11-Naive_Bayes

October 20, 2024

1 Naive Bayes Classifier

The **Naive Bayes classifier** is a family of probabilistic classifiers that rely on Bayes' Theorem and the assumption of feature independence. The classifier is called naive because it assumes that all the features in a dataset are independent of each other, which is rarely true in real-world scenarios, but it simplifies the computation.

Naive Bayes works by calculating the probabilities for each class and then predicting the class with the highest probability based on the input features. It is a powerful and simple algorithm for classification tasks, especially in domains like text classification (e.g., spam detection, sentiment analysis).

When to Use Naive Bayes Classifier? Naive Bayes is particularly useful in the following scenarios:

- **When working with high-dimensional datasets**: It performs well with many features, such as in text classification where each word can be a feature.
- **When the assumption of conditional independence holds**: Even when this assumption is not completely true, Naive Bayes often performs surprisingly well in practice.
- **For real-time predictions**: Naive Bayes is computationally efficient and can handle large datasets quickly, making it suitable for real-time applications.
- **Text classification, sentiment analysis, and spam filtering**: These tasks involve high-dimensional and sparse data, where Naive Bayes shines.
- **Categorical input features**: The algorithm handles categorical variables well, especially in its Multinomial Naive Bayes variant.

Who Should Use Naive Bayes Classifier? Naive Bayes is an excellent choice for:

- **Data scientists and machine learning practitioners** who need a fast, simple, and interpretable model for classification tasks.
- **Business analysts** looking for text-based solutions, such as customer sentiment analysis, email spam filtering, or document categorization.
- **Researchers and developers** working with large-scale text processing, like in Natural Language Processing (NLP).
- **Beginners in machine learning**, as Naive Bayes is easy to implement and understand, while still being effective in many real-world scenarios.

Advantages of Naive Bayes Classifier:

- **Fast and efficient**: Naive Bayes is computationally efficient and works well with large datasets.
- **Handles high-dimensional data**: Especially useful for text classification problems where the data is sparse and high-dimensional.
- **Simple to implement**: The algorithm is simple to code and interpret, making it a good starting point for beginners.
- **Requires less training data**: It performs well with relatively small datasets because it estimates fewer parameters.
- **Performs well in practice**: Even though the independence assumption rarely holds true, Naive Bayes often works surprisingly well.

1.0.1 Disadvantages of Naive Bayes Classifier:

- **Strong independence assumption**: The assumption that features are conditionally independent given the class can lead to suboptimal results if this condition is not met in the data.
- **Zero probability problem**: If a feature value that wasn't seen in the training data appears in the test set, the model will assign a zero probability to that class. This can be mitigated using techniques like Laplace smoothing.
- **Not suitable for continuous data without modifications**: The basic form of Naive Bayes does not handle continuous data well unless it is adapted (e.g., Gaussian Naive Bayes).
- **Not ideal for highly correlated features**: Naive Bayes struggles when features are correlated since the independence assumption is violated.

Real-World Applications of Naive Bayes Classifier:

- **Email Spam Detection**: Classifies emails as spam or not based on the occurrence of certain words.
- **Text Classification**: Categorizes documents or news articles into different topics based on word occurrences (Multinomial Naive Bayes).
- **Sentiment Analysis**: Determines whether a review or a tweet is positive or negative by analyzing word frequencies (commonly used in NLP).
- **Medical Diagnosis**: Predicts diseases based on symptoms by calculating the probability of each condition.
- **Recommender Systems**: Helps in filtering and recommending content based on a user's previous interactions and preferences.

[55]:
```
import pandas as pd
import numpy as np

from sklearn.model_selection import train_test_split
```

```
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import classification_report
from sklearn.model_selection import GridSearchCV

num_items = 41
data = {
    'City Population': np.random.randint(10000, 1000000, num_items),
    'Continent': np.random.choice(['Asia', 'Europe', 'North America', 'South␣
 ↪America'], num_items),
    'Venue Capacity': np.random.randint(500, 20000, num_items),
    'Day Of Week': np.random.choice(['Monday', 'Tuesday', 'Wednesday',␣
 ↪'Thursday', 'Friday', 'Saturday', 'Sunday'], num_items),
    'Multiple Concerts': np.random.randint(0, 2, num_items),
    'Sold Out': np.random.randint(0, 2, num_items)
}

df = pd.DataFrame(data)
#df = pd.read_csv('pear_jam_tour2.csv', encoding="unicode_escape")
df
```

[55]:

	City Population	Continent	Venue Capacity	Day Of Week	\
0	491817	North America	16665	Thursday	
1	701437	Europe	9246	Monday	
2	166930	Europe	8403	Thursday	
3	588655	South America	4672	Saturday	
4	388841	South America	1405	Friday	
5	882813	Asia	18118	Sunday	
6	158157	Asia	1666	Wednesday	
7	117802	Asia	14382	Sunday	
8	44163	North America	2502	Saturday	
9	893911	North America	17897	Friday	
10	981604	Europe	15859	Tuesday	
11	780283	North America	19257	Wednesday	
12	265129	Asia	4524	Thursday	
13	131209	Asia	14857	Thursday	
14	405167	Europe	9712	Monday	
15	84701	North America	2482	Thursday	
16	822257	North America	5842	Sunday	
17	424221	South America	17013	Saturday	
18	305672	Asia	2628	Sunday	
19	720117	North America	3999	Thursday	
20	285205	North America	6290	Saturday	
21	121093	North America	10833	Thursday	
22	240183	Asia	8535	Saturday	
23	368908	North America	19451	Tuesday	
24	989683	North America	8082	Saturday	
25	743758	South America	10294	Monday	

26	951668	Asia	15488	Friday
27	356715	Europe	1282	Wednesday
28	428998	South America	17548	Friday
29	490859	Europe	9096	Friday
30	815163	Asia	12759	Tuesday
31	568414	North America	3550	Friday
32	569051	South America	5245	Sunday
33	327147	Asia	4974	Wednesday
34	703689	North America	19689	Monday
35	368834	Europe	15419	Monday
36	259567	North America	16369	Tuesday
37	654837	North America	19461	Friday
38	560703	North America	19339	Thursday
39	466105	South America	12232	Tuesday
40	167668	South America	2890	Thursday

	Multiple Concerts	Sold Out
0	0	0
1	0	0
2	0	0
3	0	1
4	1	1
5	1	1
6	1	0
7	0	0
8	0	0
9	1	1
10	1	1
11	0	1
12	0	1
13	1	1
14	0	0
15	0	0
16	0	1
17	1	0
18	0	1
19	0	0
20	1	0
21	1	1
22	1	1
23	1	1
24	0	0
25	0	0
26	0	1
27	0	0
28	0	0
29	0	0

```
30                  1         1
31                  0         1
32                  1         1
33                  0         0
34                  0         0
35                  0         0
36                  0         0
37                  1         1
38                  1         0
39                  0         1
40                  0         1
```

[56]:
```
df2 = pd.get_dummies(df[['Continent', 'Day Of Week']])

df2
```

[56]:

	Continent_Asia	Continent_Europe	Continent_North America
0	False	False	True
1	False	True	False
2	False	True	False
3	False	False	False
4	False	False	False
5	True	False	False
6	True	False	False
7	True	False	False
8	False	False	True
9	False	False	True
10	False	True	False
11	False	False	True
12	True	False	False
13	True	False	False
14	False	True	False
15	False	False	True
16	False	False	True
17	False	False	False
18	True	False	False
19	False	False	True
20	False	False	True
21	False	False	True
22	True	False	False
23	False	False	True
24	False	False	True
25	False	False	False
26	True	False	False
27	False	True	False
28	False	False	False
29	False	True	False

30	True	False	False
31	False	False	True
32	False	False	False
33	True	False	False
34	False	False	True
35	False	True	False
36	False	False	True
37	False	False	True
38	False	False	True
39	False	False	False
40	False	False	False

	Continent_South America	Day Of Week_Friday	Day Of Week_Monday	\
0	False	False	False	
1	False	False	True	
2	False	False	False	
3	True	False	False	
4	True	True	False	
5	False	False	False	
6	False	False	False	
7	False	False	False	
8	False	False	False	
9	False	True	False	
10	False	False	False	
11	False	False	False	
12	False	False	False	
13	False	False	False	
14	False	False	True	
15	False	False	False	
16	False	False	False	
17	True	False	False	
18	False	False	False	
19	False	False	False	
20	False	False	False	
21	False	False	False	
22	False	False	False	
23	False	False	False	
24	False	False	False	
25	True	False	True	
26	False	True	False	
27	False	False	False	
28	True	True	False	
29	False	True	False	
30	False	False	False	
31	False	True	False	
32	True	False	False	
33	False	False	False	

```
34                    False                False                 True
35                    False                False                 True
36                    False                False                False
37                    False                 True                False
38                    False                False                False
39                     True                False                False
40                     True                False                False

    Day Of Week_Saturday  Day Of Week_Sunday  Day Of Week_Thursday  \
0                  False                False                 True
1                  False                False                False
2                  False                False                 True
3                   True                False                False
4                  False                False                False
5                  False                 True                False
6                  False                False                False
7                  False                 True                False
8                   True                False                False
9                  False                False                False
10                 False                False                False
11                 False                False                False
12                 False                False                 True
13                 False                False                 True
14                 False                False                False
15                 False                False                 True
16                 False                 True                False
17                  True                False                False
18                 False                 True                False
19                 False                False                 True
20                  True                False                False
21                 False                False                 True
22                  True                False                False
23                 False                False                False
24                  True                False                False
25                 False                False                False
26                 False                False                False
27                 False                False                False
28                 False                False                False
29                 False                False                False
30                 False                False                False
31                 False                False                False
32                 False                 True                False
33                 False                False                False
34                 False                False                False
35                 False                False                False
36                 False                False                False
37                 False                False                False
```

38	False	False	True
39	False	False	False
40	False	False	True

	Day Of Week_Tuesday	Day Of Week_Wednesday
0	False	False
1	False	False
2	False	False
3	False	False
4	False	False
5	False	False
6	False	True
7	False	False
8	False	False
9	False	False
10	True	False
11	False	True
12	False	False
13	False	False
14	False	False
15	False	False
16	False	False
17	False	False
18	False	False
19	False	False
20	False	False
21	False	False
22	False	False
23	True	False
24	False	False
25	False	False
26	False	False
27	False	True
28	False	False
29	False	False
30	True	False
31	False	False
32	False	False
33	False	True
34	False	False
35	False	False
36	True	False
37	False	False
38	False	False
39	True	False
40	False	False

[57]:
```
df3 = pd.concat([df, df2], axis=1)

df3
```

[57]:

	City Population	Continent	Venue Capacity	Day Of Week
0	491817	North America	16665	Thursday
1	701437	Europe	9246	Monday
2	166930	Europe	8403	Thursday
3	588655	South America	4672	Saturday
4	388841	South America	1405	Friday
5	882813	Asia	18118	Sunday
6	158157	Asia	1666	Wednesday
7	117802	Asia	14382	Sunday
8	44163	North America	2502	Saturday
9	893911	North America	17897	Friday
10	981604	Europe	15859	Tuesday
11	780283	North America	19257	Wednesday
12	265129	Asia	4524	Thursday
13	131209	Asia	14857	Thursday
14	405167	Europe	9712	Monday
15	84701	North America	2482	Thursday
16	822257	North America	5842	Sunday
17	424221	South America	17013	Saturday
18	305672	Asia	2628	Sunday
19	720117	North America	3999	Thursday
20	285205	North America	6290	Saturday
21	121093	North America	10833	Thursday
22	240183	Asia	8535	Saturday
23	368908	North America	19451	Tuesday
24	989683	North America	8082	Saturday
25	743758	South America	10294	Monday
26	951668	Asia	15488	Friday
27	356715	Europe	1282	Wednesday
28	428998	South America	17548	Friday
29	490859	Europe	9096	Friday
30	815163	Asia	12759	Tuesday
31	568414	North America	3550	Friday
32	569051	South America	5245	Sunday
33	327147	Asia	4974	Wednesday
34	703689	North America	19689	Monday
35	368834	Europe	15419	Monday
36	259567	North America	16369	Tuesday
37	654837	North America	19461	Friday
38	560703	North America	19339	Thursday
39	466105	South America	12232	Tuesday
40	167668	South America	2890	Thursday

	Multiple Concerts	Sold Out	Continent_Asia	Continent_Europe	\
0	0	0	False	False	
1	0	0	False	True	
2	0	0	False	True	
3	0	1	False	False	
4	1	1	False	False	
5	1	1	True	False	
6	1	0	True	False	
7	0	0	True	False	
8	0	0	False	False	
9	1	1	False	False	
10	1	1	False	True	
11	0	1	False	False	
12	0	1	True	False	
13	1	1	True	False	
14	0	0	False	True	
15	0	0	False	False	
16	0	1	False	False	
17	1	0	False	False	
18	0	1	True	False	
19	0	0	False	False	
20	1	0	False	False	
21	1	1	False	False	
22	1	1	True	False	
23	1	1	False	False	
24	0	0	False	False	
25	0	0	False	False	
26	0	1	True	False	
27	0	0	False	True	
28	0	0	False	False	
29	0	0	False	True	
30	1	1	True	False	
31	0	1	False	False	
32	1	1	False	False	
33	0	0	True	False	
34	0	0	False	False	
35	0	0	False	True	
36	0	0	False	False	
37	1	1	False	False	
38	1	0	False	False	
39	0	1	False	False	
40	0	1	False	False	

	Continent_North America	Continent_South America	Day Of Week_Friday	\
0	True	False	False	
1	False	False	False	
2	False	False	False	

```
3                    False                    True               False
4                    False                    True                True
5                    False                   False               False
6                    False                   False               False
7                    False                   False               False
8                     True                   False               False
9                     True                   False                True
10                   False                   False               False
11                    True                   False               False
12                   False                   False               False
13                   False                   False               False
14                   False                   False               False
15                    True                   False               False
16                    True                   False               False
17                   False                    True               False
18                   False                   False               False
19                    True                   False               False
20                    True                   False               False
21                    True                   False               False
22                   False                   False               False
23                    True                   False               False
24                    True                   False               False
25                   False                    True               False
26                   False                   False                True
27                   False                   False               False
28                   False                    True                True
29                   False                   False                True
30                   False                   False               False
31                    True                   False                True
32                   False                    True               False
33                   False                   False               False
34                    True                   False               False
35                   False                   False               False
36                    True                   False               False
37                    True                   False                True
38                    True                   False               False
39                   False                    True               False
40                   False                    True               False

    Day Of Week_Monday  Day Of Week_Saturday  Day Of Week_Sunday  \
0                False                 False               False
1                 True                 False               False
2                False                 False               False
3                False                  True               False
4                False                 False               False
5                False                 False                True
6                False                 False               False
```

7	False	False	True
8	False	True	False
9	False	False	False
10	False	False	False
11	False	False	False
12	False	False	False
13	False	False	False
14	True	False	False
15	False	False	False
16	False	False	True
17	False	True	False
18	False	False	True
19	False	False	False
20	False	True	False
21	False	False	False
22	False	True	False
23	False	False	False
24	False	True	False
25	True	False	False
26	False	False	False
27	False	False	False
28	False	False	False
29	False	False	False
30	False	False	False
31	False	False	False
32	False	False	True
33	False	False	False
34	True	False	False
35	True	False	False
36	False	False	False
37	False	False	False
38	False	False	False
39	False	False	False
40	False	False	False

	Day Of Week_Thursday	Day Of Week_Tuesday	Day Of Week_Wednesday
0	True	False	False
1	False	False	False
2	True	False	False
3	False	False	False
4	False	False	False
5	False	False	False
6	False	False	True
7	False	False	False
8	False	False	False
9	False	False	False
10	False	True	False

11	False	False	True
12	True	False	False
13	True	False	False
14	False	False	False
15	True	False	False
16	False	False	False
17	False	False	False
18	False	False	False
19	True	False	False
20	False	False	False
21	True	False	False
22	False	False	False
23	False	True	False
24	False	False	False
25	False	False	False
26	False	False	False
27	False	False	True
28	False	False	False
29	False	False	False
30	False	True	False
31	False	False	False
32	False	False	False
33	False	False	True
34	False	False	False
35	False	False	False
36	False	True	False
37	False	False	False
38	True	False	False
39	False	True	False
40	True	False	False

[58]:
```
df4 = df3.drop(columns=['Continent', 'Day Of Week'], axis=1)
df4
```

[58]:

	City Population	Venue Capacity	Multiple Concerts	Sold Out
0	491817	16665	0	0
1	701437	9246	0	0
2	166930	8403	0	0
3	588655	4672	0	1
4	388841	1405	1	1
5	882813	18118	1	1
6	158157	1666	1	0
7	117802	14382	0	0
8	44163	2502	0	0
9	893911	17897	1	1
10	981604	15859	1	1
11	780283	19257	0	1

```
12           265129            4524                   0         1
13           131209           14857                   1         1
14           405167            9712                   0         0
15            84701            2482                   0         0
16           822257            5842                   0         1
17           424221           17013                   1         0
18           305672            2628                   0         1
19           720117            3999                   0         0
20           285205            6290                   1         0
21           121093           10833                   1         1
22           240183            8535                   1         1
23           368908           19451                   1         1
24           989683            8082                   0         0
25           743758           10294                   0         0
26           951668           15488                   0         1
27           356715            1282                   0         0
28           428998           17548                   0         0
29           490859            9096                   0         0
30           815163           12759                   1         1
31           568414            3550                   0         1
32           569051            5245                   1         1
33           327147            4974                   0         0
34           703689           19689                   0         0
35           368834           15419                   0         0
36           259567           16369                   0         0
37           654837           19461                   1         1
38           560703           19339                   1         0
39           466105           12232                   0         1
40           167668            2890                   0         1

    Continent_Asia  Continent_Europe  Continent_North America  \
0            False             False                     True
1            False              True                    False
2            False              True                    False
3            False             False                    False
4            False             False                    False
5             True             False                    False
6             True             False                    False
7             True             False                    False
8            False             False                     True
9            False             False                     True
10           False              True                    False
11           False             False                     True
12            True             False                    False
13            True             False                    False
14           False              True                    False
15           False             False                     True
```

16	False	False	True
17	False	False	False
18	True	False	False
19	False	False	True
20	False	False	True
21	False	False	True
22	True	False	False
23	False	False	True
24	False	False	True
25	False	False	False
26	True	False	False
27	False	True	False
28	False	False	False
29	False	True	False
30	True	False	False
31	False	False	True
32	False	False	False
33	True	False	False
34	False	False	True
35	False	True	False
36	False	False	True
37	False	False	True
38	False	False	True
39	False	False	False
40	False	False	False

	Continent_South America	Day Of Week_Friday	Day Of Week_Monday	\
0	False	False	False	
1	False	False	True	
2	False	False	False	
3	True	False	False	
4	True	True	False	
5	False	False	False	
6	False	False	False	
7	False	False	False	
8	False	False	False	
9	False	True	False	
10	False	False	False	
11	False	False	False	
12	False	False	False	
13	False	False	False	
14	False	False	True	
15	False	False	False	
16	False	False	False	
17	True	False	False	
18	False	False	False	
19	False	False	False	

20	False	False	False
21	False	False	False
22	False	False	False
23	False	False	False
24	False	False	False
25	True	False	True
26	False	True	False
27	False	False	False
28	True	True	False
29	False	True	False
30	False	False	False
31	False	True	False
32	True	False	False
33	False	False	False
34	False	False	True
35	False	False	True
36	False	False	False
37	False	True	False
38	False	False	False
39	True	False	False
40	True	False	False

	Day Of Week_Saturday	Day Of Week_Sunday	Day Of Week_Thursday	\
0	False	False	True	
1	False	False	False	
2	False	False	True	
3	True	False	False	
4	False	False	False	
5	False	True	False	
6	False	False	False	
7	False	True	False	
8	True	False	False	
9	False	False	False	
10	False	False	False	
11	False	False	False	
12	False	False	True	
13	False	False	True	
14	False	False	False	
15	False	False	True	
16	False	True	False	
17	True	False	False	
18	False	True	False	
19	False	False	True	
20	True	False	False	
21	False	False	True	
22	True	False	False	
23	False	False	False	

24	True	False	False
25	False	False	False
26	False	False	False
27	False	False	False
28	False	False	False
29	False	False	False
30	False	False	False
31	False	False	False
32	False	True	False
33	False	False	False
34	False	False	False
35	False	False	False
36	False	False	False
37	False	False	False
38	False	False	True
39	False	False	False
40	False	False	True

	Day Of Week_Tuesday	Day Of Week_Wednesday
0	False	False
1	False	False
2	False	False
3	False	False
4	False	False
5	False	False
6	False	True
7	False	False
8	False	False
9	False	False
10	True	False
11	False	True
12	False	False
13	False	False
14	False	False
15	False	False
16	False	False
17	False	False
18	False	False
19	False	False
20	False	False
21	False	False
22	False	False
23	True	False
24	False	False
25	False	False
26	False	False
27	False	True

```
28                False                 False
29                False                 False
30                 True                 False
31                False                 False
32                False                 False
33                False                  True
34                False                 False
35                False                 False
36                 True                 False
37                False                 False
38                False                 False
39                 True                 False
40                False                 False
```

[59]:
```
X = df4.drop(columns=['Sold Out'], axis=1)
y = df4['Sold Out']

X_train, X_test, y_train, y_test = train_test_split(X,y, random_state=33,␣
↪test_size=0.2)
gnb = GaussianNB()

gnb.fit(X_train, y_train)
y_pred = gnb.predict(X_test)
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.80      0.80      0.80         5
           1       0.75      0.75      0.75         4

    accuracy                           0.78         9
   macro avg       0.78      0.78      0.78         9
weighted avg       0.78      0.78      0.78         9
```

[60]:
```
gnb.score(X_train, y_train)
```

[60]:
```
0.5625
```

[61]:
```
gnb.score(X_test, y_test)
```

[61]:
```
0.7777777777777778
```

1.0.2 Add in parameter

[66]:
```
param_grid = {
    'var_smoothing': [0.00000001, 0.000000001, 0.00000001]
}

grid_search = GridSearchCV(gnb,param_grid, cv=5, scoring='accuracy', n_jobs=-1)
grid_search.fit(X_train, y_train)
```

[66]:
```
GridSearchCV(cv=5, estimator=GaussianNB(), n_jobs=-1,
             param_grid={'var_smoothing': [1e-08, 1e-09, 1e-08]},
             scoring='accuracy')
```

[67]:
```
grid_search.best_params_
```

[67]:
```
{'var_smoothing': 1e-08}
```

[68]:
```
grid_search.best_score_
```

[68]:
```
0.4095238095238095
```