In the name of God, the Compassionate, the Merciful

STATISTICAL PATTERN RECOGNITION

1- Introduction

- (??) **Pattern recognition** is the act of taking in raw data and taking an action based on the category of the pattern.
- (??) **Pattern classification** is to take in raw data, eliminate noise, and process it to select the most likely model that it represents.

Pattern Recognition Approaches

- **Statistical**: Focus on statistics of the patterns.
- **Structural (Syntactic)**: Classifiers are defined using a set of logical rules. Grammars can group rules.

Bias-Variance Dilemma

- **Variance**: Simple decision boundaries (e.g., linear) seem to miss some obvious trends in the data.
- **Bias**: Complex decision boundaries seem to lock onto the idiosyncracies of the training data set.
- **Generalization** is the best trade-off.

The Sub-problems of Pattern Classification:

1-Feature Extraction, 2-Noise, 3-Overfitting, 4-Model Selection, 5-Prior Knowledge, 6-Missing Features, 7-Mereology (the problem of *subsets and supersets*), 8-Segmentation(e.g., in speech recognition), 9-Context(input-dependent information), 10-Invariances (e.g., to translation), 11-Evidence Pooling(e.g., voting), 12-Costs and Risks, 13-Computational Complexity

Learning

Types of Learning

- **Unsupervised learning:** The system forms clusters or "natural groupings" of the input patterns.
- **Supervised learning:** Classify data using labeled samples.
- **Semi-supervised learning:** make use of both labeled and unlabeled data for training - typically a small amount of labeled data with a large amount of unlabeled data.
- **Reinforcement Learning:** only teaching feedback is available.

Types of Learning (Algorithmic view point)

- **Inductive:** Learns a labeling function over the space
- **Transductive:** Just labels the given test queries

Version Space: Any $h \in H$ between the most specific consistent hypotheses, and the most general consistent hypotheses with training set.

VC Dimension

- If a set of examples can be partitioned in all possible ways, we say the hypothesis space H **shatters** that set of examples.
- If we have a set of N examples, we need all possible 2^N hypotheses to shatter the set of examples.
- VC dimension is the size of the largest finite subset of examples in the input space X shattered by H.
- VC dimension can be infinite.
- VC dimension of the set of classification functions (H) is: maximum number of training examples (N) that can be shattered by H or in other word can learned by the machine without error for all possible labeling of the classification functions.
- number of points needed to learn a class of interest reliably is proportional to the VC dimension.
- In general, the VC dimension of the space of hyperplanes in r dimensions is $r + 1$.
- Suppose that VC(H) = d. Therefore $2d \leq |H|$ and $d = \text{VC}(H) \leq log_2|H|$.
- For a linear classifier VC=d+1 ???????

Correctness Criteria:

Precision	$\frac{TP}{TP+FP}$
Racal (Hit rate)	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Balanced accuracy	$\frac{Racal+Specificity}{2}$
F-Measure	$2.\frac{Precision.Racal}{Precision+Racal}$

2- Bayesian decision theory

Bayes' formula:

$P(\omega_j|x) = \frac{p(x|\omega_j)P(\omega_j)}{p(x)}$ $\qquad posterior = \frac{likelihood \times prior}{evidence}$

Bayes' decision rule:

$$P(\omega_1|x) \lessgtr_{\omega_1}^{\omega_2} P(\omega_2|x) \qquad \omega^* = \arg\max_i P(\omega_i|x)$$

- Under this rule $P(error|x) = \min[P(\omega_1|x), P(\omega_2|x)]$.
- By eliminating this scale factor $p(x)$:

$$P(x|\omega_1)P(\omega_1) \lessgtr_{\omega_1}^{\omega_2} P(\omega_2|x)P(\omega_2)$$

Bayesian Decision Theory

Loss Function

- The loss function states exactly how costly each action is, and is used to convert a probability determination into a decision.
- The loss function $\lambda_{ij} = \lambda(\alpha_i|\omega_j)$ describes the loss incurred for taking action α_i when the state of nature (class) is ω_j.

Risk

- An expected loss is called a risk.
$$R(\alpha_i|\text{x}) = \sum_{j=1}^{c} \lambda(\alpha_i|\omega_j)P(\omega_j|\text{x}).$$
- $R(\alpha_i|\text{x})$ is the conditional risk associated with action α_i.
- The overall risk is the expected loss associated with a given decision rule. $R = \int R(\alpha(\text{x})|\text{x})p(\text{x})d\text{x}$
- minimum-risk decision rule: ω_1 if $R(\alpha_1|\text{x}) < R(\alpha_2|\text{x})$ where

$$R(\alpha_1|\text{x}) = \lambda_{11}P(\omega_1|\text{x}) + \lambda_{12}P(\omega_2|\text{x})$$
$$R(\alpha_2|\text{x}) = \lambda_{21}P(\omega_1|\text{x}) + \lambda_{22}P(\omega_2|\text{x})$$

SO:
$$(\lambda_{21} - \lambda_{11})P(\omega_1|\text{x}) > (\lambda_{12} - \lambda_{22})P(\omega_2|\text{x})$$
$$(\lambda_{21} - \lambda_{11})p(\text{x}|\omega_1)P(\omega_1) > (\lambda_{12} - \lambda_{22})p(\text{x}|\omega_2)P(\omega_2)$$

likelihood ratio $\frac{p(\text{x}|\omega_1)}{p(\text{x}|\omega_2)} > \frac{\lambda_{12}-\lambda_{22}}{\lambda_{21}-\lambda_{11}} \frac{P(\omega_2)}{P(\omega_1)}$

Minimum-Error-Rate Classification

- *symmetrical* or *zero-one* loss function
$$\lambda_{ij} = \begin{cases} 0 & i = j \\ 1 & i \neq j \end{cases}$$
$R(\alpha_i|\text{x}) = 1 - P(\omega_i|\text{x})$

Minimax Criterion

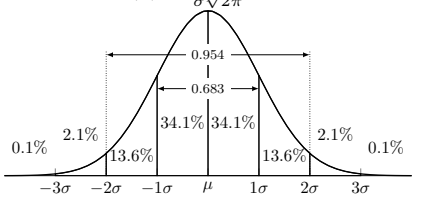
- minimize the maximum possible overall risk.
- minimax risk:
$$R_{mm} = \lambda_{22} + (\lambda_{12} - \lambda_{22}) \int_{\mathcal{R}_1} p(\text{x}|\omega_2)d\text{x}$$
$$= \lambda_{11} + (\lambda_{21} - \lambda_{11}) \int_{\mathcal{R}_2} p(\text{x}|\omega_1)d\text{x}$$

Neyman-Pearson Criterion: minimize the overall risk subject to a constraint.

2.4 Classifiers, Discriminant Functions????????????

The Normal Density

Univariate Density: $p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$



Entropy

- The entropy is a non-negative quantity that describes the fundamental uncertainty in the values of points selected randomly from a distribution.
$$H(p(x)) = -\int p(x) \ln p(x) dx$$
- measured in *nats*. If a $\log_2$ is used instead, the unit is the *bit*.
- The uniform distribution has maximum entropy (on a given interval).
- Normal distribution has the maximum entropy of all distributions having a given mean and variance
- **Central Limit Theorem**: The aggregate effect of a large number of small, independent random disturbances will lead to a Gaussian distribution

Multivariate Density:

$$N(\mu, \Sigma) = P(\text{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(\text{x}-\mu)^t \Sigma^{-1}(\text{x}-\mu)}$$

- For $p(\text{x}) \sim N(\mu, \Sigma)$ and $A_{d\times k}$, define $\text{y} = A^t x$ then:
$$p(\text{y}) \sim N(A^t\mu, A^t\Sigma A)$$
- **Whitening transformation**: If we define Φ to be the matrix whose columns are the orthonormal eigenvectors of Σ, and Λ the diagonal matrix of the corresponding eigenvalues, then the transformation $A_w = \Phi\Lambda^{-1/2}$ applied to the coordinates insures that the transformed distribution has covariance matrix equal to the identity matrix. The transform A_w is called a *whitening transformation.*
- **Mahalanobis distance:** $r^2 = (\text{x} - \mu)^t\Sigma^{-1}(\text{x} - \mu)$

Missing Features:

- let $\text{x} = [\text{x}_g, \text{x}_b]$, where x_g represents the known or "good" features and x_b represents the "bad" ones.
- $P(\omega_j|\text{x}_g) = \frac{\int P(\omega_i|\text{x}_g,\text{x}_b)d\text{x}_b}{p(\text{x}_g)} = \frac{\int P(\omega_i|\text{x}_g,\text{x}_b)p(\text{x}_b,\text{x}_g)d\text{x}_b}{p(\text{x}_g)}$
- we must integrate (marginalize) the posterior probability over the bad features. Finally we use the Bayes decision rule on the resulting posterior probabilities, i.e., choose ω_i if $P(\omega_i|\text{x}_g) > P(\omega_j|\text{x}_g)$ for all i and j.

3-ML and Bayesian parameter estimation

parameter estimation

- **Maximum Likelihood (ML):** Maximum likelihood and several other methods view the parameters as quantities whose values are fixed but unknown. The best estimate of their value is defined to be the one that maximizes the probability of obtaining the samples actually observed.
- **Bayesian:** Bayesian methods view the parameters as random variables having some known a priori distribution.

Maximum Likelihood Estimation

- Nearly always have good convergence properties as the number of training samples increases.
- Maximum likelihood estimation often can be simpler than alternate methods, such as Bayesian techniques.
- Our problem is to use the information provided by the training samples to obtain good estimates for the unknown parameter vectors $\theta_1, ..., \theta_c$ associated with each category.
- $\mathcal{D}$ is the set of **i.i.d** training samples and θ is unknown parameters. So, $p(\mathcal{D}|\theta) = \prod_{k=1}^{n} p(x_k|\theta)$.

log-likelihood

- We define $l(\theta)$ as the log-likelihood function:
$$l(\theta) \equiv \ln p(\mathcal{D}|\theta) = \sum_{k=1}^{n} \ln p(\mathrm{x}_k|\theta)$$
- $\hat{\theta}$ is estimated parameters and $\hat{\theta} = \theta$ when $n \to \infty$
$$\hat{\theta} = \arg\max_{\theta} l(\theta)$$
- Let Δ_θ be the gradient operator, then
$$\Delta_\theta l(\theta) = \sum_{k=1}^{n} \Delta_\theta \ln p(x_k|\theta)$$
- Thus, a set of necessary conditions for the ML estimate for θ can be obtained from the set of p equations:
$$\Delta_\theta l = 0$$

Gaussian Case

- Sample mean ($\hat{\mu}_n$): $\frac{1}{n}\sum_{k=1}^{n} \mathrm{x}_k$
- Sample variance ($\widehat{\Sigma}_n$): $\frac{1}{n}\sum_{k=1}^{n}(\mathrm{x}_k - \hat{\mu})(\mathrm{x}_k - \hat{\mu})^t$
  - The maximum likelihood estimate for the variance is biased.

Bayesian estimation

- In Bayesian learning we consider θ to be a random variable, and training data allows us to convert a distribution on this variable into a posterior probability density.
- The primary task in Bayesian Parameter Estimation is the computation of the posterior density $p(\theta|D)$ by Bayes formula.

The Class-Conditional Densities

- Given the sample $\mathcal{D}$, Bayes formula then becomes:
$$P(\omega_i|\mathrm{x}, D) = \frac{p(\mathrm{x}|\omega_i, \mathcal{D})P(\omega_i|\mathcal{D})}{\sum_{j=1}^{c} p(\mathrm{x}|\omega_j, \mathcal{D})P(\omega_j|\mathcal{D})}$$
- with the samples in $\mathcal{D}_i$ belonging to ω_i:
$$P(\omega_i|\mathrm{x}, D) = \frac{p(\mathrm{x}|\omega_i, \mathcal{D}_i)P(\omega_i)}{\sum_{j=1}^{c} p(\mathrm{x}|\omega_j, \mathcal{D}_j)P(\omega_j)}$$

Parameter Distribution:
$$p(\mathrm{x}|\mathcal{D}) = \int p(\mathrm{x}, \theta|\mathcal{D})d\theta = \int p(\mathrm{x}|\theta)p(\theta|\mathcal{D})d\theta$$

Recursive Bayesian Estimation

??? –

When do ML and Bayes methods differ?

- ML and Bayes solutions are equivalent in the asymptotic limit of infinite training data.
- ML methods are often to be prefered computationally.
- ML: require merely differential calculus techniques or gradient search for $\hat{\theta}$
- Baysian: need complex multidimensional integration.
- ML: easier to interpret and understand.
- Bayesian methods use more of the information brought to the problem than do ML methods. If such information is reliable, Bayes methods can be expected to give better results.

4-Nonparametric Methods

- Most real-world entities have multimodal distributions whereas all classical parametric densities are unimodal.
- **nonparametric**: arbitrary distributions and without the assumption that the underlying form of the densities are known. e.g., Histograms, Kernel Density Estimation/Parzen Windows, k-Nearest Neighbor Density Estimation.

Histogram Density Estimator

- **Advantages:**
  - Simple to evaluate and simple to use.
  - One can throw away $\mathcal{D}$ once the histogram is computed.
  - Can be computed sequentially if data continues to come in.
- **Disadvantages:**
  - The estimated density has discontinuities due to the bin edges rather than any property of the underlying density.
  - Scales poorly (curse of dimensionality): we would have M^D bins if we divided each variable in a D-dimensional space into M bins.

Density estimation

- Let $\mathcal{R}$ denote a small region containing x.
- The probability P that a vector x will fall in a region $\mathcal{R}$ is given by $P = \int_{\mathcal{R}} p(\mathrm{x}')d\mathrm{x}'$
- The probability that k of these n samples fall in $\mathcal{R}$ is given by the binomial law $P_k = \binom{n}{k} p^k(1-p)^{n-k}$
- and the expected value for k is $E[K] = nP$
- Assuming continuous $p(x)$ and that $\mathcal{R}$ is so small that $p(x)$ does not appreciably vary within it, we can write: $\int_{\mathcal{R}} p(x')dx' \simeq p(x)V$ where x is a point within $\mathcal{R}$ and V is the volume enclosed by $\mathcal{R}$. SO: $p(x) \simeq \frac{k}{nV}$
- SO what ... ????
- ...
- ...
- There are two common ways of obtaining regions that satisfy these conditions:
  - Shrink an initial region by specifying the volume V_n as some function of n such as $V_n = 1/\sqrt{n}$. Then, we need to show that $p_n(\mathrm{x})$ converges to $p(\mathrm{x})$. (e.g, Parzen window)
  - Specify k_n as some function of n such as $V_n = 1/\sqrt{n}$. Then, we grow the volume V_n until it encloses k_n neighbors of x. (e.g., k-nearest-neighbor).

Parzen windows

- assuming that the region $\mathcal{R}_n$ is a d-dimensional hypercube.
- k_n: The number of samples falling in the hypercube, by defining window function $\varphi(u)$

k_n Nearest Neighbor Methods

- Selecting the best window/bandwidth is a severe limiting factor for Parzen window estimators.
- k_n-NN methods circumvent this problem by making the window size a function of the actual training data.
- The basic idea here is to center our window around x and let it grow until it capture k_n samples

5-Linear Discriminant Functions

Linear Discriminant Function

The Two-Category Case:

- Discriminant function: $g(\mathrm{x}) = \mathrm{w}^t\mathrm{x} + w_0$
  where w is the weight vector and w_0 the bias or threshold weight.
- The distance from x to the hyperplane.
$$\mathrm{x} = \mathrm{x}_p + r\frac{\mathrm{w}}{\|\mathrm{w}\|}; \quad r = \frac{g(\mathrm{x})}{\|w\|}$$
  where x_p is the normal projection of x onto H, and r is the desired algebraic distance. positive if x is on the positive side and negative if x is on the negative side.

The Multi-Category Case:

- reduce the problem to $c-1$ two-class problems
- use $c(c-1)/2$ linear discriminants, one for everypair of classes.

Generalized Linear Discriminant Functions

- Linear discriminant function: $g(x) = w_0 + \sum_{i=1}^{d} w_i x_i$
- Quadratic discriminant function:
$$g(x) = w_0 + \sum_{i=1}^{d} w_i x_i + \sum_{i=1}^{d}\sum_{j=1}^{d} w_{ij} x_i x_j$$
- Generalized linear discriminant function:
$$g(x) = \sum_{i=1}^{\hat{d}} a_i y_i(x) = \mathrm{a}^t\mathrm{y}$$
  y augmented feature vector, **a** augmented weight vector
- The $\hat{d}$ functions $y_i(x)$ merely map points in d-dimenional x-space to points in $\hat{d}$-dimensional y-space.
- The resulting discriminant function is not linear in x, but it is linear in y.
- The hyperplane decision surface $\hat{H}$ defined by $\mathrm{a}^t\mathrm{y} = 0$

The Two-Category Linearly-Separable Case

- if exists a weight vector that classifies all of the samples correctly, the samples are said to be **linearly separable**.

References:

[1] Pattern Classification, by Richard O. Duda, David G. Stork, Peter E. Hart

Made by ma.mehralian using LaTeX