

▼ MACHINE LEARNING LABORATORY

1818128_JAYASREE T

☆ EXP 1 : NAIVE BAYES CLASSIFIER

▼ Importing the Dataset

```
[1] import pandas as pd
df = pd.read_csv("play_tennis.csv")
# Drop Day column as it is not required to build the model
df.drop('day', axis='columns', inplace=True)
df.head()
```

	outlook	temp	humidity	wind	play
0	Sunny	Hot	High	Weak	No
1	Sunny	Hot	High	Strong	No
2	Overcast	Hot	High	Weak	Yes
3	Rain	Mild	High	Weak	Yes
4	Rain	Cool	Normal	Weak	Yes



1818128_JAYASREE T_EXP 1.ipynb

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment

Share



+ Code + Text

✓ RAM
Disk

Editing

▼ Label Encoding

```
[2] # LabelEncoder converts data into numbers from 0 to (n-1)

from sklearn.preprocessing import LabelEncoder
df[df.columns[:]] = df[df.columns[:]].apply(LabelEncoder().fit_transform)
df.head()
```

	outlook	temp	humidity	wind	play
0	2	1	0	1	0
1	2	1	0	0	0
2	0	1	0	1	1
3	1	2	0	1	1
4	1	0	1	1	1

▼ Categorising Features(X) and target(y) variables

```
[3] X = df.iloc[:, :-1]
     y = df.iloc[:, -1]
```

▼ Splitting Dataset into training and testing sets



1818128_JAYASREE T_EXP 1.ipynb

File Edit View Insert Runtime Tools Help [All changes saved](#)

Comment

Share



+ Code + Text

✓ RAM
Disk

Editing

▼ Splitting Dataset into training and testing sets

```
[4] from sklearn.model_selection import train_test_split
     X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3,random_state=1)
     len(X_train)
```

9

▼ Training the model

```
[5] from sklearn.naive_bayes import GaussianNB
     model = GaussianNB()
     model.fit(X_train, y_train) # Fit adjusts the weight according to the data values.
```

```
GaussianNB(priors=None, var_smoothing=1e-09)
```

▼ Measuring the accuracy

```
[17] from sklearn.metrics import accuracy_score
      y_pred = model.predict(X_test)
      accuracy = accuracy_score(y_test, y_pred)
      print("The Accuracy is: ",accuracy)
```

```
The Accuracy is: 0.8
```

1818128_JAYASREE T_EXP 1.ipynb

colab.research.google.com/drive/1Hy0CZjbsgf4srwnrpo6_bk6PDlkAEx08#scrollTo=sOEFmgbpr_93

1818128_JAYASREE T_EXP 1.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

Making predictions

[14] df.loc[1] # For this condition,the prediction is made

outlook 2
temp 1
humidity 0
wind 0
play 0
Name: 1, dtype: int64

[20] # Outlook=Sunny(2), Temperature=Hot(1), Humidity=High(0), Wind=Strong(0)

result = model.predict([[2,1,0,0]])
print("Predicted value: ",result)
if result:
 print("Play Tennis: YES")
else:
 print("Play Tennis: NO")

Predicted value: [0]
Play Tennis: NO

If x=(Outlook=Sunny, Temperature=Hot, Humidity=High, Wind=Strong)

The predicted value gives [0] that is **No**.