

CS4770

Pattern Recognition

Parameter Estimation

P. J. Narayanan
Monsoon 2006

Parametric Form

- We assume the probability distribution $p(x|\omega)$ and $P(\omega)$ are known.
- Priors are possible to know. Likelihoods are harder.
- We can assume the **functional form** of the probabilities are known. For example, it is a Gaussian, or binomial, etc.
- Gaussian is given by $N(\mu, \Sigma)$. $d + d(d + 1)/2$ unknowns define the probability density, after the form is known. Poisson distribution needs only d unknowns, etc.

- The distribution is fully specified by a set of **parameters** (denoted by the *parameter vector* θ). If we can know θ , we know the probability and can use Bayesian decision theory for classification.
- For Gaussians, estimating $p(\mathbf{x}|\omega)$ is reduced to one of estimating parameters $\theta_1 = \mu$ and $\theta_2 = \Sigma$.

Estimating θ

- Parameter vector is estimated using a number of observations.
- We are given data points belonging to each class ω_i whose feature vectors \mathbf{x} can be measured.
- The parameters can be estimated from a large number of such **training data**.
- We will get an estimated parameter vector $\hat{\theta}$ and not the true parameter vector θ .

Training Data Set

- Given c data sets D_1, D_2, \dots, D_c with samples in D_i drawn independently with probability $p(\mathbf{x}|\omega_i)$.
- $p(\mathbf{x}|\omega_i)$ has a known parametric form, with an unknown parameter vector θ_i .
- Assumption: D_i gives information about only the class ω_i . We can focus on estimating θ for a given D and apply it separately to each class.
- Assumption: The samples in D are random variables as per $p(\mathbf{x}|\omega)$. They have been drawn independently and are **independent and identically distributed (IID)**.

Maximum-Likelihood Estimation

- Estimate the parameter vector $\hat{\theta}$ that maximizes the likelihood $p(D|\theta)$.
- Look for parameters that best explain the training data!
- Since the training data is IID, $p(D|\theta) = \prod_{i=1}^n p(\mathbf{x}_i|\theta)$
- Maximize $p(D|\theta)$ with respect to θ .
- Log-likelihood of $\ln p(D|\theta)$ is easier to handle in practice and has the same maximum point $\hat{\theta}$.

Maximizing Likelihood

- Differentiate with respect to θ and set to zero.
- Parameter vector θ has p components $\theta_1, \dots, \theta_p$.
- Use the gradient operator ∇_{θ} for differentiation.
- $l(\theta) = \ln p(D|\theta) = \sum_i \ln p(\mathbf{x}_i|\theta)$.
Maximum-likelihood estimate is $\hat{\theta} = \arg \max_{\theta} l(\theta)$.
- Set $\nabla_{\theta} l(\theta) = \sum_i \nabla_{\theta} \ln p(\mathbf{x}_i|\theta) = 0$. Solve for $\hat{\theta}$.

Maximum, minimum, etc.

- Setting gradient to 0 gives global or local maximum or minimum.
- Choose the right one by examining them all.
- If priors on the parameter vector $p(\theta)$ are known, we can do a **maximum a posteriori** (or MAP) estimation by maximizing $l(\theta)p(\theta)$.

Example: Gaussian with unknown μ

- θ is the same as μ .
- $\ln p(\mathbf{x}_i|\mu) = -\frac{1}{2} \ln[(2\pi)^d |\boldsymbol{\Sigma}|] - \frac{1}{2}(\mathbf{x}_i - \mu)^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \mu)$
- $\nabla_{\mu} \ln p(\mathbf{x}_i|\mu) = \frac{\partial \ln p(\mathbf{x}_i|\mu)}{\partial \mu} = \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \mu)$.
- $\sum_i \boldsymbol{\Sigma}^{-1}(\mathbf{x}_i - \mu) = 0$. $\sum_i (\mathbf{x}_i - \mu) = 0$.
- Thus, $\mu = \sum_i \mathbf{x}_i / n$.
- The sample mean is the maximum-likelihood estimate for the class mean!

Example: General Guassian

- Parameter vector has $\theta_1 = \mu$ and $\theta_2 = \Sigma$.
In the univariate case, $\theta_2 = \sigma^2$.

- $\ln p(x_i|\theta) = -\frac{1}{2} \ln[2\pi\theta_2] - \frac{1}{2\theta_2}(x_i - \theta_1)^2$.

- $\frac{\partial \ln p(x_i|\theta)}{\partial \theta_1} = \frac{1}{\theta_2}(x_i - \theta_1)$. $\frac{\partial \ln p(x_i|\theta)}{\partial \theta_2} = -\frac{1}{2\theta_2} + \frac{(x_i - \theta_1)^2}{2\theta_2^2}$

- Set components of the gradient individually to zero,
 $\sum_i \frac{(x_i - \hat{\theta}_1)}{\hat{\theta}_2} = 0$ and $-\sum_i \frac{1}{\hat{\theta}_2} + \sum_i \frac{(x_i - \hat{\theta}_1)^2}{\hat{\theta}_2^2} = 0$.

- Rearranging, we get $\hat{\theta}_1 = \hat{\mu} = \sum_i x_i/n$.
 $\hat{\theta}_2 = \hat{\sigma}^2 = \sum_i (x_i - \hat{\mu})^2/n$.
- We can similarly get for multivariate case:
 $\hat{\mu} = \sum_i \mathbf{x}_i/n$ and $\hat{\Sigma} = \sum_i (\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^\top/n$.
- Sample mean and variance are also the maximum-likelihood mean and variance respectively!

Biased Estimate

- The ML estimate for variance (or covariance) is *biased*. Its expected value over all data sets of size n is not equal to the true value.
- $\mathcal{E}\{\hat{\Sigma}\} = \frac{n-1}{n}\Sigma$.
- A simple unbiased estimator is:
$$\mathbf{C} = \frac{1}{n-1} \sum_i (\mathbf{x}_i - \hat{\mu})(\mathbf{x}_i - \hat{\mu})^\top$$
- \mathbf{C} is unbiased, $\hat{\Sigma}$ is asymptotically unbiased.
- Neither is “correct” or “incorrect”. They are 2 different estimates of the parameters.

Number of Training Samples

- Parameters are **learned** from the training samples D .
- How many samples n are required for d -dimensional \mathbf{x} ?
- Clearly, $n > d$. Else the covariance matrix Σ will be singular!
- $\mathbf{x}\mathbf{x}^\top = [x_1\mathbf{x} \quad x_2\mathbf{x} \quad \cdots \quad x_d\mathbf{x}]$. Clearly of rank 1.
 $\mathbf{x}\mathbf{x}^\top + \mathbf{y}\mathbf{y}^\top = [x_1\mathbf{x} + y_1\mathbf{y} \quad x_2\mathbf{x} + y_2\mathbf{y} \quad \cdots \quad x_d\mathbf{x} + y_d\mathbf{y}]$. Of rank 2. (All columns are linear combination of 2 vectors!)
- More samples the merrier usually.

Dimensionality

- More the merrier?
- On first sight, seems so. Classes that are not separable in lower dimensions could be so in higher dimensions.
- This is not always the case. Classification performance worsens with addition of more features.
- Reasons: Independence may not hold, parametric assumption may be violated etc.

- Classification error decreases if the Mahalanobis distance $r^2 = (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})$ increases.
- For independent features, $r^2 = \sum_j \left(\frac{x_j - \mu_j}{\sigma_j} \right)^2$.
- Features whose means are wide apart with respect to the variance are good features.
- In practice, the classification can get worse by adding features beyond some limit.

Given Insufficient Samples

- Simplify the model: Reduce dimensions, assume independence etc.
- Feature independence assumption results in reasonable classifiers even when features are not independent.
- **Overfitting:** Too many parameters may overfit the data, resulting in poor generalization.
- Generalization: doing well on data not used for training.

Dimensionality Reduction

- Question: How to represent data using few values?
- How to represent data using a single point in the least squared error sense?
- Minimize squared error $\mathbf{J}(\mathbf{x}) = \sum_{i=1}^n (\mathbf{x} - \mathbf{x}_i)^T (\mathbf{x} - \mathbf{x}_i)$.
- $\nabla_{\mathbf{x}} \mathbf{J} = 0 = \sum_i (\mathbf{x} - \mathbf{x}_i)$. $\mathbf{x} = \sum_i \mathbf{x}_i / n$.
- **The sample mean** is the best zero-dimensional representative of the data!

One-Dimensional Approximation

- Best line to which the data is projected. Let \mathbf{e} be the direction vector (unit vector in the line direction).
- Represent points as $\mathbf{x} = \mu + a\mathbf{e}$.
Each point \mathbf{x}_i corresponds to a number a_i .
- Minimize $\mathbf{J}(a_1, \dots, a_n, \mathbf{e}) = \sum_i (\mu + a_i\mathbf{e} - \mathbf{x}_i)^\top (\mu + a_i\mathbf{e} - \mathbf{x}_i)$
 $= \sum_i a_i^2 \|\mathbf{e}\|^2 - 2 \sum_i a_i \mathbf{e}^\top (\mathbf{x}_i - \mu) + \sum_i \|\mathbf{x}_i - \mu\|^2$.
- Which a_i minimizes \mathbf{J} , for a given \mathbf{e} ?
- Set $\partial \mathbf{J} / \partial a_i = 0$. Or, $2a_i \mathbf{e}^\top \mathbf{e} - 2\mathbf{e}^\top (\mathbf{x}_i - \mu) = 0$.
Gives $a_i = \mathbf{e}^\top (\mathbf{x}_i - \mu)$. Project each sample onto \mathbf{e} .

- Substituting this, we get $\mathbf{J} = -\sum_i a_i^2 + \sum_i \|\mathbf{x}_i - \mu\|^2$
 $= -\sum_i \mathbf{e}^\top (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^\top \mathbf{e} + K = -\mathbf{e}^\top \mathbf{S} \mathbf{e} + K$, where the **scatter matrix** $\mathbf{S} = \sum_i (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^\top$ is related to $\hat{\Sigma}$, the sample covariance matrix.
- \mathbf{J} is minimum when $\mathbf{e}^\top \mathbf{S} \mathbf{e}$ (always positive) is maximum, with $\|\mathbf{e}\| = 1$. Maximize $\mathbf{e}^\top \mathbf{S} \mathbf{e} - \lambda(\mathbf{e}^\top \mathbf{e} - 1)$ with respect to \mathbf{e} .
- Differentiating, $2\mathbf{S}\mathbf{e} - 2\lambda\mathbf{e} = \mathbf{0}$. Gives $\mathbf{S}\mathbf{e} = \lambda\mathbf{e}$. Therefore, $\mathbf{e}^\top \mathbf{S} \mathbf{e} = \lambda$. Minimum of \mathbf{J} corresponds to largest λ .
- $\mathbf{e} = \phi_1$ is the eigenvector corresponding to the largest eigenvalue of \mathbf{S} or the sample covariance matrix.

Optimal Representation

- The best d' -dimensional representation consists of projection onto the top d' number of eigenvectors of $\hat{\Sigma}$.
- $\mathbf{x} = \mu + \sum_{i=1}^{d'} a_i \phi_i$ is the best approximation for \mathbf{x} .
 $a_i = \phi_i^\top (\mathbf{x}_i - \mu)$ are its principal components.
- Since μ, ϕ_i are known quantities, $\mathbf{x}' = [a_1 \ a_2 \ \cdots \ a_{d'}]^\top$ is the optimal reduced dimension representation of \mathbf{x} .
- $\mathbf{x}' = \Phi'^\top \mathbf{x}$ where $\Phi' = [\phi_1 \ \phi_2 \ \cdots \ \phi_{d'}]$ converts each sample to the reduced representation.

Principal Component Analysis

- Question: How to represent data using few values?
- How to represent data using a single point in the least squared error sense? **Use the sample mean!**
- How to represent data using its variation along one direction through the mean? **Project to a line along the largest eigenvalue of $S = \sum_i (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$ passing through the mean.** S is the *scatter-matrix* of the data samples.
- For projections to k directions: Use eigenvectors corresponding to top k eigenvalues!

- $\bar{\mathbf{x}} = \sum_{i=1}^d a_i \phi_i$, where ϕ_i are the eigenvectors.
- Projections a_i of the data to these eigenvector directions are called its principal components.
- If the eigenvectors are arranged in the decreasing order of their corresponding eigenvalues, these components are also in the decreasing order of importance.
- To eliminate k least important ones, do so from bottom. To keep the k most important ones, do so from the top!
- \mathbf{x} is represented by $[a_1, a_2, \dots, a_k]$ (with known μ, ϕ_i 's) in the reduced k -dimensional space.
- $\hat{\mathbf{x}} = \mu + \sum_{i=1}^k a_i \phi_i$ is the approximate point.

Eigenvectors, Eigenvalues, etc.

- $\mathbf{S}\Phi = \Phi\Lambda$ where $\Phi = [\phi_1, \phi_2, \dots, \phi_d]$ where ϕ_i s are the eigenvectors and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_d)$ where λ_i s are the eigenvalues of \mathbf{S} .
- \mathbf{S} is a $d \times d$ scatter matrix. Sometimes, the entire data is taken as the feature vector. Thus, the dimension $d \gg n$ where n is the number of data samples.
- $\mathbf{S} = \mathbf{A}\mathbf{A}^\top$ where $\mathbf{A} = [\bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_n]$ is the measurement matrix with $(\bar{\mathbf{x}} = \mathbf{x} - \mu)$.

- If Φ is the eigenvector matrix of $\mathbf{A}\mathbf{A}^\top$ and Ψ is the eigenvector matrix of $\mathbf{A}^\top\mathbf{A}$,
 $\mathbf{A}^\top\mathbf{A}\Psi = \Psi\Lambda$. $\mathbf{A}\mathbf{A}^\top(\mathbf{A}\Psi) = (\mathbf{A}\Psi)\Lambda$. Thus, $\Phi = \mathbf{A}\Psi$.
Computationally, this method is better if $n \ll d$.

PCA Algorithm

- Find sample mean $\mu = \sum^i \mathbf{x}_i$.
- Compute $\bar{\mathbf{x}} = \mathbf{x} - \mu$ for all i .
- Let $\mathbf{A} = [\bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_n]$ be the measurement matrix.
- Compute eigenvectors Ψ and eigenvalues Λ of $\mathbf{A}^T \mathbf{A}$.
- Eigenvectors of $\mathbf{S} = \mathbf{A} \mathbf{A}^T$ are $\mathbf{A} \Psi$.
- Arrange Φ and Λ in the descending order of the λ_i values.

- Find k such that $\frac{\sum_{i=1}^k \lambda_i}{\sum_{i=1}^n \lambda_i} > \Theta$ for some threshold Θ .
- Let ϕ_1, \dots, ϕ_k be the corresponding eigenvectors.
- Represent data using k -dimensional vectors $\mathbf{y} = \Phi^T \mathbf{x}$.
- Features \mathbf{y} are uncorrelated; their covariance matrix is diagonal.
- Removes data redundancy and effects compression (represent using fewer components.)

Discrimination Between Classes

- PCAs represent data well, but could perform poorly in discriminating between two classes.
- The minute differences could be in the amount of data thrown out by PCA! They could do well in telling classes apart.
- What is a measure of discrimination between 2 classes??
- High distance between means; low variation within each class!
- What is the projection to one-dimension with maximum discrimination?

Most Discriminating Line

- Given: A set $\mathbf{D} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ of training samples partitioned into n_1 samples belonging to class ω_1 with mean \mathbf{m}_1 and n_2 samples belonging to ω_2 with mean \mathbf{m}_2 .
- Let $y = \mathbf{w}^T \mathbf{x}$ be the projection onto a line given by \mathbf{w} of point \mathbf{x} .
- Let μ_1, μ_2 be the means of projected points and $\mathbf{S}_1, \mathbf{S}_2$ be the scatter matrices of each class.
- $\mu_1 = \frac{1}{n_1} \sum_{\mathbf{x}_i \in \omega_1} \mathbf{w}^T \mathbf{x}_i = \mathbf{w}^T \mathbf{m}_1$. Similarly, $\mu_2 = \mathbf{w}^T \mathbf{m}_2$.

- Scatter of $y_i \in \omega_1$: $s_1 = \sum_{\mathbf{x}_i \in \omega_1} (\mathbf{w}^\top \mathbf{x}_i - \mu_1)^2$
 $= \sum_{\omega_1} \mathbf{w}^\top (\mathbf{x}_i - \mathbf{m}_1) (\mathbf{x}_i - \mathbf{m}_1)^\top \mathbf{w} = \mathbf{w}^\top \mathbf{S}_1 \mathbf{w}$

where \mathbf{S}_1 is the scatter matrix of class ω_1 .

Similarly, scatter of $y_i \in \omega_2$: $s_2 = \mathbf{w}^\top \mathbf{S}_2 \mathbf{w}$.

- Maximize the objective function (for separability):

$$J = \frac{|\mu_1 - \mu_2|^2}{s_1^2 + s_2^2} = \frac{(\mathbf{w}^\top \mu_1 - \mathbf{w}^\top \mu_2)^2}{\mathbf{w}^\top \mathbf{S}_1 \mathbf{w} + \mathbf{w}^\top \mathbf{S}_2 \mathbf{w}} = \frac{\mathbf{w}^\top \mathbf{S}_b \mathbf{w}}{\mathbf{w}^\top \mathbf{S}_w \mathbf{w}}$$

- $\mathbf{S}_w = \mathbf{S}_1 + \mathbf{S}_2 = \sum_{\omega_1} (\mathbf{x} - \mathbf{m}_1) (\mathbf{x} - \mathbf{m}_1)^\top + \sum_{\omega_2} (\mathbf{x} - \mathbf{m}_2) (\mathbf{x} - \mathbf{m}_2)^\top$ is the **within-class scatter matrix**.
- $\mathbf{S}_b = (\mathbf{m}_1 - \mathbf{m}_2) (\mathbf{m}_1 - \mathbf{m}_2)^\top$ is the **between-class scatter matrix**. This has rank 1.

Fisher Linear Discriminant

- Maximize $\mathbf{J}(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$ with respect to the line \mathbf{w} .
- $\nabla_{\mathbf{w}} \mathbf{J} = (k_1 \mathbf{S}_b \mathbf{w} - k_2 \mathbf{S}_w \mathbf{w}) / k_1^2 = 0$
- Generalized eigenvalue problem: $\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$.
- $\mathbf{S}_b \mathbf{w} = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \mathbf{w} = \kappa(\mathbf{m}_1 - \mathbf{m}_2)$ for some constant κ .
- If \mathbf{S}_w is non-singular, take $\mathbf{w} = \mathbf{S}_w^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$ as the solution, since scale of \mathbf{w} is not important,
- Fisher linear discriminant given by this line gives the best discrimination between 2 classes in a 1-D space.

- If $\omega_1 \sim N(\Sigma, \mathbf{m}_1)$ and $\omega_2 \sim N(\Sigma, \mathbf{m}_2)$, Bayesian decision boundary is $\mathbf{w}^T \mathbf{x} + w_0 = 0$ where $\mathbf{w} = \Sigma^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$.
- Fisher discriminant is optimal with sample means and covariances as estimates of the real ones.

Multiple Discriminants

- For a multiclass problem, project to $c - 1$ directions.
- $y_i = \mathbf{w}_i^T \mathbf{x}$. or $\mathbf{y} = \mathbf{W}^T \mathbf{x}$ with a $d \times (c - 1)$ matrix \mathbf{W} .
- Use $\mathbf{S}_B = \sum_{i=1}^c n_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T$ and $\mathbf{S}_W = \sum_{i=1}^c \mathbf{S}_i$ where \mathbf{m} is the total mean, \mathbf{m}_i is the mean of class ω_i and \mathbf{S}_i is its scatter matrix.
- In reduced space: $\bar{\mathbf{S}}_W = \mathbf{W}^T \mathbf{S}_W \mathbf{W}$ and $\bar{\mathbf{S}}_B = \mathbf{W}^T \mathbf{S}_B \mathbf{W}$.
- Maximize the criterion function $\mathbf{J}(\mathbf{W}) = \frac{|\mathbf{W}^T \mathbf{S}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_W \mathbf{W}|}$ to get best discrimination.
- Generalized eigenvectors of $\mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{S}_W \mathbf{w}_i$.

Parameter Estimation: Summary

- Parameters that define the probability distributions are estimated from training data.
- Choose $\hat{\theta}$ that maximizes the conditional probability $p(D|\theta)$ of the training samples.
- For Gaussian case, sample means and covariances are good estimates for the distribution parameters.
- Compute a reduced representation using PCA or discriminant analysis before fitting the parameters.