

Semester	T.E. Semester V – Computer Engineering
Subject	Data Warehousing and Mining
Subject Professor In-charge	Prof. Kavita Shirsat
Assisting Teachers	Prof. Kavita Shirsat
Laboratory	M-313A

Student Name	Vibodh Bhosure
Roll Number	20102A0032
Grade and Subject Teacher's Signature	

Experiment Number	04	
Experiment Title	To implement Naïve Bayes Algorithm and find the class name for the given set of attributes.	
Resources / Apparatus Required	Hardware: Computer system	Software: Python
Description	<p>The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:</p> <p>Naïve: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.</p> <p>Bayes: It is called Bayes because it depends on the principle of Bayes' Theorem.</p> <p>Working of Naïve Bayes' Classifier can be understood with the help of the below example:</p> <p>Suppose we have a dataset of weather conditions and corresponding target variable "Play". So using this dataset we need to decide that whether we should play or not on a particular day according to the weather conditions. So to solve this problem, we need to follow the below steps:</p>	

	<ol style="list-style-type: none"> 1. Convert the given dataset into frequency tables. 2. Generate Likelihood table by finding the probabilities of given features. 3. Now, use Bayes theorem to calculate the posterior probability.
Program	<pre># -*- coding: utf-8 -*- """NaiveBayes.ipynb Automatically generated by Colaboratory. Original file is located at https://colab.research.google.com/drive/1H7tMjL iwAwctpf6XDQfZTWFacHQfQwiJ """ import pandas as pd import numpy as np import io from google.colab import files uploaded = files.upload() df = pd.read_csv(io.BytesIO(uploaded['Book1.csv'])) print(df) attributes = {} attributes['Age'] = input('age:') attributes['Income'] = input('income:') attributes['Student'] = input('student:') attributes['Credit_Rating'] = input('credit rating:') attributes count_yes = 0 p = dict() for i in df['Buys_Computer']: if i == 'Yes': count_yes += 1 count_no = len(df) - count_yes p['Yes'] = count_yes/len(df) p['No'] = count_no/len(df) p def count_value(attribute1, value1, attribute2, value2): global df count = 0 for index, row in df.iterrows(): if row[attribute1] == value1 and row[attribute2] == value2: count +=1 return count p_yes = 1</pre>

	<pre>p_no = 1 for u,v in attributes.items(): p_yes *= count_value(u, v, 'Buys_Computer', 'Yes')/count_yes p_no *= count_value(u, v, 'Buys_Computer', 'No')/count_no p_yes *= p['Yes'] p_no *= p['No'] print(p_yes, p_no) if p_yes > p_no: print('Class: Yes') else: print('Class: No')</pre>																																																																																																									
Output	<div>Dataset –</div> <table><thead><tr><th></th><th>RID</th><th>Age</th><th>Income</th><th>Student</th><th>Credit_Rating</th><th>Buys_Computer</th></tr></thead><tbody><tr><td>0</td><td>1</td><td>Youth</td><td>high</td><td>NO</td><td>Fair</td><td>No</td></tr><tr><td>1</td><td>2</td><td>Youth</td><td>high</td><td>NO</td><td>Excellent</td><td>No</td></tr><tr><td>2</td><td>3</td><td>Middle_aged</td><td>high</td><td>NO</td><td>Fair</td><td>Yes</td></tr><tr><td>3</td><td>4</td><td>senior</td><td>medium</td><td>NO</td><td>Fair</td><td>Yes</td></tr><tr><td>4</td><td>5</td><td>senior</td><td>Low</td><td>Yes</td><td>Fair</td><td>Yes</td></tr><tr><td>5</td><td>6</td><td>senior</td><td>Low</td><td>Yes</td><td>Excellent</td><td>No</td></tr><tr><td>6</td><td>7</td><td>Middle_aged</td><td>Low</td><td>Yes</td><td>Excellent</td><td>Yes</td></tr><tr><td>7</td><td>8</td><td>Youth</td><td>medium</td><td>NO</td><td>Fair</td><td>No</td></tr><tr><td>8</td><td>9</td><td>Youth</td><td>Low</td><td>Yes</td><td>Fair</td><td>Yes</td></tr><tr><td>9</td><td>10</td><td>senior</td><td>medium</td><td>Yes</td><td>Fair</td><td>Yes</td></tr><tr><td>10</td><td>11</td><td>Youth</td><td>medium</td><td>Yes</td><td>Excellent</td><td>Yes</td></tr><tr><td>11</td><td>12</td><td>Middle_aged</td><td>medium</td><td>NO</td><td>Excellent</td><td>Yes</td></tr><tr><td>12</td><td>13</td><td>Middle_aged</td><td>high</td><td>Yes</td><td>Fair</td><td>Yes</td></tr><tr><td>13</td><td>14</td><td>senior</td><td>medium</td><td>NO</td><td>Excellent</td><td>No</td></tr></tbody></table> <div>Set of Attributes –</div> <div>age:senior income:high student:NO credit rating:Fair {'Age': 'senior', 'Income': 'high', 'Student': 'NO', 'Credit_Rating': 'Fair'}</div> <div>{'Yes': 0.6428571428571429, 'No': 0.35714285714285715}</div> <div>Class Predicted –</div> <div>Class: No</div>		RID	Age	Income	Student	Credit_Rating	Buys_Computer	0	1	Youth	high	NO	Fair	No	1	2	Youth	high	NO	Excellent	No	2	3	Middle_aged	high	NO	Fair	Yes	3	4	senior	medium	NO	Fair	Yes	4	5	senior	Low	Yes	Fair	Yes	5	6	senior	Low	Yes	Excellent	No	6	7	Middle_aged	Low	Yes	Excellent	Yes	7	8	Youth	medium	NO	Fair	No	8	9	Youth	Low	Yes	Fair	Yes	9	10	senior	medium	Yes	Fair	Yes	10	11	Youth	medium	Yes	Excellent	Yes	11	12	Middle_aged	medium	NO	Excellent	Yes	12	13	Middle_aged	high	Yes	Fair	Yes	13	14	senior	medium	NO	Excellent	No
	RID	Age	Income	Student	Credit_Rating	Buys_Computer																																																																																																				
0	1	Youth	high	NO	Fair	No																																																																																																				
1	2	Youth	high	NO	Excellent	No																																																																																																				
2	3	Middle_aged	high	NO	Fair	Yes																																																																																																				
3	4	senior	medium	NO	Fair	Yes																																																																																																				
4	5	senior	Low	Yes	Fair	Yes																																																																																																				
5	6	senior	Low	Yes	Excellent	No																																																																																																				
6	7	Middle_aged	Low	Yes	Excellent	Yes																																																																																																				
7	8	Youth	medium	NO	Fair	No																																																																																																				
8	9	Youth	Low	Yes	Fair	Yes																																																																																																				
9	10	senior	medium	Yes	Fair	Yes																																																																																																				
10	11	Youth	medium	Yes	Excellent	Yes																																																																																																				
11	12	Middle_aged	medium	NO	Excellent	Yes																																																																																																				
12	13	Middle_aged	high	Yes	Fair	Yes																																																																																																				
13	14	senior	medium	NO	Excellent	No																																																																																																				
Conclusion:	Hence a data set of people buying computer was passed and a random set of values for attributes was give and class label for that set of attributes was found.																																																																																																									