CSCE 970 Lecture 2:
Bayesian-Based Classifiers

Stephen D. Scott

January 10, 2001

Introduction

- A Bayesian classifier classifies instance in the most probable class
- Given M classes $\omega_1, \ldots, \omega_M$ and feat. vector $\mathbf{x}$, find conditional probabilities

$$P(\omega_i \mid \mathbf{x}) \;\; \forall i = 1, \ldots, M,$$

  called a posteriori (posterior) probabilities, and predict with largest
- Will use training data to estimate probability density function (pdf) that yields $P(\omega_i \mid \mathbf{x})$ and classify to ω_i that maximizes

Bayesian Decision Theory

- Use ω_1 and ω_2 only
- Need a priori (prior) probabilities of classes: $P(\omega_1)$ and $P(\omega_2)$
- Estimate from training data:

  $P(\omega_i) \approx N_i/N, \;\; N_i =$ no. of class $\omega_i, \;\; N = N_1 + N_2$

  (will be accurate for sufficiently large N)
- Also need likelihood of $\mathbf{x}$ given class $= \omega_i$: $p(\mathbf{x} \mid \omega_i)$ (is a pdf if $\mathbf{x} \in \Re^\ell$)
- Now apply Bayes Rule:

$$P(\omega_i \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \omega_i) P(\omega_i)}{p(\mathbf{x})}$$

  and classify to ω_i that maximizes

Bayesian Decision Theory
(Cont'd)

- But $p(\mathbf{x})$ is same for all ω_i, so since we want max:

If $p(\mathbf{x} \mid \omega_1)P(\omega_1) > p(\mathbf{x} \mid \omega_2)P(\omega_2)$, classif. $\mathbf{x}$ as ω_1

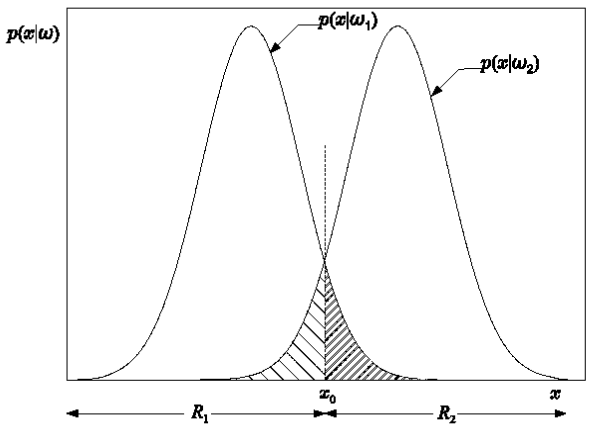
If $p(\mathbf{x} \mid \omega_1)P(\omega_1) < p(\mathbf{x} \mid \omega_2)P(\omega_2)$, classif. $\mathbf{x}$ as ω_2

- If prior probs. equal ($P(\omega_1) = P(\omega_2) = 1/2$) then decide based on:

$$p(\mathbf{x} \mid \omega_1) \gtrless p(\mathbf{x} \mid \omega_2)$$

- Since can estimate $P(\omega_i)$, now only need $p(\mathbf{x} \mid \omega_i)$

Bayesian Decision Theory
Example



- $\ell = 1$ feature, $P(\omega_1) = P(\omega_2)$, so predict at dotted line

- Total error probability = shaded area:

$$P_e = \int_{-\infty}^{x_0} p(x \mid \omega_2)dx + \int_{x_0}^{+\infty} p(x \mid \omega_1)dx$$

Bayesian Decision Theory
Probability of Error

- In general, error is

$$\begin{aligned} P_e &= P(\mathbf{x} \in R_2, \omega_1) + P(\mathbf{x} \in R_1, \omega_2) \\ &= P(\mathbf{x} \in R_2 \mid \omega_1)P(\omega_1) + P(\mathbf{x} \in R_1 \mid \omega_2)P(\omega_2) \\ &= P(\omega_1) \int_{R_2} p(\mathbf{x} \mid \omega_1)d\mathbf{x} + P(\omega_2) \int_{R_1} p(\mathbf{x} \mid \omega_2)d\mathbf{x} \\ &= \int_{R_2} P(\omega_1 \mid \mathbf{x})p(\mathbf{x})d\mathbf{x} + \int_{R_1} P(\omega_2 \mid \mathbf{x})p(\mathbf{x})d\mathbf{x} \end{aligned}$$

- Since R_1 and R_2 cover entire space,

$$\int_{R_1} P(\omega_1 \mid \mathbf{x})p(\mathbf{x})d\mathbf{x} + \int_{R_2} P(\omega_1 \mid \mathbf{x})p(\mathbf{x})d\mathbf{x} = \boxed{P(\omega_1)}$$

- Thus

$$P_e = \boxed{P(\omega_1)} - \int_{R_1} \left(P(\omega_1 \mid \mathbf{x}) - P(\omega_2 \mid \mathbf{x})\right) p(\mathbf{x})d\mathbf{x},$$

which is minimized if

$$R_1 = \left\{\mathbf{x} \in \Re^\ell : P(\omega_1 \mid \mathbf{x}) > P(\omega_2 \mid \mathbf{x})\right\},$$

which is what the Bayesian classifier does!

Bayesian Decision Theory
$\ell > 2$

- If number of classes $\ell > 2$, then classify according to

$$\operatorname*{argmax}_{\omega_i} P(\omega_i \mid \mathbf{x})$$

- Proof of optimality still holds

Bayesian Decision Theory
Minimizing Risk

- What if different errors have different penalties, e.g. cancer diagnosis?
  - False negative worse than false positive

- Define λ_{ki} as loss (penalty, risk) if we predict ω_i when correct answer is ω_k (forms $L =$ loss matrix)

- Can minimize average loss:

$$\begin{aligned} r &= \sum_{k=1}^{M} P(\omega_k) \sum_{i=1}^{M} \lambda_{ki} \overbrace{\int_{R_i} p(\mathbf{x} \mid \omega_k)d\mathbf{x}}^{\text{prob. of error } ki} \\ &= \sum_{i=1}^{M} \int_{R_i} \left(\sum_{k=1}^{M} \lambda_{ki}\, p(\mathbf{x} \mid \omega_k)P(\omega_k) \right) d\mathbf{x} \end{aligned}$$

by minimizing each integral:

$$R_i = \left\{ \mathbf{x} \in \Re^\ell : \sum_{k=1}^{M} \lambda_{ki}\, p(\mathbf{x} \mid \omega_k)P(\omega_k) < \sum_{k=1}^{M} \lambda_{kj}\, p(\mathbf{x} \mid \omega_k)P(\omega_k) \ \forall j \neq i \right\}$$

Bayesian Decision Theory

Minimizing Risk

Example

- Let $\ell = 2$, $P(\omega_1) = P(\omega_2) = 1/2$, $L = \begin{pmatrix} 0 & \lambda_{12} \\ \lambda_{21} & 0 \end{pmatrix}$, and $\lambda_{21} > \lambda_{12}$
- Then

$$R_2 = \left\{\mathbf{x} \in \Re^2 : \lambda_{21}\, p(\mathbf{x} \mid \omega_2) > \lambda_{12}\, p(\mathbf{x} \mid \omega_1)\right\}$$
$$= \left\{\mathbf{x} \in \Re^2 : p(\mathbf{x} \mid \omega_2) > p(\mathbf{x} \mid \omega_1)\frac{\lambda_{12}}{\lambda_{21}}\right\},$$

which slides threshold left of x_0 on slide 5 since $\lambda_{12}/\lambda_{21} < 1$

Discriminant Functions

- Rather than using probabilities (or risk functions) directly, sometimes easier to work with a function of them, e.g.

$$g_i(\mathbf{x}) = f(P(\omega_i \mid \mathbf{x}))$$

$f(\cdot)$ is monotonically increasing function, $g_i(\mathbf{x})$ is called discriminant function

- Then $R_i = \left\{\mathbf{x} \in \Re^\ell : g_i(\mathbf{x}) > g_j(\mathbf{x}) \;\; \forall j \neq i\right\}$
- Common choice of $f(\cdot)$ is natural logarithm (multiplications become sums)
- Still requires good estimate of pdf
  - Will look at a tractable case next
  - In general, cannot necessarily easily estimate pdf, so use other cost functions (Chapters 3 & 4)

Normal Distributions

- Assume the pdf of likelihood functions follow a normal (Gaussian) distribution for $1 \leq i \leq M$:

$$p(\mathbf{x} \mid \omega_i) = \frac{1}{(2\pi)^{\ell/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \Sigma_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right)$$

  · $\boldsymbol{\mu}_i = E[\mathbf{x}]$ = mean value of ω_i class

  · $|\Sigma_i|$ = determinant of Σ_i, ω_i's covariance matrix:

$$\Sigma_i = E\left[(\mathbf{x}-\boldsymbol{\mu}_i)(\mathbf{x}-\boldsymbol{\mu}_i)^T\right]$$

  - Assume we know $\boldsymbol{\mu}_i$ and Σ_i $\forall i$
- Using the following discriminant function:

$$g_i(\mathbf{x}) = \ln(p(\mathbf{x} \mid \omega_i)P(\omega_i))$$

we get:

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \Sigma_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) + \ln(P(\omega_i)) - \ell/2 \ln(2\pi) - (1/2)\ln|\Sigma_i|$$

Normal Distributions

Minimum Distance Classifiers

- If $P(\omega_i)$'s equal and Σ_i's equal, can use:

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)$$

- If features statistically independent with same variance, then $\Sigma = \sigma^2 I$ and can instead use

$$g_i(\mathbf{x}) = -\frac{1}{2}\sum_{j=1}^{\ell}(x_j - \mu_{ij})^2$$

- Finding ω_i maximizing this implies finding $\boldsymbol{\mu}_i$ that minimizes Euclidian distance to $\mathbf{x}$
  - Constant distance = circle centered at $\boldsymbol{\mu}_i$
- If Σ not diagonal, then maximizing $g_i(\mathbf{x})$ is same as minimizing Mahalanobis distance:

$$\sqrt{(\mathbf{x}-\boldsymbol{\mu}_i)^T \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)}$$

  - Constant distance = ellipse centered at $\boldsymbol{\mu}_i$

Estimating Unknown pdf's
Maximum Likelihood Parameter Estimation

- If we know cov. matrix but not mean for a class ω, can parameterize ω's pdf on mean $\boldsymbol{\mu}$:

$$p(\mathbf{x}_k; \boldsymbol{\mu}) = \frac{1}{(2\pi)^{\ell/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_k - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}_k - \boldsymbol{\mu})\right)$$

and use data $\mathbf{x}_1, \ldots, \mathbf{x}_N$ from ω to estimate $\boldsymbol{\mu}$

- The maximum likelihood (ML) method estimates $\boldsymbol{\mu}$ such that the following likelihood function is maximized:

$$p(X; \boldsymbol{\mu}) = p(\mathbf{x}_1, \ldots, \mathbf{x}_N; \boldsymbol{\mu}) = \prod_{k=1}^{N} p(\mathbf{x}_k; \boldsymbol{\mu})$$

- Taking logarithm and setting gradient $= \mathbf{0}$:

$$\frac{\partial}{\partial \boldsymbol{\mu}} \underbrace{\left(-\frac{N}{2} \ln\left((2\pi)^{\ell}|\Sigma|\right) - \frac{1}{2}\sum_{k=1}^{N}(\mathbf{x}_k - \boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}_k - \boldsymbol{\mu})\right)}_{L} = \mathbf{0}$$

Estimating Unknown pdf's
ML Param Est (cont'd)

- Assuming statistical indep. of x_{ki}'s, $\Sigma_{ij}^{-1} = 0$ for $i \neq j$, so

$$\frac{\partial L}{\partial \boldsymbol{\mu}} = \begin{bmatrix} \frac{\partial L}{\partial \mu_1} \\ \vdots \\ \frac{\partial L}{\partial \mu_\ell} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial \mu_1}\left(-\frac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{\ell}\left(x_{kj} - \mu_j\right)^2 \Sigma_{jj}^{-1}\right) \\ \vdots \\ \frac{\partial}{\partial \mu_\ell}\left(-\frac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{\ell}\left(x_{kj} - \mu_j\right)^2 \Sigma_{jj}^{-1}\right) \end{bmatrix}$$

$$= \sum_{k=1}^{N} \Sigma^{-1}(\mathbf{x}_k - \boldsymbol{\mu}) = 0,$$

yielding

$$\hat{\boldsymbol{\mu}}_{ML} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_k$$

- Solve above for each class independently

- Can generalize technique for other distributions and parameters

- Has many nice properties (p. 30) as $N \to \infty$

Estimating Unknown pdf's
Maximum A Posteriori Parameter Estimation

- If $\boldsymbol{\mu}$ is norm. distrib., $\Sigma = \sigma_\mu^2 I$, mean $= \boldsymbol{\mu}_0$:

$$p(\boldsymbol{\mu}) = \frac{1}{(2\pi)^{\ell/2}\,\sigma_\mu^\ell} \exp\left(-\frac{(\boldsymbol{\mu} - \boldsymbol{\mu}_0)^T(\boldsymbol{\mu} - \boldsymbol{\mu}_0)}{2\sigma_\mu^2}\right)$$

- Maximizing $p(\boldsymbol{\mu} \mid X)$ is same as maximizing

$$p(\boldsymbol{\mu})p(X \mid \boldsymbol{\mu}) = \prod_{k=1}^{N} p(\mathbf{x}_k \mid \boldsymbol{\mu})p(\boldsymbol{\mu})$$

- Again, take log and set gradient $= 0$: $\boxed{(\Sigma = \sigma^2 I)}$

$$\sum_{k=1}^{N} \frac{1}{\sigma^2}(\mathbf{x}_k - \boldsymbol{\mu}) - \frac{1}{\sigma_\mu^2}(\boldsymbol{\mu} - \boldsymbol{\mu}_0) = \mathbf{0}$$

so

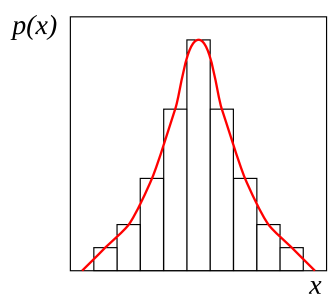
$$\hat{\boldsymbol{\mu}}_{MAP} = \frac{\boldsymbol{\mu}_0 + (\sigma_\mu^2/\sigma^2)\sum_{k=1}^{N}\mathbf{x}_k}{1 + (\sigma_\mu^2/\sigma^2)N}$$

- $\boldsymbol{\mu}_{MAP} \approx \boldsymbol{\mu}_{ML}$ if $p(\boldsymbol{\mu})$ almost uniform or $N \to \infty$

- Again, can generalize technique

Estimating Unknown pdf's
(Nonparametric Approach)
Parzen Windows

- Historgram-based technique to approximate pdf: Partition space into "bins" and count number of training vectors per bin



- Let $\phi(\mathbf{x}) = \begin{cases} 1 & \text{if } |x_j| \leq 1/2 \\ 0 & \text{otherwise} \end{cases}$

- Now approximate pdf $p(\mathbf{x})$ with

$$\hat{p}(\mathbf{x}) = \frac{1}{h^\ell}\left(\frac{1}{N}\sum_{i=1}^{N} \phi\left(\frac{\mathbf{x}_i - \mathbf{x}}{h}\right)\right)$$

Estimating Unknown pdf's
Parzen Windows
(cont'd)

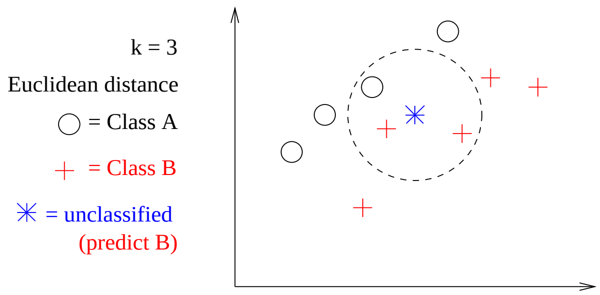
$$\hat{p}(\mathbf{x}) = \frac{1}{h^\ell}\left(\frac{1}{N}\sum_{i=1}^{N}\phi\left(\frac{\mathbf{x}_i - \mathbf{x}}{h}\right)\right)$$

- I.e. given $\mathbf{x}$, to compute $\hat{p}(\mathbf{x})$:
  - Count number of training vectors in size-h (per side) hypercube H centered at $\mathbf{x}$
  - Divide by N to est. probability of getting a point in H
  - Divide by volume of H
- Problem: Approximating continuous function $p(\mathbf{x})$ with discontinuous $\hat{p}(\mathbf{x})$
- Solution: Substitute a smooth function for $\phi(\cdot)$, e.g. $\phi(\mathbf{x}) = \left(1/(2\pi)^{\ell/2}\right)\exp\left(-\mathbf{x}^T\mathbf{x}/2\right)$

Estimating Unknown pdf's
Parzen Windows
Numeric Example

k-Nearest Neighbor Techniques

- Classify unlabeled feature vector $\mathbf{x}$ according to a majority vote of its k nearest neighbors



- As $N \rightarrow \infty$,
  - 1-NN error is at most twice Bayes opt. (P_B)
  - k-NN error is $\leq P_B + 1/\sqrt{ke}$
- Can also weight votes by relative distance
- Complexity issues: Research into more efficient algorithms, approximation algorithms