Applied Artificial Intelligence
Practical # 3

Name	Ninad Karlekar	**Roll Number**	22306A1012
Subject/Course:	Applied Artificial Intelligence	**Class**	M.Sc. IT – Sem III
Topic	Bayes Theorem	**Batch**	1

Topic:Implement Bayes Theorem using Python

A. AIM: Suppose we are given the probability of Mike has a cold as 0.25, the probability of Mike was observed sneezing when he had cold in the past was 0.9 and the probability of Mike was observed sneezing when he did not have cold as 0.20. Find the probability of Mike having a cold given that he sneezes.

DESCRIPTION:

Bayes' Theorem states that the conditional probability of an event, based on the occurrence of another event, is equal to the likelihood of the second event given the first event multiplied by the probability of the first event.

$$P(H|E) = P(E|H) * \frac{P(H)}{P(E)}$$

Code:

```
def bayes_theorem(p_h, p_e_given_h, p_e_given_not_h):
    p_not_h = 1 - p_h
    p_e = (p_e_given_h * p_h) + (p_e_given_not_h * p_not_h)
    p_h_given_e = (p_e_given_h * p_h) / p_e
    return p_h_given_e
p_h = float(input("Enter the probability of NK having a cold: "))
p_e_given_h = float(input("Enter the probability of observing sneezing when NK has a cold: "))
p_e_given_not_h = float(input("Enter the probability of observing sneezing when NK does not have a cold: "))
result = bayes_theorem(p_h, p_e_given_h, p_e_given_not_h)
print("NK's probability of having a cold given that he sneezes (P(H|E)) is:", round(result, 2))
```

OUTPUT: -

```
========================= RESTART: D:/NK/AAI/aai_3_1.py ========================
Enter the probability of NK having a cold: 0.25
Enter the probability of observing sneezing when NK has a cold: 0.9
Enter the probability of observing sneezing when NK does not have a cold: 0.20
NK's probability of having a cold given that he sneezes (P(H|E)) is: 0.6
```

B. AIM: Suppose that a test for using a particular drug is 97% sensitive and 95% specific. That is, the test will produce 97% true positive results for drug users and 95% true negative results for non-drug users. These are the pieces of data that any screening test will have from their history of tests. Bayes' rule allows us to use this kind of data-driven knowledge to calculate the final probability. Suppose, we also know that 0.5% of the general population are users of the drug. What is the probability that a randomly selected individual with a positive test is a drug user?

DESCRIPTION:

$$P(\text{User}|+) = \frac{P(+|\text{User}).P(\text{User})}{P(+)} = \frac{P(+|\text{User}).P(\text{User})}{P(+|\text{User}).P(\text{User}) + P(+|\text{Non-user}).P(\text{Non-user})}$$

Here,

$P(\text{User}) = \text{Prevelance rate}$
$P(\text{Non-user}) = 1 - \text{Prevelance rate}$
$P(+|\text{User}) = \text{Sensitivity}$
$P(-|\text{Non-user}) = \text{Specificity}$
$P(+|\text{Non-user}) = 1 - \text{Specificity}$

Code:

```
def drug_user(prob_th=0.5,sensitivity=0.97,specificity=0.95,prevelance=0.005,verbose=True):
#FORMULA
    p_user = prevelance
    p_non_user = 1-prevelance
    p_pos_user = sensitivity
    p_neg_user = specificity
    p_pos_non_user = 1-specificity
    num = p_pos_user*p_user
    den = p_pos_user*p_user+p_pos_non_user*p_non_user
    prob = num/den
    print("Probability of the NK being a drug user is", round(prob,3))
    if verbose:
        if prob > prob_th:
            print("The NK could be an user")
        else:
            print("The NK may not be an user")
    return prob
drug_user()
```

Output:

```
========================= RESTART: D:/NK/AAI/aai_3_2.py
Probability of the NK being a drug user is 0.089
The NK may not be an user
```