1. Given the following data about weather and playing tennis:

Outlook: Sunny, Temperature: Hot, Humidity: High, Windy: False, Play: No

Outlook: Rain, Temperature: Mild, Humidity: Normal, Windy: False, Play: Yes

Calculate the probability of playing tennis using Naive Bayes classifier. Show all

Solution

steps.

Given Data:

1. (Sunny, Hot, High, False, No)
2. (Rain, Mild, Normal, False, Yes)

New Instance to Classify:

(Outlook=Sunny, Temperature=Mild, Humidity=Normal, Windy=False)

Step 1: Calculate Prior Probabilities

- P(Play=Yes) = 1/2 = 0.5
- P(Play=No) = 1/2 = 0.5

Step 2: Calculate Likelihoods

For Play=Yes:

- P(Outlook=Sunny|Yes) = 0/1 = 0 (Laplace smoothing: (0+1)/(1+3) = 0.25)
- P(Temperature=Mild|Yes) = 1/1 = 1.0
- P(Humidity=Normal|Yes) = 1/1 = 1.0
- P(Windy=False|Yes) = 1/1 = 1.0

For Play=No:

- P(Outlook=Sunny|No) = 1/1 = 1.0
- P(Temperature=Mild|No) = 0/1 = 0 (Laplace smoothing: (0+1)/(1+3) = 0.25)
- P(Humidity=Normal|No) = 0/1 = 0 (Laplace smoothing: (0+1)/(1+2) = 0.33)
- P(Windy=False|No) = 1/1 = 1.0

Step 3: Apply Naive Bayes Formula

Using Laplace smoothing (adding 1 to numerator and k to denominator, where k is number of possible values):

Posterior for Play=Yes:

$P(Yes|X) \propto P(Yes) \times P(Sunny|Yes) \times P(Mild|Yes) \times P(Normal|Yes) \times P(False|Yes)$

$= 0.5 \times 0.25 \times 1.0 \times 1.0 \times 1.0 = 0.125$

Posterior for Play=No:

$P(No|X) \propto P(No) \times P(Sunny|No) \times P(Mild|No) \times P(Normal|No) \times P(False|No)$

$= 0.5 \times 1.0 \times 0.25 \times 0.33 \times 1.0 \approx 0.04125$

Step 4: Normalize Probabilities

Total = 0.125 + 0.04125 = 0.16625

- $P(Yes|X) = 0.125/0.16625 \approx 0.75$ (75%)
- $P(No|X) = 0.04125/0.16625 \approx 0.25$ (25%)

Final Prediction: Play Tennis (Yes) with 75% probability

2 . Derive the EM algorithm steps for a mixture of two Gaussian distributions. Explain how it handles missing data.

Solution

Problem Setup:

Given data points from two Gaussian distributions with parameters:

- μ_1, σ_1^2 (mean and variance of first Gaussian)
- μ_2, σ_2^2 (mean and variance of second Gaussian)
- π (mixing coefficient, probability a point comes from first Gaussian)

EM Algorithm Steps:

1. **Initialization:**
   - Randomly initialize parameters ($\mu_1, \sigma_1^2, \mu_2, \sigma_2^2, \pi$)
2. **Expectation Step (E-Step):**
   For each data point x_i:
   - Calculate responsibility $\gamma(z_{1i}) = P(z_{1i}=1|x_i) = \pi \cdot N(x_i|\mu_1,\sigma_1^2) / [\pi \cdot N(x_i|\mu_1,\sigma_1^2) + (1-\pi) \cdot N(x_i|\mu_2,\sigma_2^2)]$
   - $\gamma(z_{2i}) = 1 - \gamma(z_{1i})$
3. **Maximization Step (M-Step):**
   Update parameters:
   - $N_1 = \Sigma\gamma(z_{1i})$, $N_2 = \Sigma\gamma(z_{2i})$
   - $\mu_1 = (\Sigma\gamma(z_{1i})x_i)/N_1$
   - $\mu_2 = (\Sigma\gamma(z_{2i})x_i)/N_2$
   - $\sigma_1^2 = (\Sigma\gamma(z_{1i})(x_i-\mu_1)^2)/N_1$
   - $\sigma_2^2 = (\Sigma\gamma(z_{2i})(x_i-\mu_2)^2)/N_2$
   - $\pi = N_1/(N_1+N_2)$

- $\pi = N_1/(N_1+N_2)$

4. **Convergence Check:**
   Repeat E and M steps until log-likelihood converges:
   $\ln p(X|\mu,\sigma,\pi) = \Sigma \ln[\pi N(x_i|\mu_1,\sigma_1^2) + (1-\pi)N(x_i|\mu_2,\sigma_2^2)]$

Handling Missing Data:

- In E-step, for missing features, marginalize over possible values
- Or use only available features in probability calculations
- The algorithm naturally handles missing data by using expected values (E-step fills in missing information probabilistically)

3. Calculate the sample complexity for learning a threshold function on the real line with error ε and confidence δ. Show your working.

Solution

3. Sample Complexity for Threshold Function

Threshold Function: $f(x) = 1$ if $x \geq \theta$, else 0

PAC Learning Framework:

- Concept class C: All threshold functions on $\mathbb{R}$
- VC-dimension of C: 2 (can shatter 2 points but not 3)
- For ε error and δ confidence, sample complexity m satisfies:
  $m \geq (1/\varepsilon)[\ln(1/\delta) + \mathrm{VCdim}(C)\cdot\ln(1/\varepsilon)]$

Calculation:

1. VC-dimension of threshold functions = 2
2. Using standard PAC bound:
   $m \geq (1/\varepsilon)[\ln(1/\delta) + 2\ln(1/\varepsilon)]$

Example: For $\varepsilon=0.1$, $\delta=0.05$:

$m \geq (1/0.1)[\ln(1/0.05) + 2\ln(1/0.1)]$

$= 10[\ln(20) + 2\ln(10)]$

$\approx 10[3 + 2\times 2.3026]$

$\approx 10[3 + 4.6052]$

≈ 76.052

Thus, we need at least 77 samples to guarantee with 95% confidence that our error is ≤ 10%.

4. Design a Bayesian network for a student's performance prediction system considering factors like study hours, previous grades, attendance, and difficulty level. Calculate conditional probabilities for a given scenario.

Solution

Network Structure:

1. Difficulty (D): Hard, Easy
2. Study Hours (S): Low, Medium, High
3. Previous Grades (G): Poor, Average, Good
4. Attendance (A): Low, High
5. Performance (P): Fail, Pass, Excellent

Conditional Probability Tables (CPTs):

Difficulty:

- P(D=Hard) = 0.4
- P(D=Easy) = 0.6

Study Hours given Difficulty:

D	S=Low	S=Medium	S=High
Hard	0.6	0.3	0.1
Easy	0.2	0.5	0.3

Previous Grades given Study Hours:

S	G=Poor	G=Average	G=Good
Low	0.7	0.2	0.1
Medium	0.3	0.5	0.2
High	0.1	0.3	0.6

Attendance given Previous Grades:

G	A=Low	A=High
Poor	0.8	0.2
Average	0.5	0.5
Good	0.2	0.8

Performance given all parents:

Complex table combining all parent states (example rows):

D	S	G	A	P=Fail	P=Pass	P=Ex
Hard	Low	Poor	Low	0.9	0.1	0.0
Easy	High	Good	High	0.0	0.3	0.7

Example Calculation:

Find P(P=Excellent | D=Easy, S=High, G=Good, A=High) = 0.7 from CPT

5. If S is a collection of 14 examples with 9 YES and 5 NO examples in which one of the attributes is wind speed. The values of Wind can be Weak or Strong. The classification of these 14 examples are 9 YES and 5 NO. For attribute Wind, suppose there are 8 occurrences of Wind = Weak and 6 occurrences of Wind = Strong. For Wind = Weak, 6 of the examples are YES and 2 are NO. For Wind = Strong, 3 are YES and 3 are NO. Find the Entropy(weak) and Entropy(strong).

Solution

Given:

- Total examples: 14 (9 YES, 5 NO)
- Wind: Weak (8), Strong (6)
  - Weak: 6 YES, 2 NO
  - Strong: 3 YES, 3 NO

Entropy Formula:

$H(S) = -\Sigma p_i \log_2 p_i$

Entropy for Wind=Weak:

$H(Weak) = -[P(YES|Weak)\log_2 P(YES|Weak) + P(NO|Weak)\log_2 P(NO|Weak)]$

$= -[(6/8)\log_2(6/8) + (2/8)\log_2(2/8)]$

$= -[0.75 \times \log_2(0.75) + 0.25 \times \log_2(0.25)]$

$= -[0.75 \times (-0.415) + 0.25 \times (-2)]$

$= -[-0.311 - 0.5]$

$= 0.811$ bits

Entropy for Wind=Strong:

$H(Strong) = -[P(YES|Strong)\log_2 P(YES|Strong) + P(NO|Strong)\log_2 P(NO|Strong)]$

$= -[(3/6)\log_2(3/6) + (3/6)\log_2(3/6)]$

$= -[0.5 \times \log_2(0.5) + 0.5 \times \log_2(0.5)]$

$= -[0.5 \times (-1) + 0.5 \times (-1)]$

$= -[-0.5 - 0.5]$

$= 1$ bit

Information Gain Calculation (for completeness):

$H(S) = -[(9/14)\log_2(9/14) + (5/14)\log_2(5/14)] \approx 0.940$ bits

$IG = H(S) - [\ (8/14)H(Weak) + (6/14)H(Strong)\ ]$

$= 0.940 - [\ (8/14)(0.811) + (6/14)(1)\ ]$

$= 0.940 - [0.463 + 0.429] \approx 0.048$ bits