

## Experiment 2.2

### Naive Bayes

NAME – RAJDEEP JAISWAL  
UID – 20BCS2761  
BRANCH – CSE BTECH  
SEC – WM 902 B  
SUB MACHINE LEARNING LAB

#### 1. Aim/Overview of the practical:

Apply naive bayes on any dataset.

#### 2. Source Code:

```
import pandas as pd
```

```
[47] data_path = "/content/drive/MyDrive/ML_Lab/playsheet_dataset.csv"
```

```
[48] df = pd.read_csv(data_path)
```

```
[49] df.head()
```

	Outlook	Temp	Humidity	Windy	Play
0	Rainy	Hot	High	f	no
1	Rainy	Hot	High	t	no
2	Overcast	Hot	High	f	yes
3	Sunny	Mild	High	f	yes
4	Sunny	Cool	Normal	f	yes

```
[52] df['Play'].value_counts()
```

```
yes    9  
no     5  
Name: Play, dtype: int64
```

```
✓ [53] ## Before feeding it to naive bayes we have to make it in integer form
1s overlook = pd.get_dummies(df['Outlook'], drop_first = True, prefix = 'Overlook_')
```

```
✓ [54] temp = pd.get_dummies(df['Temp'], drop_first = True, prefix = 'Temp_')
```

```
✓ [55] humidity = pd.get_dummies(df['Humidity'], drop_first = True, prefix = 'Humidity_')
```

```
✓ [56] windy = pd.get_dummies(df['Windy'], drop_first = True, prefix = 'Windy_')
```

```
✓ [57] df['Play'] = df['Play'].map({"yes":1, "no": 0})
```

```
✓ [58] ## Now merging all the data
0s final_df = pd.concat([df, overlook, temp, humidity, windy], axis = 1)
```

```
✓ [59] ## Now dropping the unnecessary columns
0s final_df.drop(['Outlook', 'Temp', 'Humidity', 'Windy'], axis = 1, inplace = True)
```

```
✓ [60] final_df.head()
```

	Play	Overlook_Rainy	Overlook_Sunny	Temp_Hot	Temp_Mild	Humidity_Normal	Windy_t
0	0	1	0	1	0	0	0
1	0	1	0	1	0	0	1
2	1	0	0	1	0	0	0

```
✓ [63] ## splitting the data into independent and dependent variable  
0s X = final_df.drop('Play', axis = 1)  
y = final_df['Play']
```


```
✓ [64] ## Now splitting the data into train and test split  
0s from sklearn.model_selection import train_test_split
```


```
✓ [65] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 10)
```

```
✓ [66] ### now applying the naive bayes classifier  
0s from sklearn.naive_bayes import GaussianNB
```

```
✓ [67] nb = GaussianNB()  
0s nb.fit(X_train, y_train)  
pred = nb.predict(X_test)
```

```
✓ [68] ## now testing the accuracy score of model  
0s from sklearn.metrics import accuracy_score, confusion_matrix
```

```
✓  accuracy_score(pred, y_test)  
0s ## Got 100% of accuracy
```

```
 0.6666666666666666
```

```
✓ [70] confusion_matrix(pred, y_test)  
0s
```

```
array([[0, 0],  
       [1, 2]])
```

```
[ ]
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Learning outcomes (What I have learnt):

1. Learn about the Naive Bayes algorithm
2. Learn to perform the Naive Bayes algorithm on weather dataset
3. Learnt about the exploratory data analysis
4. Learn to optimize the Model
5. Got the clear concept of Naive Bayes classifier

## Evaluation Grid :

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Student Performance (Conduct of experiment) objectives/Outcomes.		12
2.	Viva Voce		10
3.	Submission of Work Sheet (Record)		8
	Total		30