

## Experiment -19

```
import math

# Dataset: [Income, Age, Loan Amount, Class]
# Class: 1 = Loan Approved, 0 = Loan Rejected
data = [
    [50000, 25, 200000, 1],
    [30000, 22, 150000, 0],
    [70000, 45, 100000, 1],
    [25000, 30, 180000, 0],
    [90000, 50, 120000, 1],
    [40000, 28, 160000, 0]
]

# Separate data by class
classes = {}
for row in data:
    classes.setdefault(row[-1], []).append(row[:-1])

# Mean and Variance
def mean(nums): return sum(nums) / len(nums)

def var(nums):
    m = mean(nums)
    return sum((x - m) ** 2 for x in nums) / len(nums) + 1e-6 # avoid zero variance

# Gaussian Probability
def gaussian(x, m, v):
    return (1 / math.sqrt(2 * math.pi * v)) * math.exp(-(x - m) ** 2 / (2 * v))

# Train Model
stats = {}
for c in classes:
    stats[c] = [(mean(col), var(col)) for col in zip(*classes[c])]

# Prediction Function
def predict(sample):
    probs = {}
    for c in stats:
        probs[c] = 1
        for i in range(len(sample)):
            m, v = stats[c][i]
            probs[c] *= gaussian(sample[i], m, v)
```

```
return max(probs, key=probs.get)
```

```
# Test Sample
```

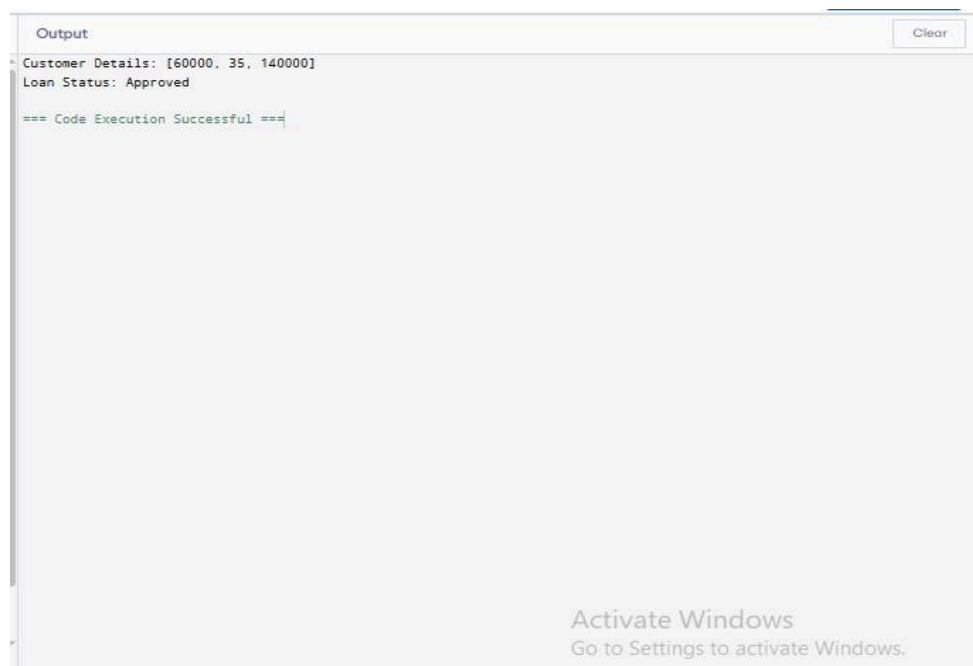
```
test = [60000, 35, 140000]
```

```
result = predict(test)
```

```
print("Customer Details:", test)
```

```
print("Loan Status:", "Approved" if result == 1 else "Rejected")
```

Output:



The screenshot shows a Jupyter Notebook output cell. The title bar of the output window is labeled "Output" and has a "Clear" button on the right. The output text is as follows:

```
Customer Details: [60000, 35, 140000]  
Loan Status: Approved  
=== Code Execution Successful ===
```

At the bottom of the output window, there is a watermark that reads "Activate Windows" and "Go to Settings to activate Windows."