# **Assignment-3**

**Problem Statement 1:** In a survey conducted by a non-banking financial company, a sample of 200 customers yielded that x of them were highly satisfied with the timely disbursal of their loans.

Write a Python code to perform the following operations:

- 1. Read an integer input that specifies the number of highly satisfied customers
- 2. Calculate an approximate 90% confidence interval for the proportion of the loan customers who are highly satisfied with disbursal time
- Find out the Margin of Error using scipy.stats.norm.ppf
- Calculate and print the confidence interval values rounded up to five decimal places and separated by a space

#### Note:

- Margin of Error = Critical Value \* Standard Error of Statistic
- Confidence Interval = Sample Statistic ± Margin of Error

Example: Let's say 172 out of 200 customers were highly satisfied. Sample Input:

The confidence interval should be printed as -

Sample Output: 0.82856 0.89144

**Problem Statement 2:** A radar unit is used to measure the speeds of cars on a motorway. The speeds are normally distributed with a mean of 75 km/hr and a standard deviation of 15 km/hr.

Write a Python code to perform the following operations:

- 1. Find the probability that a car picked at random is traveling at more than X km/hr
- Take the speed X as an input
- Print the probability value rounded up to four decimal places

#### Hint:

Use Normal Distribution.

### Sample Input:

100

### Sample Output:

0.0478

**Problem Statement 3:** Write a Python program to load the "kerala.csv" data into a DataFrame and perform the following tasks:

- 1. Explore the DataFrame using info() and describe() functions
- 2. June and July are the peak months of rainfall. Consider that if it rains more than 500mm, then chances of flood become more; create a Datarame with columns "YEAR", "JUN\_GT\_500" (Contains a boolean value to show whether it rained more than 500 mm in the month of June), "JUL\_GT\_500" (Contains a boolean value to show whether it rained more than 500 mm in the month of July), and "FLOODS" (Contains a boolean value to show whether it flooded that year)
- 3. Calculate the probability of flood given it rained more than 500 mm in June  $(P(A \mid B))$
- 4. Calculate the probability of rain more than 500 mm in June, given it flooded that year (P(B|A))
- 5. Probability of flood given it rained more than 500 mm in July
- 6. Probability of rain more than 500 mm in July given it flooded that year (P(B|A))

## Sample Input:



### Sample Output:

 June and July are the peak months of rainfall. Consider that if it rains more than 500mm, then chances of flood become more; create a Datarame with columns – "YEAR", "JUN\_GT\_500", "JUL\_GT\_500", and "FLOODS"

	YEAR	JUN_GT_500	JUL_GT_500	FLOODS	COUNT
0	1901	1	1	1	1
1	1902	0	1	1	1
2	1903	1	1	1	1
3	1904	1	1	1	1
4	1905	1	1	0	1

Calculate the probability of flood given it rained more than 500 mm in June (P(A|B))

```
Probailitity of flood given it rained more than 500 mm in June (P(A|B)): P(Flood|June): 0.59375
```

3. Calculate the probability of rain more than 500 mm in June, given it flooded that year (P(B|A))

4. Probability of flood given it rained more than 500 mm in July

```
Probabilitity of flood given it rained more than 500 mm in July: P(Flood|July): 0.59375
```

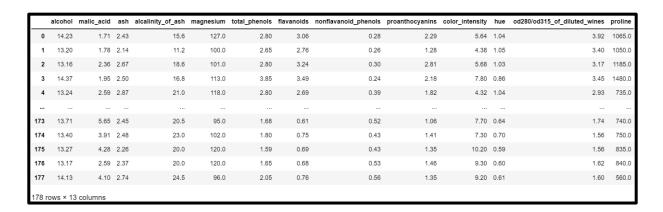
5. Probability of rain more than 500 mm in July given it flooded that year (P(B|A))

```
Probability of rain more than 500 mm in July given it flooded that year (P(B|A)): P(July|Flood): 0.95000000000000002
```

**Problem Statement 4**: Write a Python program to load the wine dataset using the Sklearn library to a DataFrame and perform the following tasks:

- 1. Convert the dataset into DataFrame using pandas.
- 2. Generate the sample size of 50 and give a random state as 100.
- 3. Calculate Z-critical, Margin of Error, and Confidence Interval for alcohol at 95% significance interval on generated sample data.

### **Sample Input:**



## **Sample Output:**

Z-critical value: 1.6448536269514722 Margin of Error: 0.17880519784197366

Confidence Interval: (12.794794802158027, 13.152405197841976)

