WEEK 7: TUTORIAL

TASK :1

Step1: calculate probabilities for Acc -yes and No
P(Accident=Yes) =5/10 = 0.5
P (Accident=No) =5/10 = 0.5

Step 2: Calculate Conditional Probabilities for given sample
P (Rain Accident=Yes) = 1/5 = 0.2
P (Good |Accident=Yes) = 1/5 =0.2
P (Normal Accident=Yes) = 1/5 =0.2
P (No Accident=Yes) = 2/5 =0.4

P (Rain Accident-No) = 2/5 = 0.4
P (Good Accident-No) = 3/5 = 0.6
P (Normal Accident-No) = 2/5 = 0.4
P (No Accident=No) = 4/5 = 0.8

P (X accident-yes) =0.2 x 0.2 0.2 x 0.4 = 0.0032
P (X accident=no) = 0.4 x 0.6 x 0.4 × 0.8 = 0.0768
P (Accident-yes X) = 0.0032 x 0.5 = 0.0016
P (Accident=no X) = 0.0768 x 0.5 = 0.0384

Task 2 – Weather-Based Game Prediction (Dataset 2)

Goal: Use Dataset 2 (the weather data) to classify whether to play or not play.
Class label: play $\in$ {yes, no}. Attributes: outlook, temperature, humidity, windy.

Question 1: X = (sunny, hot, high, false)

We want P(play = yes | X) and P(play = no | X).

1. From Dataset 2, count how many rows have play = yes and play = no.
   (In the standard weather dataset there are 9 "yes" and 5 "no" out of 14 examples.)

2. Compute priors:
   - P(play = yes) = 9 / 14
   - P(play = no) = 5 / 14.

3. For each attribute value in X, compute the conditional probabilities for both classes.
   Using your counts from the table:
   - P(outlook = sunny | play = yes) = (# rows with outlook=sunny AND play=yes) / 9
   - P(outlook = sunny | play = no) = (# rows with outlook=sunny AND play=no) / 5
   - P(temperature = hot | play = yes/no)
   - P(humidity = high | play = yes/no)
   - P(windy = false | play = yes/no)

4. Multiply to get the Naïve Bayes scores for each class:
   - Score_yes = P(play = yes) × P(sunny | yes) × P(hot | yes) × P(high | yes) × P(false | yes)
   - Score_no = P(play = no) × P(sunny | no) × P(hot | no) × P(high | no) × P(false | no).

5. Compute the normalised posteriors (optional):
  • P (yes | X) = Score_yes / (Score_yes + Score_no)
  • P(no | X) = Score no / (Score yes + Score no).

Task 3 — Loan Approval (Dataset 3)

EmploymentStatus, CreditHistory, IncomeLevel → LoanApproved

1. Employed, Good, High → Yes
2. Unemployed, Bad, Low → No
3. Employed, Good, Medium → Yes
4. Employed, Bad, Medium → No
5. Unemployed, Good, Low → Yes

Naive Bayes Tutorial

1) Priors

Total rows N=5N=5N=5.

- P(Yes)=3/5P (\text {Yes}) = 3/5P(Yes)=3/5 (rows 1,3,5)
- P(No)=2/5P (\text {No}) = 2/5P(No)=2/5 (rows 2,4)

Task 3 – Loan Approval Prediction (Dataset 3)

Dataset 3 fields: EmploymentStatus (Employed/Unemployed), CreditHistory (Good/Bad), IncomeLevel (High/Medium/Low), LoanApproved (Yes/No).

EmploymentStatus	CreditHistory	IncomeLevel	LoanApproved
Employed	Good	High	Yes
Unemployed	Bad	Low	No
Employed	Good	Medium	Yes
Employed	Bad	Medium	No
Unemployed	Good	Low	Yes

Question 1: Applicant A

Applicant A: EmploymentStatus = Employed, CreditHistory = Good, IncomeLevel = Medium.
We want P(LoanApproved = Yes | A) and P(LoanApproved = No | A).

1. Count class frequencies from the table:
  • LoanApproved = Yes: 3 rows
  • LoanApproved = No: 2 rows
  • Total = 5 rows.

2. Priors:
- P(Yes) = 3 / 5
- P(No) = 2 / 5.

3. Conditional probabilities for Applicant A.
For the YES class (look only at the 3 'Yes' rows):
- P(Employed | Yes) = 2 / 3
- P(Good | Yes) = 3 / 3
- P(Income = Medium|Yes) = 1 / 3.

For the NO class (look only at the 2 'No' rows):
- P(Employed | No) = 1 / 2
- P(Good | No) = 0 / 2 = 0
- P(Income = Medium|No) = 1 / 2.

4. Compute Naïve Bayes scores:
- Score_Yes = P(Yes) × P(Employed | Yes) × P(Good | Yes) × P(Medium | Yes)
- Score_No = P(No) × P(Employed | No) × P(Good | No) × P(Medium | No).

(Notice that P(Good | No) = 0, which will make Score_No = 0 unless you apply smoothing.)

Question 2: Applicant B

Applicant B: EmploymentStatus = Unemployed, CreditHistory = Bad, IncomeLevel = Low.
Repeat steps 1–6, this time computing:
- P(Unemployed | Yes/No)
- P(Bad | Yes/No)
- P(Low | Yes/No)

and then the corresponding scores and final decision.

Task 4 – Disease Diagnosis (Dataset 4)

Dataset 4 fields: Fever, Cough, Fatigue, TravelHistory, DiseaseDiagnosis (Positive/Negative).

Fever	Cough	Fatigue	TravelHistory	DiseaseDiagnosis
Yes	Yes	Yes	Yes	Positive
No	Yes	No	No	Negative
Yes	No	Yes	No	Positive
No	Yes	No	Yes	Negative
Yes	No	Yes	No	Positive

Question 1: New patient 1

Patient 1: Fever = Yes, Fatigue = Yes, TravelHistory = No, and Cough = No.
Objective: compute P(DiseaseDiagnosis = Positive | symptoms) using Naïve Bayes.

1. Count how many Positive and Negative cases exist:
   • Positive: 3 rows
   • Negative: 2 rows.

2. Priors:
   • P(Positive) = 3 / 5
   • P(Negative) = 2 / 5.

3. Conditional probabilities for Patient 1.
   For Positive (look only at Positive rows):
   • P(Fever = Yes | Positive) = 3 / 3 = 1
   • P(Fatigue = Yes | Positive) = 3 / 3 = 1
   • P(TravelHistory = No | Positive)= 2 / 3
   • P(Cough = No | Positive) = 2 / 3.
   For Negative (look only at Negative rows):
   • P(Fever = Yes | Negative) = 0 / 2 = 0
   • P(Fatigue = Yes | Negative) = 0 / 2 = 0
   • P(TravelHistory = No | Negative)= 1 / 2
   • P(Cough = No | Negative) = 0 / 2 = 0.

4. Compute Naïve Bayes scores:
   • Score_Pos = P(Positive) × 1 × 1 × (2/3) × (2/3)
   • Score_Neg = P(Negative) × 0 × 0 × (1/2) × 0 = 0 (again, unless used).

Question 2: New patient 2

Patient 2: No Fever, Cough = Yes, Fatigue = No, TravelHistory = No.
Repeat the same counting and Naïve Bayes steps for this combination and compare the scores.

<u>POPRTFOLIO</u>

In Week 7, I practised using the Naïve Bayes classifier to make predictions from small labelled datasets. For **Task 1 (Accident Prediction)**, I first calculated the prior probabilities for Accident = Yes and Accident = No (both 0.5). Then I computed the conditional probabilities for the given sample values. After multiplying the likelihoods, the final scores showed **Accident = No** because the posterior for “No” (0.0384) was much higher than “Yes” (0.0016), so the model predicts that an accident is unlikely for that case.

For **Task 2 (Weather Game Prediction)**, I followed the same method using the weather dataset where Play has two classes (Yes/No). I identified the class totals (9 Yes, 5 No), calculated priors, then found the likelihoods for each attribute (outlook, temperature, humidity, windy). By multiplying these with the priors, I compared the two class scores to decide whether the model recommends playing.

For **Task 3 (Loan Approval)** and **Task 4 (Disease Diagnosis)**, I repeated the Naïve Bayes steps and noticed an important issue: **some conditional probabilities become zero** when a feature value never appears in a class. This highlights why smoothing (like Laplace smoothing) is often needed in real systems. Overall, these tasks helped me understand probability-based classification and how assumptions in Naïve Bayes affect results.