

## EXPERIMENT – 6

### PROGRAM:

```
print("Enter number of samples:")
```

```
n = int(input())
```

```
print("Enter number of features:")
```

```
m = int(input())
```

```
print("Enter data (features and class):")
```

```
X = []
```

```
y = []
```

```
for _ in range(n):
```

```
    data = input().split()
```

```
    X.append(data[:-1])
```

```
    y.append(data[-1])
```

```
y_pred = []
```

```
classes = list(set(y))
```

```
for sample in X:
```

```
    class_probs = {}
```

```
    total = len(y)
```

```
for cls in classes:
```

```
    class_count = y.count(cls)
```

```
    prior = class_count / total
```

```
    likelihood = 1.0
```

```
for i in range(m):
```

```
feature = sample[i]

feature_count = sum(1 for j in range(n) if y[j] == cls and X[j][i] == feature)

likelihood *= (feature_count + 1) / (class_count + m)

class_probs[cls] = prior * likelihood

pred = max(class_probs, key=class_probs.get)
y_pred.append(pred)

cm = [[0]*len(classes) for _ in range(len(classes))]

cls_idx = {cls:i for i, cls in enumerate(classes)}

for true, pred in zip(y, y_pred):
    cm[cls_idx[true]][cls_idx[pred]] += 1

print("Confusion Matrix:")
for row in cm:
    print(row)

accuracy = sum(cm[i][i] for i in range(len(classes))) / n
print("Accuracy:", accuracy)
```