CAP 5610: Machine Learning

Lecture 3:
Bayes Classifiers

Instructor: Dr. Gita Sukthankar
Email: gitars@eecs.ucf.edu

Reference Materials From: GuoJun Qi, Ray Mooney

Announcement

No office hours for me on Thursday

Neda’s office hours: Tue-Friday 9:30-11:00 at HEC 308

Survey: Machine Learning Interests

- Cybersecurity
- Robotics/computer vision
- Graphics
- Natural language processing
- Evolutionary computing
- Bioinformatics
- Reinforcement learning
- Financial applications
- Data science
- Specialty applications:
  - Traffic/driving
  - IoT
  - Manufacturing
  - Modeling/simulation

Joint Distributions

- Joint distribution over
  - Input vector X = (X_1, X_2)
    - X_1 =B or ¬B (drinking beer or not)
    - X_2 = H or ¬ H (headache or not)
  - Output vector Y = F or ¬F (binary class)

Input vector X

Output Y

¬F	¬B	¬H	0.4
¬F	¬B	H	0.1
¬F	B	¬H	0.17
¬F	B	H	0.2
F	¬B	¬H	0.05
F	¬B	H	0.05
F	B	¬H	0.015
F	B	H	0.015

Prior Distribution

- Prior for positive class

$P(Y=F) = 0.05+0.05+0.015+0.015$

$= 0.13$

- Prior for negative class

$P(Y=\neg F) = 0.4+0.1+0.17+0.2 = 0.87$

¬F	¬B	¬H	0.4
¬F	¬B	H	0.1
¬F	B	¬H	0.17
¬F	B	H	0.2
F	¬B	¬H	0.05
F	¬B	H	0.05
F	B	¬H	0.015
F	B	H	0.015

Marginalize over values of the other variables

Class-conditional Distribution

- By Bayes Rule, $P(X_1, X_2 | Y = F) = \frac{P(X_1, X_2, Y=F)}{P(Y=F)}$
- $P(X_1 = B, X_2 = H | Y = F) = \frac{P(X_1=B, X_2=H, Y=F)}{P(Y=F)} = \frac{0.015}{0.13} = 0.1154$

X_1 \ X_2	H	¬H
B	0.1154	?
¬B	?	?

¬F	¬B	¬H	0.4
¬F	¬B	H	0.1
¬F	B	¬H	0.17
¬F	B	H	0.2
F	¬B	¬H	0.05
F	¬B	H	0.05
F	B	¬H	0.015
F	B	H	0.015

Class-conditional Distribution

- Bayes Rule $P(X_1, X_2|Y = \neg F) = \frac{P(X_1, X_2, Y = \neg F)}{P(Y = \neg F)}$
- $P(X_1 = B, X_2 = H|Y = \neg F) = \frac{P(X_1 = B, X_2 = H, Y = \neg F)}{P(Y = \neg F)} = \frac{0.2}{0.87} = 0.2299$

X_1 \ X_2	H	¬H
B	0.2299	?
¬B	?	?

¬F	¬B	¬H	0.4
¬F	¬B	H	0.1
¬F	B	¬H	0.17
¬F	B	H	0.2
F	¬B	¬H	0.05
F	¬B	H	0.05
F	B	¬H	0.015
F	B	H	0.015

Posterior Distribution

- $P(Y = F|X_1, X_2) = \frac{P(X_1,X_2,Y=F)}{P(X_1,X_2)}$
  - $P(Y = F|X_1 = B, X_2 = H) = \frac{P(X_1=B,X_2=H,Y=F)}{P(X_1=B,X_2=H)} = \frac{0.015}{0.015 + 0.2}$

- $P(Y = \neg F|X_1, X_2) = \frac{P(X_1,X_2,Y=\neg F)}{P(X_1,X_2)}$

$$= \frac{0.2}{0.015 + 0.2}$$

¬F	¬B	¬H	0.4
¬F	¬B	H	0.1
¬F	B	¬H	0.17
¬F	B	H	0.2
F	¬B	¬H	0.05
F	¬B	H	0.05
F	B	¬H	0.015
F	B	H	0.015

Prior, class-conditional, and posterior distribution

- Prior distribution for a class P(Y) –
  - has no input, the fraction of a particular class in a population
    - What’s the fraction of digit 9 among all digits [0-9]?
    - What’s the fraction of people who are infected with Flu?

Class-conditional Distribution

- Given a class, it is the distribution from which we can draw an example for this class.

Digit 9

Flu

X_1 \ X_2	H	¬H
B	?	
¬B		

$P(X_1,X_2|Y= F)$

Posterior Distribution

- Posterior distribution for classes $P(Y|X_1,X_2)$ – given an input X, what’s the likelihood of a particular class?
  - How likely is this image a digit 9?

Digit 9 ?

Decision Theory

- Maximum A Posteriori (MAP) Rule: given an input vector X, making an optimal decision about the class label (i.e., Y) in a certain sense
  - Minimizing the classification error

  Case I: When $P(Y=F|X_1,X_2)>P(Y=\neg F|X_1,X_2)$, X shall belong to F (i.e., X is infected with Flu)

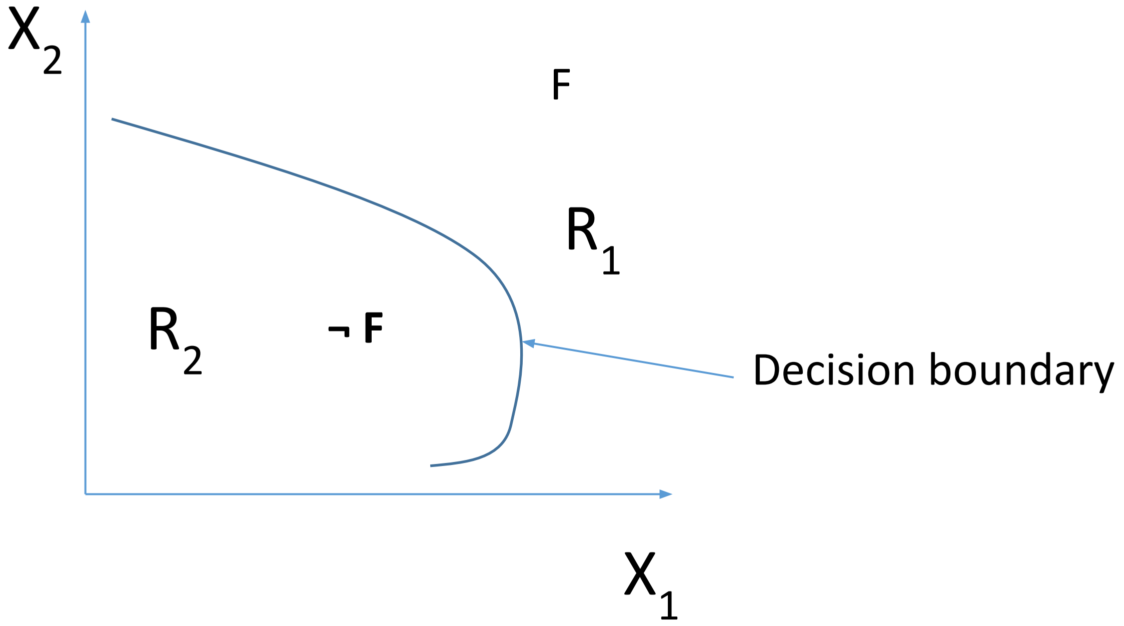
  Case II: When $P(Y=F|X_1,X_2)<P(Y=\neg F|X_1,X_2)$, X shall belong to ¬ F (i.e., X is not infected with Flu)

Proof: MAP rule gives the minimal classification error.

Note: the MAP decision rule is different than the MAP estimation process that we spoke about last week!

Proof

- The decision region defines a region in the feature space such that every point in this region belongs to a particular class.



Error Model

- $p(error) = \int_{R_1} p(X, Y = C_2)\, dX + \int_{R_2} p(X, Y = C_1)\, dX$

$$= \int_{R_1} p(Y = C_2|X)p(X)\, dX + \int_{R_2} p(Y = C_1|X)p(X)\, dX$$

- For each X, it either belongs to R1 or R2; to minimize the error rate, it shall be assigned to the region with a smaller posterior probability.

Intuition: errors occur when instances from class 2 fall in region 1 and class 1 fall in region 2

Likelihood Ratio

- Maximum A Posterior rule

  Case I: When $P(Y=F|X_1,X_2)>P(Y=\neg F|X_1,X_2)$, X shall belong to F (i.e., X is infected with Flu)

  Case II: When $P(Y=F|X_1,X_2)<P(Y=\neg F|X_1,X_2)$, X shall belong to $\neg$ F (i.e., X is not infected with Flu)

- Likelihood Ratios

$$f(X) = \frac{P(Y=F|X_1,X_2)}{P(Y=\neg F|X_1,X_2)}$$

Where f(X) > 1, X belongs to F, otherwise X belongs to $\neg$ F

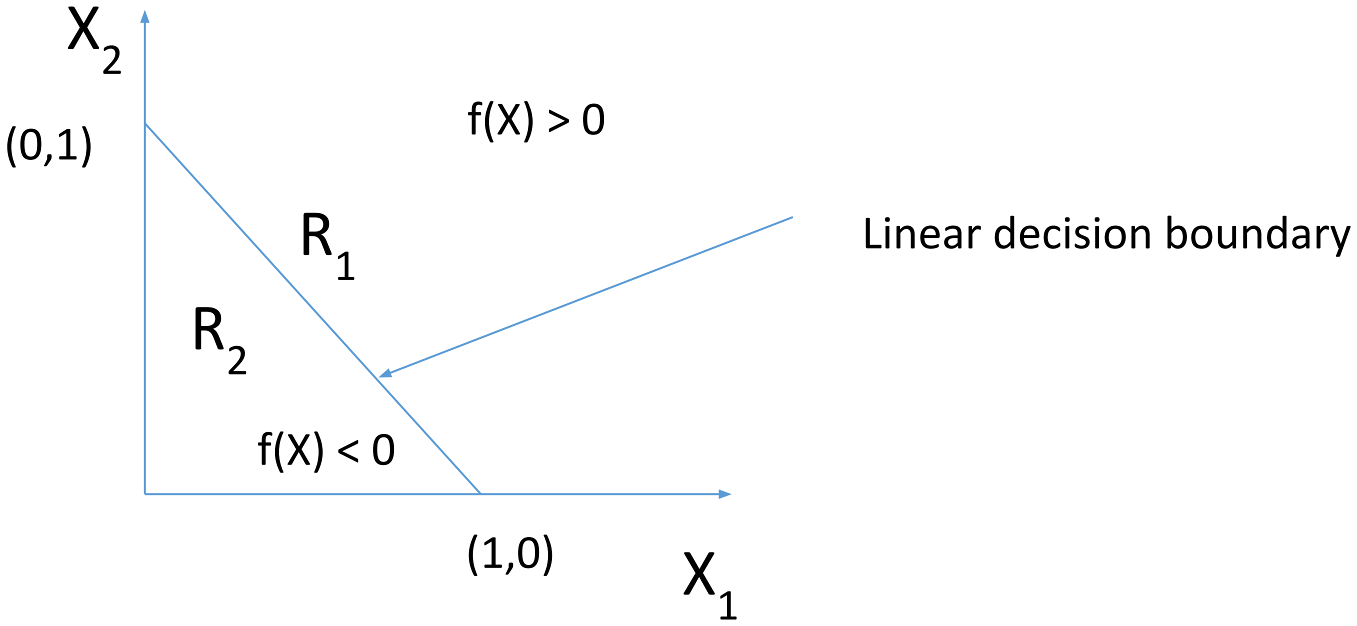
Discriminative Function

- Given an input X, a discriminative function decides its class by comparing f(X) with a certain threshold.

$$f(X)\begin{cases} > 1, X \in F \\ < 1, X \in \neg F \end{cases}$$

Example: Linear Discriminative Function

- $f(X) = X_1 + X_2 - 1$ with threshold 0.



Bayes Error

- Bayes error is the minimal error that is made by MAP rule.
- It is the lowest bound of error rate that can be achieved by any classifier

$$p(\text{error}|X) = \begin{cases} p(Y = C_1|X), if\ P(Y = C_2|X) > P(Y = C_1|X) \\ p(Y = C_2|X), if\ P(Y = C_1|X) > P(Y = C_2|X) \end{cases}$$
$$= \min\{p(Y = C_1|X), p(Y = C_2|X)\}$$

- To minimize errors, choose the least risky class, i.e. the class for which the *expected loss* is smallest
- A conceptual measure for the fundamental hardness of separating y-values given only features x

Nearest Neighbor Error

- The error made by nearest neighbor classifier (1-NN) is bounded by twice the Bayes error.
- Given an example X, its nearest neighbor is X_{NN}; the true class of X is Y, and test the true class of X_{NN} is Y_{NN}.

$$\begin{aligned} p_{NN}(error|X, X_{NN}) &= p(Y = C_1, Y_{NN} = C_2|X, X_{NN}) + p(Y = C_2, Y_{NN} = C_1|X, X_{NN}) \\ &= p(Y = C_1|X)p(Y_{NN} = C_2|X_{NN}) + p(Y = C_2|X)p(Y_{NN} = C_1|X_{NN}) \end{aligned}$$

(independence)

When the size of training set is large enough (approaching infinity), X_{NN} will also approach to X

$$p_{NN}(error|X) = 2p(Y = C_1|X)p(Y = C_2|X)$$

Bayes error and NN asymptotic error

- Bayes error: $p(error|X) = \min\{p(Y = C_1|X), p(Y = C_2|X)\}$
- NN asymptotic error: $p_{NN}(error|X) = 2p(Y = C_1|X)p(Y = C_2|X)$

$$p_{NN}(error|X) < 2p(error|X)$$

Even though nearest neighbor is not an optimal classifier.

Bayesian Classifier

- Comparing the posterior distribution
  - Given an input feature vector *X,*

$$
\begin{array}{l}
C_1, \text{if } P(Y = C_1|X) > P(Y = C_2|X) \\
C_2, \text{if } P(Y = C_2|X) > P(Y = C_1|X)
\end{array}
$$

where $P(Y = C_i|X) \propto P(X|Y = Ci)P(Y = Ci), i = 1,2$

Practical Issues

- Prior distribution $P(Y = C_i), i = 1,2$
  - Counting the fraction of two classes in training set
- Class-conditional distribution $P(X|Y = C_i), i = 1,2$
  - Modeled from the training examples belonging to two classes

Four training examples

X_1(Drinking beer)	X_2(Headache)	Y (Flu)
0	1	1
1	1	1
1	0	0
0	0	0

$$P(X = (0,1)|Y = 1) = \frac{\#(X = (0,1), Y = 1)}{\#(Y = 1)} = \frac{1}{2}$$

$$P(X = (1,1)|Y = 1) = \frac{\#(X = (1,1), Y = 1)}{\#(Y = 1)} = \frac{1}{2}$$

$$P(X = (1,0)|Y = 1) = \frac{\#(X = (1,0), Y = 1)}{\#(Y = 1)} = \frac{0}{2}$$

$$P(X = (0,0)|Y = 1) = \frac{\#(X = (0,0), Y = 1)}{\#(Y = 1)} = \frac{0}{2}$$

N attributes of feature vector

- Input vector: $\boldsymbol{X}=(X_1,X_2,\ldots,X_N)$
- Estimate $P(\boldsymbol{X}|Y=C_i)$, how many examples suffice to do estimation ?
  - Assume X is binary vector, then at least 2^N examples are required to ensure that each possible assignment of binary attributes to **X** has one training example.
  - N=20, 2^N = 1,048,576
  - N=30, 2^N = 1,073,741,824
  - In MNIST, N = 28X28 (pixel), 2^N = $1.01X10^{236,}$

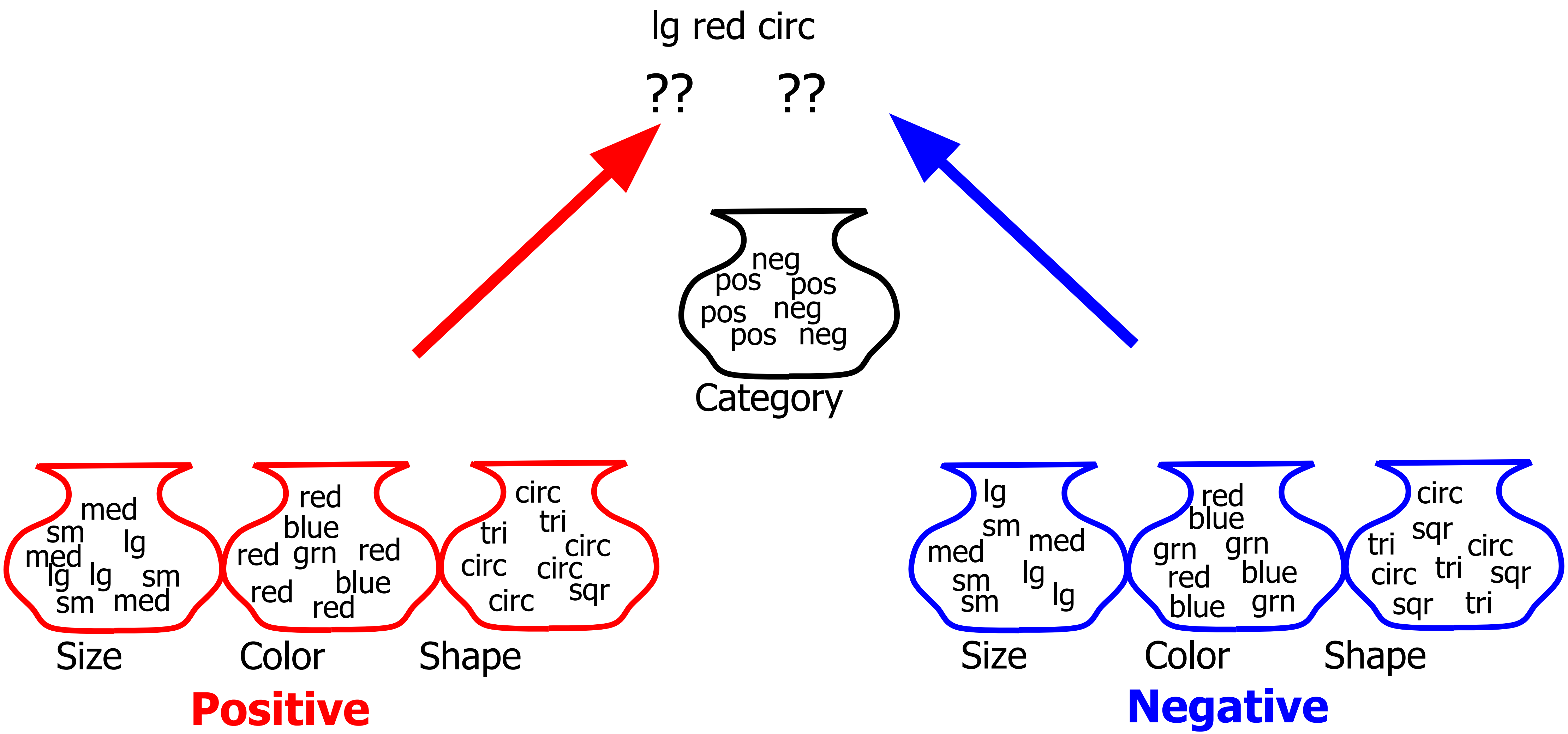
Naive Bayesian Classifier

- If we assume features of an instance are independent **given the category** (*conditionally independent*).

- Therefore, we then only need to know $P(X_i \mid Y)$ for each possible pair of a feature-value and a category.
- If Y and all X_i are binary, this requires specifying only $2n$ parameters:
  - $P(X_i\text{=true} \mid Y\text{=true})$ and $P(X_i\text{=true} \mid Y\text{=false})$ for each X_i
  - $P(X_i\text{=false} \mid Y) = 1 - P(X_i\text{=true} \mid Y)$

- Compared to specifying 2^n parameters without any independence assumptions.

Naive Bayes Inference Problem



Naive Bayes Example

Probability	positive	negative
P(*Y*)	0.5	0.5
P(small | *Y*)	0.4	0.4
P(medium | *Y*)	0.1	0.2
P(large | *Y*)	0.5	0.4
P(red | *Y*)	0.9	0.3
P(blue | *Y*)	0.05	0.3
P(green | *Y*)	0.05	0.4
P(square | *Y*)	0.05	0.4
P(triangle | *Y*)	0.05	0.3
P(circle | *Y*)	0.9	0.3

We learn these probabilities from the training data.

Test Instance:
<medium ,red, circle>

Naive Bayes Example

Probability	positive	negative
P(*Y*)	0.5	0.5
P(medium | *Y*)	0.1	0.2
P(red | *Y*)	0.9	0.3
P(circle | *Y*)	0.9	0.3

Test Instance:
<medium ,red, circle>

Answer:
Drawn from the positive urn

P(positive | *X*) = P(positive)*P(medium | positive)*P(red | positive)*P(circle | positive) / P(*X*)
0.5 * 0.1 * 0.9 * 0.9
= 0.0405 / P(*X*) = 0.0405 / 0.0495 = 0.8181

P(negative | *X*) = P(negative)*P(medium | negative)*P(red | negative)*P(circle | negative) / P(*X*)
0.5 * 0.2 * 0.3 * 0.3
= 0.009 / P(*X*) = 0.009 / 0.0495 = 0.1818

P(positive | *X*) + P(negative | *X*) = 0.0405 / P(*X*) + 0.009 / P(*X*) = 1

P(*X*) = (0.0405 + 0.009) = 0.0495

For purposes of making a decision, we can ignore the denominator since it is the same for both classes.

Classification Methodologies

- There are three methodologies:

  a) Model a classification rule directly

  Examples: k-NN, linear classifier, SVM, neural nets, …

  b) Model the probability of class memberships given input data

  Examples: logistic regression, probabilistic neural nets (softmax),…

  c) Make a probabilistic model of data within each class

  Examples: naive Bayes, model-based ….

- Important ML taxonomy for learning models

  probabilistic models vs non-probabilistic models

  discriminative models vs generative models

Background

- Based on the taxonomy, we can see the essence of different supervised learning models (classifiers) more clearly.

	Probabilistic	Non-Probabilistic
Discriminative	• Logistic Regression • Probabilistic neural nets • ……..	• K-nn • Linear classifier • SVM • Neural networks • ……
Generative	• Naïve Bayes • Model-based (e.g., GMM) • ……	N.A. (?)

Summary

- Recap prior distribution, class-conditional distribution, posterior distribution
- Maximum A Posteriori (MAP) Rule to decide the class assigned to each input vector X
- Likelihood Ratio and discriminant function
- Decision Boundary and Region
- Practical Issues:
  - Estimate prior distribution and class-conditional distribution from training example
  - Naive Bayes
- Discriminative vs. generative models

References

PRML (Bishop) Section 1.5

ML:PP (Murphy) Section 3.5 and Section 5.7