بسم الله الرحمن الرحیم

بازشناسی الگو

درس ۱۰

انتخاب ویژگی

Feature Selection

کاظم فولادی قلعه
دانشکده مهندسی، پردیس فارابی
دانشگاه تهران

http://courses.fouladi.ir/pr

انتخاب ویژگی

۱

مقدمه

FEATURE SELECTION

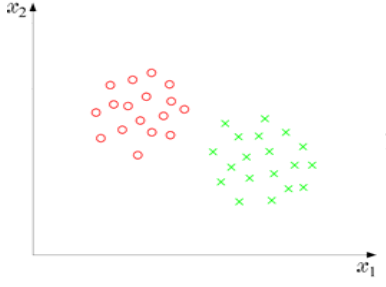
- **The goals**:
  - Select the "optimum" number l of features
  - Select the "best" l features

- Large l has a three-fold disadvantage:
  - High computational demands
  - Low generalization performance
  - Poor error estimates

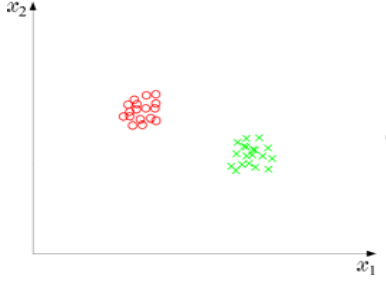
- Given N
  - l must be large enough to learn
    - what makes classes different
    - what makes patterns in the same class similar
  - l must be small enough not to learn what makes patterns of the same class different.
  - In practice, $l < N/3$ has been reported to be a sensible choice for a number of cases.

- Once l has been decided, choose the l most informative features
  - Best: **Large** between class distance,
    Small within class variance

Bad choice



Not bad choice



Good choice

- The basic philosophy
  - Discard individual features with poor information content
  - The remaining information rich features are examined jointly as vectors

انتخاب ویژگی

۲

انتخاب ویژگی بر اساس آزمون فرض آماری

❖ Feature Selection Based on Statistical Hypothesis Testing

➢ **The Goal:** For each individual feature, find whether the values, which the feature takes for the different classes, differ significantly.
That is, answer

$$\begin{cases} H_1: & \text{The values of the feature differ significantly} \\ H_0: & \text{The values of the feature do not differ significantly} \end{cases}$$

If they do not differ significantly,
reject feature from subsequent stages.

❖ Hypothesis Testing Basics

➢ Hypothesis Testing: *The steps:*

- N measurements $x_i, i = 1,2,...,N$ are known
- Define a function of them

  $q = f(x_1, x_2, ..., x_N)$: test statistic

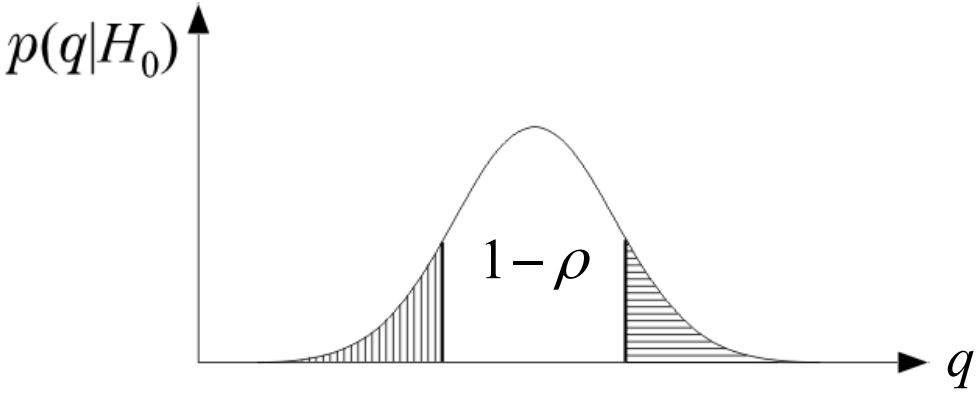
  so that $p_q(q;\theta)$ is easily parameterized in terms of θ.
- Let D be an interval, where q has a high probability to lie under H_0, i.e., $p_q(q|\theta_0)$
- Let $\overline{D}$ be the complement of D

  $D \longrightarrow$ Acceptance Interval

  $\overline{D} \longrightarrow$ Critical Interval
- If q, resulting from $x_1, x_2, ..., x_N$, lies in D, we accept H_0, otherwise we reject it.

➢ Probability of an error

$$p_q(q \in \overline{D} \,|\, H_0) = \rho$$



- ρ is preselected and it is known as the significance level.

❖ **Application:** The known variance case:

- Let x be a random variable and the experimental samples, $x_i = 1,2,\ldots,N$, are assumed mutually independent. Also let

$$E[x] = \mu$$

$$E[(x-\mu)^2] = \sigma^2$$

- Compute the sample mean

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

- This is also a random variable with mean value

$$E[\bar{x}] = \frac{1}{N}\sum_{i=1}^{N} E[x_i] = \mu$$

That is, it is an Unbiased Estimator

- The variance $\sigma_{\bar{x}}^2$

$$E[(\bar{x}-\mu)^2]=E[(\frac{1}{N}\sum_{i=1}^{N}x_i-\mu)^2]$$

$$=\frac{1}{N^2}\sum_{i=1}^{N}E[(x_i-\mu)^2]+\frac{1}{N^2}\sum_i\sum_j E[(x_i-\mu)(x_j-\mu)]$$

Due to independence

$$\sigma_{\bar{x}}^2=\frac{1}{N}\sigma_x^2$$

That is, it is Asymptotically Efficient

- Hypothesis test

$$H_1: E[x]\neq\hat{\mu}$$
$$H_0: E[x]=\hat{\mu}$$

- Test Statistic: Define the variable $q=\dfrac{\bar{x}-\hat{\mu}}{\sigma/\sqrt{N}}$

➢ Central limit theorem under H_0

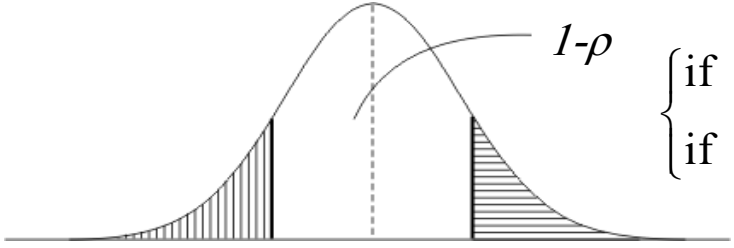
$$p_{\bar{x}}(\bar{x}) = \frac{\sqrt{N}}{\sqrt{2\pi}\sigma} \exp\left(-\frac{N(\bar{x} - \hat{\mu})^2}{2\sigma^2} \right) \qquad \bar{x} \sim N\left(\hat{\mu}, \frac{\sigma^2}{N}\right)$$

➢ Thus, under H_0

$$p_q(q) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{q^2}{2} \right) \quad q \sim N(0,1) \qquad \boxed{q = \frac{\bar{x} - \hat{\mu}}{\sigma / \sqrt{N}}}$$

➢ The decision steps

- Compute q from x_i, $i = 1, 2, \ldots, N$
- Choose significance level ρ
- Compute from $N(0,1)$ tables $D = [-x_\rho, x_\rho]$



$$\begin{cases} \text{if} \quad q \in D & \text{accept } H_0 \\ \text{if} \quad q \in \overline{D} & \text{reject } H_0 \end{cases}$$

➢ An example:

A random variable x has variance $\sigma^2 = (0.23)^2$. $N = 16$ measurements are obtained giving $\bar{x} = 1.35$. The significance level is $\rho = 0.05$.

Test the hypothesis $\begin{cases} H_0 : \mu = \hat{\mu} = 1.4 \\ H_1 : \mu \neq \hat{\mu} \end{cases}$

➢ Since σ^2 is known, $q = \dfrac{\bar{x} - \hat{\mu}}{\sigma / 4}$ is $N(0,1)$.

From tables, we obtain the values with acceptance intervals $[-x_\rho, x_\rho]$ for normal $N(0,1)$

$1-\rho$	0.8	0.85	0.9	0.95	0.98	0.99	0.998	0.999
x_ρ	1.28	1.44	1.64	1.96	2.32	2.57	3.09	3.29

➢ Thus

$$\text{Prob}\left\{-1.967 < \frac{\bar{x} - \hat{\mu}}{0.23/4} < 1.967\right\} = 0.95$$

or

$$\text{Prob}\left\{-0.113 < \bar{x} - \hat{\mu} < 0.113\right\} = 0.95$$

or

$$\boxed{\text{Prob}\left\{1.237 < \hat{\mu} < 1.463\right\} = 0.95}$$

- Since $\hat{\mu} = 1.4$ lies within the above acceptance interval, we accept H_0, i.e.,

$$\mu = \hat{\mu} = 1.4$$

The interval [1.237, 1.463] is also known as **confidence interval** at the $1 - \rho = 0.95$ level.

We say that: There is no evidence at the 5% level that the mean value is not equal to $\hat{\mu}$

❖ The Unknown Variance Case

➢ Estimate the variance. The estimate

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2$$

is unbiased, i.e.,

$$E[\hat{\sigma}^2] = \sigma^2$$

➢ Define the test statistic

$$q = \frac{\bar{x} - \mu}{\hat{\sigma}/\sqrt{N}}$$

- This is no longer Gaussian. If x is Gaussian, then q follows a ***t*-distribution**, with N-1 degrees of freedom

$$q = \frac{\bar{x} - \mu}{\hat{\sigma} / \sqrt{N}}$$

- An example:

x is Gaussian, $N = 16$, obtained from measurements, $\bar{x} = 1.35$ and $\hat{\sigma}^2 = (0.23)^2$. Test the hypothesis

$H_0 : \mu = \hat{\mu} = 1.4$

at the significance level $\rho = 0.025$.

➢ Table of acceptance intervals for t-distribution

Degrees of Freedom	1-ρ →	0.9	0.95	0.975	0.99
12		1.78	2.18	2.56	3.05
13		1.77	2.16	2.53	3.01
14		1.76	2.15	2.51	2.98
15		1.75	2.13	2.49	2.95
16		1.75	2.12	2.47	2.92
17		1.74	2.11	2.46	2.90
18		1.73	2.10	2.44	2.88

➢ $$\text{Prob}\left\{-2.49 < \frac{\bar{x} - \hat{\mu}}{\hat{\sigma}/4} < 2.49\right\} = 0.975$$

$$1.207 < \hat{\mu} < 1.493$$

Thus, $\hat{\mu} = 1.4$ is accepted

❖ Application in Feature Selection

- The goal here is to test against zero the difference $\mu_1 - \mu_2$ of the respective means in ω_1, ω_2 of a single feature.
- Let x_i $i = 1, \ldots, N$, the values of a feature in ω_1
- Let y_i $i = 1, \ldots, N$, the values of the same feature in ω_2
- Assume in both classes $\sigma_1^2 = \sigma_2^2 = \sigma^2$ (unknown or not)
- The test becomes $\begin{cases} H_0: \ \Delta\mu = \mu_1 - \mu_2 = 0 \\ H_1: \ \Delta\mu \neq 0 \end{cases}$

- Define

  $z = x - y$

- Obviously

  $E[z] = \mu_1 - \mu_2$

- Define the average

$$\bar{z} = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i) = \bar{x} - \bar{y}$$

- Known Variance Case: Define $q = \dfrac{(\bar{x} - \bar{y}) - (\hat{\mu}_1 - \hat{\mu}_2)}{\sigma\sqrt{\dfrac{2}{N}}}$

  - This is $N(0,1)$ and one follows the procedure as before.

➢ Unknown Variance Case:

Define the test statistic $$q = \frac{(\bar{x} - \bar{y}) - (\mu_1 - \mu_2)}{S_z \sqrt{\frac{2}{N}}}$$

$$S_z^2 = \frac{1}{2N-2}\left(\sum_{i=1}^{N}(x_i - \bar{x})^2 + \sum_{i=1}^{N}(y_i - \bar{y})^2\right)$$

- q is *t*-distribution with $2N-2$ degrees of freedom,
- Then apply appropriate tables as before.

➢ Example: The values of a feature in two classes are:

ω_1: 3.5, 3.7, 3.9, 4.1, 3.4, 3.5, 4.1, 3.8, 3.6, 3.7

ω_2: 3.2, 3.6, 3.1, 3.4, 3.0, 3.4, 2.8, 3.1, 3.3, 3.6

Test if the mean values in the two classes differ significantly, at the significance level $\rho = 0.05$

- We have

$$\omega_1: \ \bar{x} = 3.73, \ \hat{\sigma}_1^2 = 0.0601$$
$$\omega_2: \ \bar{y} = 3.25, \ \hat{\sigma}_2^2 = 0.0672$$

For $N = 10$

$$S_z^2 = \frac{1}{2}(\hat{\sigma}_1^2 + \hat{\sigma}_2^2)$$

$$q = \frac{(\bar{x} - \bar{y}) - 0}{S_z\sqrt{\frac{2}{10}}}$$

$$\boxed{q = 4.25}$$

- From the table of the t-distribution with $2N - 2 = 18$ degrees of freedom and $\rho = 0.05$, we obtain $D = [-2.10, 2.10]$ and since $q = 4.25$ is outside D, H_1 is accepted and the feature is selected.

انتخاب ویژگی

۳

معیارهای جداپذیری طبقه‌ها

❖ Class Separability Measures

The emphasis so far was on individually considered features. However, such an approach cannot take into account **existing correlations among the features**. That is, two features may be rich in information, but if they are highly correlated we need not consider both of them. To this end, in order to search for possible correlations, we consider features jointly as elements of vectors. To this end:

- Discard poor in information features, by means of a statistical test.

- Choose the maximum number, ℓ, of features to be used.
  This is dictated by the specific problem.
  (e.g., the number, N, of available training patterns and the type of the classifier to be adopted)

➢ Combine remaining features to search for the "best" combination. To this end:

- Use different feature combinations to form the feature vector. Train the classifier, and choose the combination resulting in the best classifier performance.

  A major disadvantage of this approach is the high complexity. Also, local minima, may give misleading results.

- Adopt a **class separability measure** and choose the best feature combination against this cost.

➢ Class separability measures: Let $\underline{x}$ be the current feature combination vector.

- **Divergence.** To see the rationale behind this cost, consider the two-class case. Obviously, if on the average the value of $\ln \dfrac{p(\underline{x}\,|\,\omega_1)}{p(\underline{x}\,|\,\omega_2)}$ is close to zero, then $\underline{x}$ should be a poor feature combination. Define:

$$D_{12} = \int_{-\infty}^{+\infty} p(\underline{x}\,|\,\omega_1) \ln \frac{p(\underline{x}\,|\,\omega_1)}{p(\underline{x}\,|\,\omega_2)} d\underline{x}$$

$$D_{21} = \int_{-\infty}^{+\infty} p(\underline{x}\,|\,\omega_2) \ln \frac{p(\underline{x}\,|\,\omega_2)}{p(\underline{x}\,|\,\omega_1)} d\underline{x}$$

$$d_{12} = D_{12} + D_{21}$$

d_{12} is known as the **divergence** and can be used as a class separability measure.

– For the multi-class case, define d_{ij} for every pair of classes ω_i, ω_j ; and the average divergence is defined as

$$d = \sum_{i=1}^{M} \sum_{j=1}^{M} P(\omega_i) P(\omega_j) d_{ij}$$

– Some properties:

$$d_{ij} \geq 0$$

$$d_{ij} = 0, \text{ if } i = j$$

$$d_{ij} = d_{ji}$$

– Large values of d are indicative of good feature combination.

➢ **Scatter Matrices.** These are used as a measure of the way data are scattered in the respective feature space.

- Within-class scatter matrix

$$S_w = \sum_{i=1}^{M} P_i S_i$$

where

$$S_i = E\left[\left(\underline{x} - \underline{\mu}_i\right)\left(\underline{x} - \underline{\mu}_i\right)^T\right]$$

and

$$P_i \equiv P(\omega_i) \approx \frac{n_i}{N}$$

n_i the number of training samples in ω_i.

Trace $\{S_w\}$ is a measure of the average variance of the features.

- Between-class scatter matrix

$$S_b = \sum_{i=1}^{M} P_i \left(\underline{\mu}_i - \underline{\mu}_0\right)\left(\underline{\mu}_i - \underline{\mu}_0\right)^T$$

$$\underline{\mu}_0 = \sum_{i=1}^{M} P_i \underline{\mu}_i$$

Trace $\{S_b\}$ is a measure of the average distance of the mean of each class from the respective global one.

- Mixture scatter matrix

$$S_m = E\left[\left(\underline{x} - \underline{\mu}_0\right)\left(\underline{x} - \underline{\mu}_0\right)^{\mathrm{T}}\right]$$

It turns out that:

$$S_m = S_w + S_b$$

➢ Measures based on Scatter Matrices.

- $J_1 = \dfrac{\text{trace}\{S_m\}}{\text{trace}\{S_w\}}$
- $J_2 = \dfrac{|S_m|}{|S_w|} = |S_w^{-1} S_m|$
- $J_3 = \text{trace}\{S_w^{-1} S_m\}$

- Other criteria are also possible, by using various combinations of S_m, S_b, S_w.

The above J_1, J_2, J_3 criteria take high values for the cases where:

- Data are clustered together within each class.
- The means of the various classes are far.

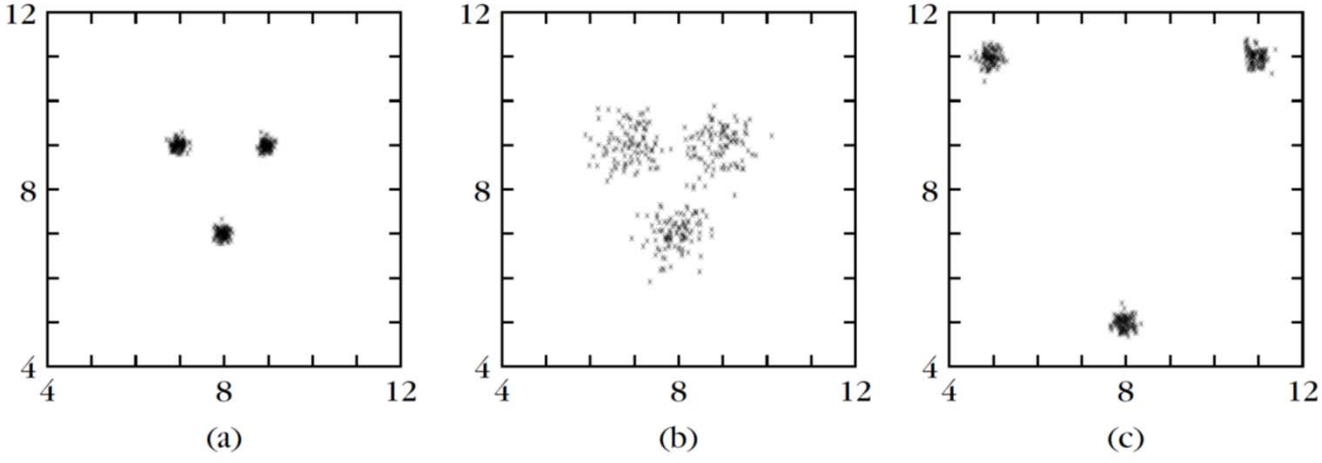


FIGURE 5.5

Classes with (a) small within-class variance and small between-class distances, (b) large within-class variance and small between-class distances, and (c) small within-class variance and large between-class distances.

- **Fisher's discriminant ratio**. In one dimension and for two equiprobable classes the determinants become:

$$|S_w| \propto \sigma_1^2 + \sigma_2^2$$

$$|S_b| \propto (\mu_1 - \mu_2)^2$$

and

$$\frac{|S_b|}{|S_w|} = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$

known as Fischer's ratio.

انتخاب ویژگی

۴

انتخاب زیرمجموعه‌ی ویژگی‌ها

❖ Ways to combine features:

Trying to form all possible combinations of ℓ features from an original set of m selected features is a computationally hard task.
Thus, a number of suboptimal searching techniques have been derived.

- Sequential forward selection
- Sequential backward selection
- Floating Search Methods

➢ Sequential forward selection.

Let x_1, x_2, x_3, x_4 the available features ($m = 4$).
The procedure consists of the following steps:

- **Adopt a class separability criterion** (could also be the error rate of the respective classifier). Compute its value for ALL features considered jointly $[x_1, x_2, x_3, x_4]^T$.
- **Eliminate one feature and for each of the possible resulting combinations**, that is $[x_1, x_2, x_3]^T$, $[x_1, x_2, x_4]^T$, $[x_1, x_3, x_4]^T$, $[x_2, x_3, x_4]^T$, compute the class reparability criterion value C. Select the best combination, say $[x_1, x_2, x_3]^T$.

- From the above selected feature vector eliminate one feature and for each of the resulting combinations, $[x_1, x_2]^T, [x_2, x_3]^T, [x_1, x_3]^T$ compute C and select the best combination.

The above selection procedure shows how one can start from m features and end up with the "best" ℓ ones. Obviously, the choice is suboptimal. The number of required calculations is:

$$\ell m - \frac{\ell(\ell-1)}{2}$$

In contrast, a full search requires:

$$\binom{m}{\ell} = \frac{m!}{\ell!(m-\ell)!}$$

operations.

➢ Sequential backward selection.

Here the reverse procedure is followed.

- **Compute C for each feature**. Select the "best" one, say x_1
- **For all possible 2D combinations** of x_1, i.e., $[x_1, x_2]$, $[x_1, x_3]$, $[x_1, x_4]$ compute C and choose the best, say $[x_1, x_3]$.
- **For all possible 3D combinations** of $[x_1, x_3]$, e.g., $[x_1, x_3, x_2]$, etc., compute C and choose the best one.

The above procedure is repeated till the "best" vector with ℓ features has been formed. This is also a suboptimal technique, requiring:

$$1+\frac{1}{2}\left((m+1)m-\ell(\ell+1)\right)$$

operations.

➢ Floating Search Methods

The above two procedures suffer from the nesting effect.
Once a bad choice has been done, there is no way to reconsider it in the following steps.

In the floating search methods one is given the opportunity in reconsidering a previously discarded feature or to discard a feature that was previously chosen.

The method is still suboptimal, however it leads to improved performance, at the expense of complexity.

➢ Remarks:

- Besides suboptimal techniques, some optimal searching techniques can also be used, provided that the optimizing cost has certain properties, e.g., monotonic.

- Instead of using a class separability measure (filter techniques) or using directly the classifier (wrapper techniques), one can modify the cost function of the classifier appropriately, so that to perform feature selection and classifier design in a single step (embedded) method.

- For the choice of the separability measure a multiplicity of costs have been proposed, including information theoretic costs.

انتخاب ویژگی

۵

اشاره‌هایی به نظریه‌ی تعمیم

❖ Hints from Generalization Theory.

Generalization theory aims at providing general bounds that relate the error performance of a classifier with the number of training points, N, on one hand, and some classifier dependent parameters, on the other. Up to now, the classifier dependent parameters that we considered were the number of its free parameters and the dimensionality, ℓ, of the subspace, in which the classifier operates. (ℓ also affects the number of free parameters).

➢ Definitions

- Let the classifier be a binary one, i.e.,

$$f : \mathbb{R}^{\ell} \rightarrow \{0,1\}$$

- Let F be the set of all functions f that can be realized by the adopted classifier (e.g., changing the synapses of a given neural network different functions are implemented).

- The shatter coefficient $S(F,N)$ of the class F is defined as:

  the **maximum** number of dichotomies of N points that can be formed by the functions in $\boldsymbol{F}$.

  The maximum possible number of dichotomies is 2^N.
  However, NOT ALL dichotomies can be realized by the set of functions in $\boldsymbol{F}$.

- The **Vapnik-Chernovenkis (VC) dimension** of a class $\boldsymbol{F}$ is the largest integer k for which $S(F,k) = 2^k$. If $S(F,N)=2^N$, $\forall N$,

  we say that the VC dimension is infinite.

  – That is, VC is the integer for which the class of functions F can achieve all possible dichotomies, 2^k.

  – It is easily seen that the VC dimension of the single perceptron class, operating in the ℓ-dimensional space, is $\ell+1$.

– It can be shown that

$$S(F,N) \leq N^{V_c} + 1$$

V_c: the VC dimension of the class.

That is, the shatter coefficient is either 2^N (the maximum possible number of dichotomies) or it is upper bounded, as suggested by the above inequality.

In words, for finite V_c and large enough N, the **shatter coefficient is bounded by a polynomial growth**.

- Note that in order to have a polynomial growth of the shatter coefficient, N must be larger than the V_c dimension.

– The V_c dimension can be considered as an **intrinsic capacity** of the classifier, and, as we will soon see, only if the number of training vectors **exceeds** this number sufficiently, we can expect good generalization performance.

- The V_c dimension may or may **not** be related to the dimension ℓ and the number of free parameters.
  - Perceptron: $V_c = \ell + 1$
  - Multilayer perceptron with hard limiting activation function

$$2\left[\frac{k_n^h}{2}\right]\ell \le V_c \le 2k_w \log_2(ek_n)$$

where k_n^h is the total number of hidden layer nodes, k_n the total number of nodes, and k_w the total number of weights.

  - Let $\{\underline{x}_i\}$ be a training data sample and assume that

$$\|\underline{x}_i\| \le r, i = 1, 2, ..., N$$

Let also a hyperplane such that

$$\|w\|^2 \leq c$$

and

$$y_i\left(\underline{w}^T \underline{x}_i + b\right) \geq 1$$

(i.e., the constraints we met in the SVM formulation). Then

$$V_c \leq \left(r^2 c, \ell\right)$$

That is, by controlling the constant c, the V_c of the linear classifier can be less than ℓ. In other words, V_c can be controlled independently of the dimension.

Thus, by minimizing $\|w\|^2$ in the SVM, one attempts to keep V_c as small as possible. Moreover, one can achieve **finite** V_c dimension, **even for infinite dimensional spaces**. This is an explanation of the potential for good generalization performance of the SVM's, as this is readily deduced from the following bounds.

➢ Generalization Performance

- Let $P_e^N(f)$ be the error rate of classifier f, based on the N training points, also known as empirical error.

- Let $P_e(f)$ be the true error probability of f (also known as generalization error), when f is confronted with data outside the finite training set.

- Let P_e be the minimum error probability that can be attained over ALL functions in the set $\boldsymbol{F}$.

- Let f^* be the function resulting by minimizing the empirical (over the finite training set) error function.

- It can be shown that:
  - $$\text{prob}\left\{\max_{f\in F}\left(P_e^N(f)-P_e(f)\right)>\varepsilon\right\}\leq 8S(F,N)\exp\left(-\frac{N\varepsilon^2}{32}\right)$$
  - $$\text{prob}\left\{P_e(f^*)-P_e>\varepsilon\right\}\leq 8S(F,N)\exp\left(-\frac{N\varepsilon^2}{128}\right)$$
  - Taking into account that for finite V_c dimension, the growth of $S(F,N)$ is only polynomial, the above bounds tell us that for a large N:
    - $P_e^N(f)$ is close to $P_e(f)$, with high probability.
    - $P_e(f^*)$ is close to P_e, with high probability.

- Some more useful bounds
  - The minimum number of points, $N(\varepsilon,\rho)$, that guarantees, with high probability, a good generalization error performance is given by

$$N(\varepsilon,\rho)\leq \max\left\{\frac{k_1V_c}{\varepsilon^2}\ln\frac{k_2V_c}{\varepsilon^2},\frac{k_3}{\varepsilon^2}\ln\frac{8}{\rho}\right\}$$

$$N\geq N(\varepsilon,\rho)$$

That is, for any

$$\text{prob}\{P_e(f)-P_e>\varepsilon\}\leq\rho$$

Where, k_1,k_2,k_3 constants. In words, for $N\geq N(\varepsilon,\rho)$ the performance of the classifier is guaranteed, with high probability, to be close to the optimal classifier in the class F. $N(\varepsilon,\rho)$ is known as the sample complexity.

– With a probability of at least $1-\rho$ the following bound holds:

$$P_e(f) \leq P_e^N(f) + \Phi\left(\frac{V_c}{N}\right)$$

where

$$\Phi\left(\frac{V_c}{N}\right) = \sqrt{\frac{V_c\left(\ln\left(\frac{2N}{V_c}+1\right) - \ln\left(\frac{\rho}{4}\right)\right)}{N}}$$

➢ Remark: Observe that all the bounds given so far are:

- Dimension free
- Distribution free

❖ Model Complexity vs. Performance

This issue has already been touched in the form of overfitting in neural networks modeling and in the form of bias-variance dilemma. A different perspective of the issue is dealt below.

➢ Structural Risk Minimization (SRM)

- Let P_B be he Bayesian error probability for a given task.
- Let $P_e(f^*)$ be the true (generalization) error of an optimally design classifier f^*, from class F, given a finite training set.

$$P_e(f^*) - P_B = \left(P_e(f^*) - P_e\right) + \left(P_e - P_B\right)$$

P_e is the minimum error attainable in F

  - If the class F is small, then the first term is expected to be small and the second term is expected to be large. The opposite is true when the class F is large

- Let $F^{(1)}$, $F^{(2)}$, … be a sequence of nested classes:

$$F^{(1)} \subset F^{(2)} \subset \ldots$$

with increasing, yet finite V_c dimensions.

$$V_{c,F^{(1)}} \leq V_{c,F^{(2)}} \leq \ldots$$

Also, let

$$\lim_{i \to \infty} \inf_{f \in F^{(i)}} P_e(f) = P_B$$

For each N and class of functions $F^{(i)}$, $i = 1, 2, \ldots$, compute the optimum $f^*_{N,i}$, with respect to the empirical error. Then from all these classifiers choose the one than minimizes, over all i, the upper bound in:

$$P_e(f^*_{N,i}) \leq P_e^N(f^*_{N,i}) + \Phi\left(\frac{V_{c,F^{(i)}}}{N}\right)$$

That is,

$$f^*_N = \arg\min_i \left[P_e^N(f^*_{N,i}) + \Phi\left(\frac{V_{c,F^{(i)}}}{N}\right) \right]$$

- Then, as $N \to \infty$

$$P_e(f_N^*) \to P_B$$

  – The term

$$\Phi\left(\frac{V_{c,F^{(i)}}}{N}\right)$$

in the minimized bound is a complexity penalty term. If the classifier model is **simple** the penalty term is **small** but the empirical error term $P_e^N(f_{N,i}^*)$ will be **large**.
The opposite is true for complex models.

- The SRM criterion aims at achieving the best trade-off between performance and complexity.

بازشناسی الگو

انتخاب ویژگی

۶

ضابطه‌ی اطلاعات بیزی

➢ Bayesian Information Criterion (BIC)

Let N the size of the training set, $\underline{\theta}_m$ the vector of the unknown parameters of the classifier, K_m the dimensionality of $\underline{\theta}_m$, and m runs over all possible models.

- The BIC criterion chooses the model by minimizing:

$$BIC = -2L\left(\hat{\underline{\theta}}_m\right) + K_m \ln N$$

  - $L(\hat{\underline{\theta}}_m)$ is the log-likelihood computed at the ML estimate $\underline{\theta}_m$, and it is the performance index.
  - $K_m \ln N$ is the model complexity term.

- Akaike Information Criterion:

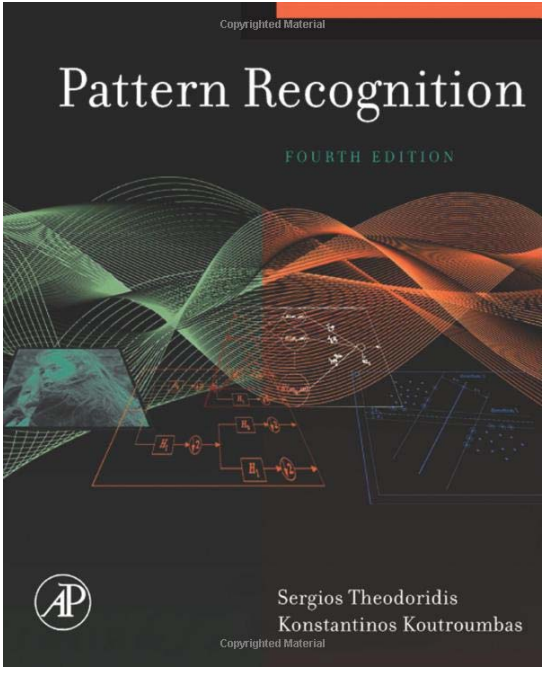
$$AIC = -2L\left(\hat{\underline{\theta}}_m\right) + 2K_m$$

انتخاب ویژگی

۷

منابع

منبع اصلی



S. Theodoridis, K. Koutroumbas,
Pattern Recognition,
Fourth Edition, Academic Press, 2009.

Chapter 5

CHAPTER

Feature Selection

5

5.1 INTRODUCTION

In all previous chapters, we considered the features that should be available prior to the design of the classifier. The goal of this chapter is to study methodologies related to the selection of these variables. As we pointed out very early in the book, a major problem associated with pattern recognition is the so-called curse of dimensionality (Section 2.5.6). The number of features at the disposal of the designer of a classification system is usually very large. As we will see in Chapter 7, this number can easily reach the order of a few dozens or even hundreds.

There is more than one reason to reduce the number of features to a sufficient minimum. Computational complexity is the obvious one. A related reason is that, although two features may carry good classification information when treated separately, there is little gain if they are combined into a feature vector because of a high mutual correlation. Thus, complexity increases without much gain. Another major reason is that imposed by the required generalization properties of the classifier, as discussed in Section 4.9 of Chapter 4. As we will state more formally at the end of this chapter, the higher the ratio of the number of training patterns N to the number of free classifier parameters, the better the generalization properties of the resulting classifier.

A large number of features are directly translated into a large number of classifier parameters (e.g., synaptic weights in a neural network, weights in a linear classifier). Thus, for a finite and usually limited number N of training patterns, keeping the number of features as small as possible is in line with our desire to design classifiers with good generalization capabilities. Furthermore, the ratio N/l enters the scene from another nearby corner. One important step in the design of a classification system is the performance evaluation stage, in which the classification error probability of the designed classifier is estimated. We not only need to design a classification system, but we must also assess its performance. As is pointed out in Chapter 10, the classification error estimate improves as this ratio becomes higher. In [Fine 83] it is pointed out that in some cases ratios as high as 10 to 20 were considered necessary.

261