Naïve Bayes Classifier

10 Sep 2022

Vineeth N Balasubramanian

आई आई टी हैदराबाद
IIT Hyderabad

Classification Methods

- k-Nearest Neighbors
- Decision Trees
- Naïve Bayes
- Support Vector Machines
- Logistic Regression
- Neural Networks
- Ensemble Methods (Boosting, Random Forests)

Probability: Review

Random variable	• Result of tossing a coin is from {Heads, Tails} • Random var X from {1,0} • Bernoulli: $P\{X=1\} = p_o^X (1 - p_o)^{(1-X)}$
Joint and conditional probability	P(A|B) = P(A, B)/P(B)
Bayes Theorem	P(A|B) = P(B|A) P(A)/P(B)

Illustration

A	0	0	1	1	1	0
B	0	1	1	0	1	1

- P(A=1) = 3/6 = 1/2, P(A=0) = 3/6 = 1/2.
- P(B=1) = 4/6 = 2/3, P(B=0) = 2/6 = 1/3.
- P(A=1, B = 1) = 2/6 = 1/3.
- P(A=1 | B = 1) = P(A=1, B = 1) / P(B=1) = 1/2.
- P(B=1 | A = 1) = P(B=1, A = 1) / P(A=1) = 2/3.
- P(A=1 | B = 1) P(B=1)/P(A=1) = 2/3 = P(B=1 | A =1).
  - Bayes' Theorem

आई आई टी हैदराबाद
IIT Hyderabad

Naïve Bayes Classifier

- Goal: Learning function f: x -> y
  - *Y*: One of *k* classes (e.g. spam/ham, digit 0-9)
  - $X=X_1,\ldots,X_n$: Values of attributes (numeric or categorical)
- Probabilistic classification
  - Most probable class given observation: $\hat{y} = \arg\max_y P(y|x)$
- Bayesian probability of a class

$$P(y|x) = \frac{\overbrace{P(x|y)}^{\text{class model}}\overbrace{P(y)}^{\text{prior}}}{\underbrace{\sum_{y'} P(x|y')P(y')}_{\text{normalizer } P(x)}}$$

Bayes Theorem

Bayes Theorem: Example

- Given:
  - A doctor knows that meningitis causes stiff neck 50% of the time
  - Prior probability of any patient having meningitis is 1/50,000
  - Prior probability of any patient having stiff neck is 1/20

- If a patient has stiff neck, what's the probability he/she has meningitis?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

What is Naïve about it?

- Consider each attribute and class label as random variables
- Given a record with attributes $(A_1, A_2, \ldots, A_n)$
  - Goal is to predict class C
  - Specifically, we want to find the value of C that maximizes $P(C|A_1, A_2, \ldots, A_n)$
- Can we estimate $P(C|A_1, A_2, \ldots, A_n)$ directly from data?

What is Naïve about it?

- Approach:
  - compute the posterior probability $P(C \mid A_1, A_2, \ldots, A_n)$ for all values of C using the Bayes theorem

Posterior

$$P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C) P(C)}{P(A_1 A_2 \ldots A_n)}$$

Prior

Likelihood

  - Choose value of C that maximizes $P(C \mid A_1, A_2, \ldots, A_n)$
  - Equivalent to choosing value of C that maximizes $P(A_1, A_2, \ldots, A_n | C)\, P(C)$

Maximum A Posteriori (MAP) Rule

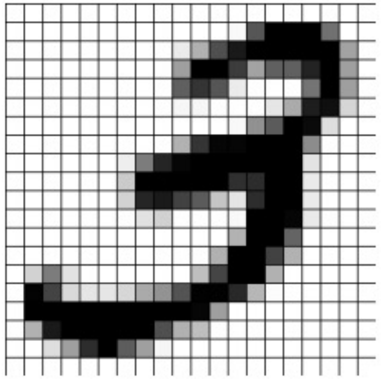
- How to estimate $P(A_1, A_2, \ldots, A_n \mid C)$?

What is Naïve about it?

$$P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C) P(C)}{P(A_1 A_2 \ldots A_n)}$$

How to compute if x is made of multiple attributes?

- 20 x 20 image of digit = 2^{400} possible combinations!



Naïve Bayes Classifier

- Assume independence among attributes A_i when class is given:
  - $P(A_1, A_2, \ldots, A_n \mid C_j) = P(A_1 \mid C_j)\, P(A_2 \mid C_j) \ldots P(A_n \mid C_j)$
  - Can estimate $P(A_i \mid C_j)$ for all A_i and C_j.
  - New point is classified to C_j if
    $P(C_j)\ \Pi_j\ P(A_i \mid C_j) = P(C_j)\, P(A_1 \mid C_j)\, P(A_2 \mid C_j) \ldots P(A_n \mid C_j)$
    is maximal

Maximum Likelihood Hypothesis

- Assume that all hypotheses (classes) are equally probable a priori, i.e., $P(C_i) = P(C_j)$ for all i,j
- This is called assuming a uniform prior. It simplifies computing the posterior:
  - $C_{ML} = \text{argmax}_c \, P(A_1, A_2, \ldots, A_n|C)$
- This hypothesis is called the maximum likelihood hypothesis.

Example

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Example: Learning Phase

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Play=*Yes*	Play=*No*
High	3/9	4/5
Normal	6/9	1/5

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

$P(\text{Play}=\textit{Yes}) = 9/14$ $P(\text{Play}=\textit{No}) = 5/14$

Example: Test Phase

- Given a new instance,
- $\mathbf{x}'$=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)
- Look up tables

P(Outlook=*Sunny*|Play=*Yes*) = 2/9	P(Outlook=*Sunny*|Play=*No*) = 3/5
P(Temperature=*Cool*|Play=*Yes*) = 3/9	P(Temperature=*Cool*|Play==*No*) = 1/5
P(Huminity=*High*|Play=*Yes*) = 3/9	P(Huminity=*High*|Play=*No*) = 4/5
P(Wind=*Strong*|Play=*Yes*) = 3/9	P(Wind=*Strong*|Play=*No*) = 3/5
P(Play=*Yes*) = 9/14	P(Play=*No*) = 5/14

- MAP rule

$P(Yes|\mathbf{x}')$: $[P(Sunny|\mathrm{Yes})P(Cool|Yes)P(High|\mathrm{Yes})P(Strong|Yes)]P(\mathrm{Play}=Yes) = 0.0053$

$P(No|\mathbf{x}')$: $[P(Sunny|\mathrm{No})\, P(Cool|\mathrm{No})P(High|No)P(Strong|No)]P(\mathrm{Play}=No) = 0.0206$

Given the fact $P(Yes|\mathbf{x}') < P(No|\mathbf{x}')$, we label $\mathbf{x}'$ to be "*No*".

Example: Another

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

A: attributes

M: mammals

N: non-mammals

$$P(A|M) = \frac{6}{7} \times \frac{6}{7} \times \frac{2}{7} \times \frac{2}{7} = 0.06$$

$$P(A|N) = \frac{1}{13} \times \frac{10}{13} \times \frac{3}{13} \times \frac{4}{13} = 0.0042$$

$$P(A|M)P(M) = 0.06 \times \frac{7}{20} = 0.021$$

$$P(A|N)P(N) = 0.004 \times \frac{13}{20} = 0.0027$$

P(A|M)P(M) > P(A|N)P(N)

=> Mammals

Pros and Cons

- **Combines prior knowledge and observed data:** prior probability of a hypothesis multiplied with probability of the hypothesis given the training data
- **Probabilistic hypothesis:** outputs not only a classification, but a probability distribution over all classes
- Robust to isolated noise points
- Handle missing values by ignoring the instance during probability estimate calculations
- Robust to irrelevant attributes
- **Incrementality:** With each training example, the prior and the likelihood can be updated dynamically: flexible and robust to errors
- Independence assumption may not hold always

Practical Issues

- For continuous attributes:
  - Discretize the range into bins
    - one ordinal attribute per bin
    - violates independence assumption
  - Two-way split: $(A < v)$ or $(A > v)$
    - choose only one of the two splits as new attribute
  - Probability density estimation:
    - Assume attribute follows a parametrized distribution, e.g. normal distribution
    - Use data to estimate parameters of distribution (e.g., mean and standard deviation) using Maximum Likelihood Estimation
    - Once probability distribution is known, can use it to estimate the conditional probability $P(A_i|c)$

Underflow Prevention

- Multiplying lots of probabilities, which are between 0 and 1 by definition, can result in floating-point underflow.
- Since log(xy) = log(x) + log(y), it is better to perform all computations by summing logs of probabilities rather than multiplying probabilities.
- Class with highest final un-normalized **log probability score** is still the most probable.

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \log P(c_j) + \sum_{i \in positions} \log P(x_i \mid c_j)$$

Density Estimation in Naïve Bayes

- Assume independence among attributes A_i when class is given:
  - $P(A_1, A_2, \ldots, A_n \mid C_j) = P(A_1 \mid C_j)\, P(A_2 \mid C_j) \ldots P(A_n \mid C_j)$
  - Can estimate $P(A_i \mid C_j)$ for all A_i and C_j.
  - New point is classified to C_j if

    $P(C_j)\ \Pi_j\, P(A_i \mid C_j) = P(C_j)\, P(A_1 \mid C_j)\, P(A_2 \mid C_j) \ldots P(A_n \mid C_j)$

    is maximal

What if this conditional distribution was Gaussian? Or a mixture of Gaussians? Or any other distribution?

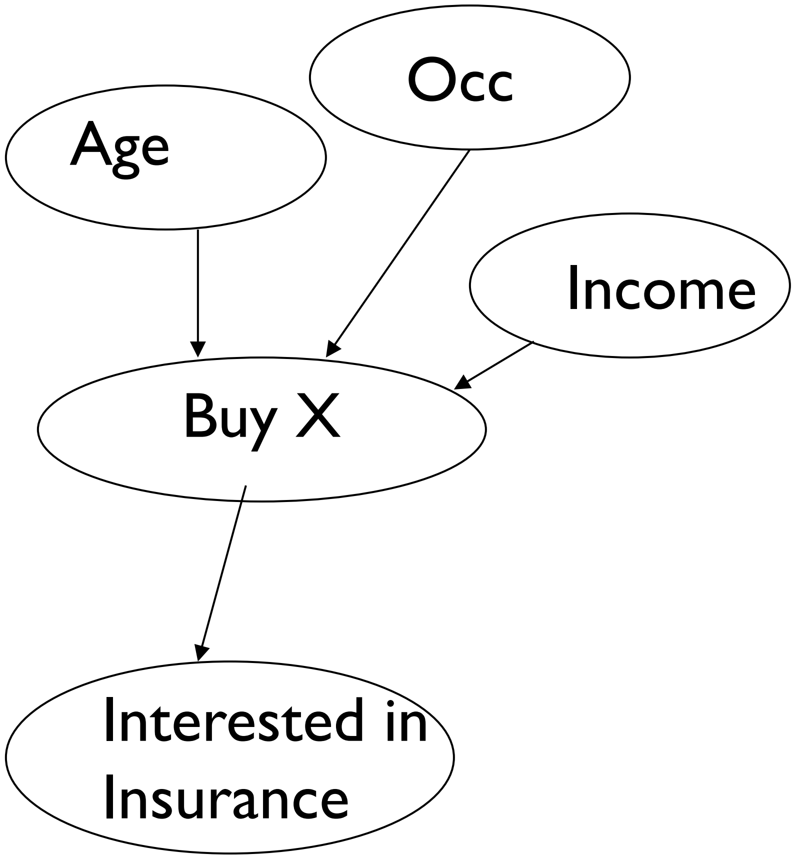
We use density estimation methods (e.g. Expectation-Maximization) to obtain the parameters of the distribution. More later when we cover unsupervised learning.

Overcoming the Independence Assumption

- Naïve Bayes assumption of conditional independence too restrictive
  - But it is intractable without some such assumptions
- **Bayesian Belief network (Bayesian net)** describe conditional independence among subsets of variables (attributes): combining prior knowledge about dependencies among variables with observed training data.
- Bayesian Net
  - Node = variables
  - Arc = dependency
  - DAG, with direction on arc representing causality
  - Variable A with parents B1, …., Bn has a conditional probability table P (A | B1, …., Bn)

Bayesian Networks: Example

- Age, Occupation and Income determine if customer will buy this product.
- Given that customer buys product, whether there is interest in insurance is now independent of Age, Occupation, Income.
- P(Age, Occ, Inc, Buy, Ins) = P(Age)P(Occ)P(Inc) P(Buy|Age,Occ,Inc)P(Int|Buy)



How to categorize Naïve Bayes Classifier?

- Inductive vs Transductive Learning
- Online vs Offline Learning
- Generative vs Discriminative Models
- Parametric vs Non-Parametric Models

How to categorize Naïve Bayes Classifier?

- Inductive vs Transductive Learning
- Online vs Offline Learning (depends!)
- Generative vs Discriminative Models
- Parametric vs Non-Parametric Models (depends!)

Readings

- "Introduction to Machine Learning" by Ethem Alpaydin, 2nd edition, Chapters 3 (3.1-3.4), Chapter 4