n^k = \frac{n!}{k! \cdot (n-k)!}$

In [62]:

```
A1_2x50 = get_combination(50,2)
A2_1x10 = get_combination(10,1)

p_2 = (A1_2x50 * A2_1x10) / get_combination(60,3)
print(f'ОТВЕТ: вероятность, что два из трёх вопросов знакомы студенту равна {round(p_2,
5)}')
```

```
ОТВЕТ: вероятность, что два из трёх вопросов знакомы студенту равна 0.3579
8
```

Задача_4

Допустим, имеется некоторая очень редкая болезнь (поражает 0.1% населения).
Вы приходите к врачу, вам делают тест на эту болезнь, и тест оказывается положительным.
Врач говорит вам, что этот тест верно выявляет 99% больных этой болезнью и всего лишь в 1% случаев даёт ложный положительный ответ.
***Вопрос*: какова вероятность, что вы действительно больны ей?**

***Подсказка*: используйте формулу Байеса с раскрытием знаменателя с помощью формулы полной вероятности.**

Решение:

Событие A - пациент болен ($\overline{A}$ - здоров).
Гипотеза Н_1 - тест "+".

Тогда вероятность того, что пациент болен найдем по формуле Байеса:
$$P(A|\text{Н}_1) = \frac{P(\text{Н}_1|A) \cdot P(A)}{P(\text{Н}_1)}$$

Где:
$P(A|\text{Н}_1)$ - вероятность того, что при положительном тесте пациент болен
$P(A)$ - вероятность болезни (у населения)
$P(\text{Н}_1|A)$ - вероятность положительного теста при болезни
$P(\text{Н}_1)$ - полная вероятность положительного теста

$P(\text{Н}_1)$ - полная вероятность положительного теста находится по формуле полной вероятности:
$$P(\text{Н}_1) = P(\text{Н}_1|A) \cdot P(A) + P\left(\text{Н}_1|\overline{A}\right) \cdot P\left(\overline{A}\right)$$

По условию задачи:
$P(A) = 0,001$
$P(\text{Н}_1|A) = 0,99$
$P\left(\text{Н}_1|\overline{A}\right) = 0,01$ - вероятность ошибки теста
$P\left(\overline{A}\right) = 0,999$ - вероятность отсутствия болезни у населения

In [66]:

```
p = 0.99 * 0.001 / (0.99 * 0.001 + 0.01 * 0.999)
print(f'ОТВЕТ: вероятность, того, что пациент действительно болен равна {round(p,5)}')
```

```
ОТВЕТ: вероятность, того, что пациент действительно болен равна 0.09016
```