University of Electronic Science and Technology of China

Multiple Kernel Learning

Pan Hao

Start Here

SVM and Kernel Trick

Multiple Kernel

Application

Contact Me
Email:691582372@qq.com
Tel:18936465392

Support Vector Machines

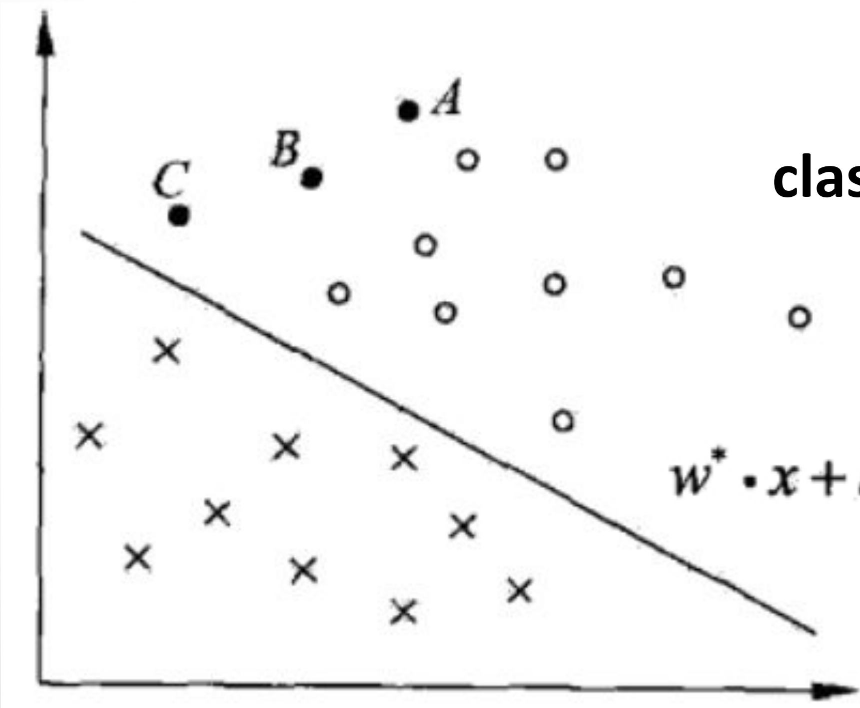
- Linear support vector machine in linearly separable case
- Linear support vector machine
- Non-linear support vector machine

simple

complex

Contact Me
Email:691582372@qq.com Tel:18936465392

Linear SVM in linearly separable case



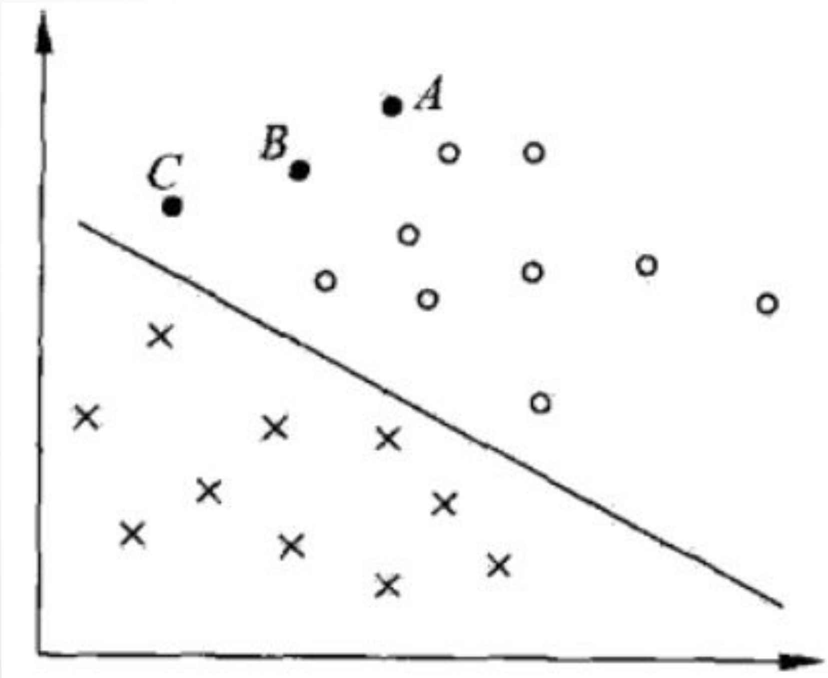
classification decision function:

$$f(x) = \mathrm{sign}(w^{*} \cdot x + b^{*})$$

$$w^{*} \cdot x + b^{*} = 0$$

classification hyperplane

Contact Me
Email:691582372@qq.com Tel:18936465392



Function margin

$$\hat{\gamma}_i = y_i(w \cdot x_i + b)$$

$$\hat{\gamma} = \min_{i=1,\cdots,N} \hat{\gamma}_i$$

Geometric margin

$$\gamma_i = y_i\left(\frac{w}{\|w\|} \cdot x_i + \frac{b}{\|w\|}\right)$$

$$\gamma = \min_{i=1,\cdots,N} \gamma_i$$

Contact Me
Email:691582372@qq.com Tel:18936465392

Constrained optimization problem

$$\max_{w,b} \quad \gamma$$

$$\text{s.t.} \quad y_i\left(\frac{w}{\|w\|}\cdot x_i+\frac{b}{\|w\|}\right)\geqslant\gamma, \quad i=1,2,\cdots,N$$

$$\gamma_i=\frac{\hat{\gamma}_i}{\|w\|}$$

$$\gamma=\frac{\hat{\gamma}}{\|w\|}$$

$$\max_{w,b} \quad \frac{\hat{\gamma}}{\|w\|}$$

$$\text{s.t.} \quad y_i(w\cdot x_i+b)\geqslant\hat{\gamma}, \quad i=1,2,\cdots,N$$

Contact Me
Email:691582372@qq.com Tel:18936465392

The common form of the optimization problem of SVM

Convex Quedratic Programming

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$

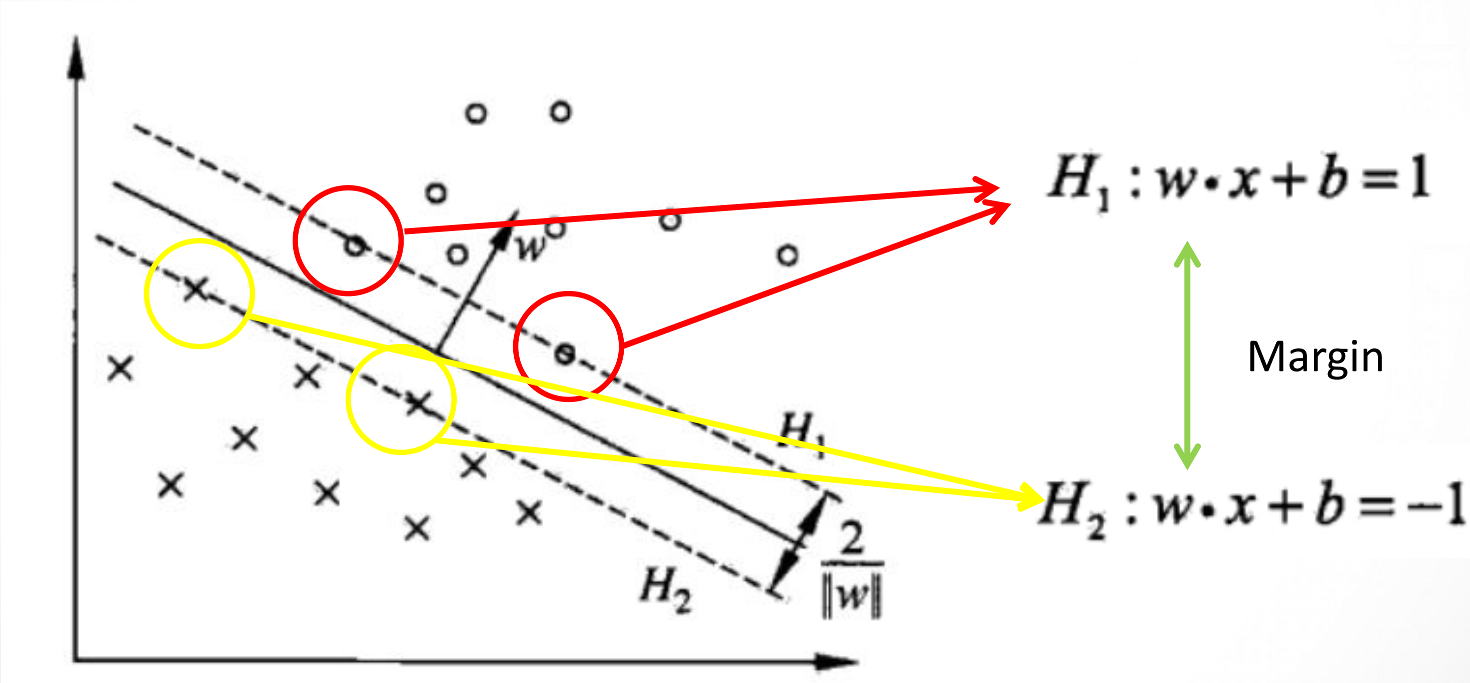
$$\text{s.t.} \quad y_i(w \cdot x_i + b) - 1 \geqslant 0, \quad i = 1, 2, \cdots, N$$

$$w^* \cdot x + b^* = 0$$

$$f(x) = \text{sign}(w^* \cdot x + b^*)$$

Contact Me
Email:691582372@qq.com Tel:18936465392

Support Vector



Contact Me
Email:691582372@qq.com Tel:18936465392

Dual Algorithm

Lagrange function:

$$L(w,b,\alpha)=\frac{1}{2}\|w\|^2-\sum_{i=1}^{N}\alpha_i y_i(w\cdot x_i+b)+\sum_{i=1}^{N}\alpha_i$$

Lagrange multiples: $\alpha=(\alpha_1,\alpha_2,\cdots,\alpha_N)^{\mathrm{T}}$, $\alpha_i\geqslant 0$, $i=1,2,\cdots,N$

According to the Lagrange dual method: $\min\limits_{w,b}\max\limits_{\alpha} L(w,b,\alpha)$

$$\boxed{\frac{1}{2}\|w\|^2}$$

New dual problem: $\max\limits_{\alpha}\min\limits_{w,b} L(w,b,\alpha)$

$$\begin{cases}\nabla_w L(w,b,\alpha)=w-\sum_{i=1}^{N}\alpha_i y_i x_i=0\\ \nabla_b L(w,b,\alpha)=\sum_{i=1}^{N}\alpha_i y_i=0\end{cases}\qquad\begin{cases}w=\sum_{i=1}^{N}\alpha_i y_i x_i\\ \sum_{i=1}^{N}\alpha_i y_i=0\end{cases}$$

Contact Me
Email:691582372@qq.com Tel:18936465392

$$L(w,b,\alpha)=\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j)-\sum_{i=1}^{N}\alpha_i y_i\left(\left(\sum_{j=1}^{N}\alpha_j y_j x_j\right)\cdot x_i+b\right)+\sum_{i=1}^{N}\alpha_i$$
$$=-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j)+\sum_{i=1}^{N}\alpha_i$$

$$\min_{w,b} L(w,b,\alpha)=-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j)+\sum_{i=1}^{N}\alpha_i$$

$$\max_{\alpha}\min_{w,b} L(w,b,\alpha)$$

$$\boxed{\begin{aligned}&\max_{\alpha}\ -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j)+\sum_{i=1}^{N}\alpha_i\\ &\text{s.t.}\quad \sum_{i=1}^{N}\alpha_i y_i=0\\ &\qquad \alpha_i \geqslant 0,\quad i=1,2,\cdots,N\end{aligned}}$$

Contact Me
Email:691582372@qq.com Tel:18936465392

The Algorithm of the Linear SVM in Linearly Separable Case

First:

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$

$$\alpha_i \geqslant 0, \quad i = 1, 2, \cdots, N$$

Second:

$$w^* = \sum_{i=1}^{N}\alpha_i^* y_i x_i$$

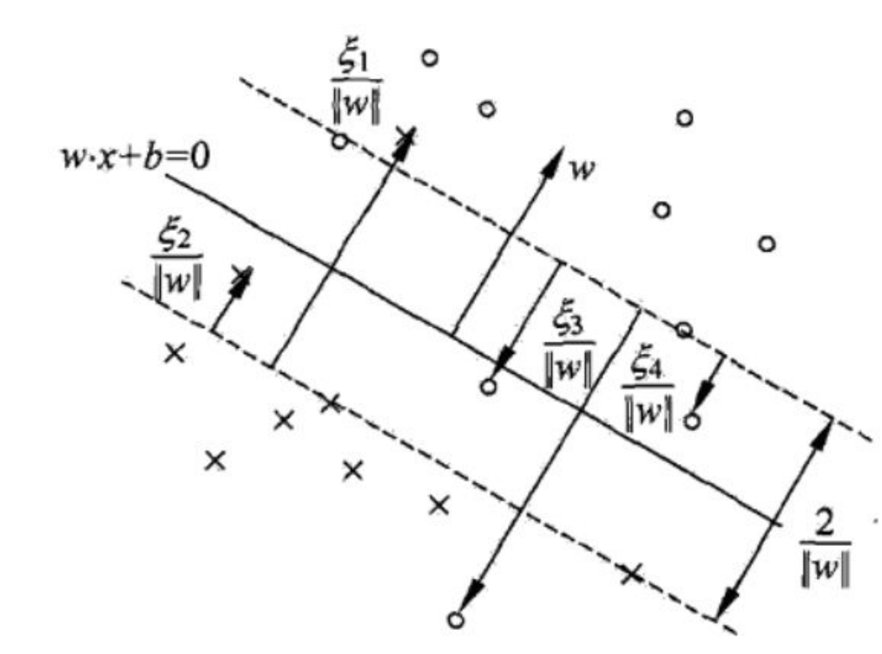
$$b^* = y_j - \sum_{i=1}^{N}\alpha_i^* y_i (x_i \cdot x_j)$$

Third:

$$w^* \cdot x + b^* = 0$$

$$f(x) = \operatorname{sign}(w^* \cdot x + b^*)$$

Contact Me
Email:691582372@qq.com Tel:18936465392



Soft Margin:

$$y_i(w \cdot x_i + b) \geqslant 1 - \xi_i$$

Target Function:

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i \quad , \quad C > 0$$

Contact Me
Email:691582372@qq.com Tel:18936465392

Prime Problem:

$$\min_{w,b,\xi} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i$$

$$\text{s.t.} \quad y_i(w \cdot x_i + b) \geqslant 1 - \xi_i, \quad i = 1,2,\cdots,N$$

$$\xi_i \geqslant 0, \quad i = 1,2,\cdots,N$$

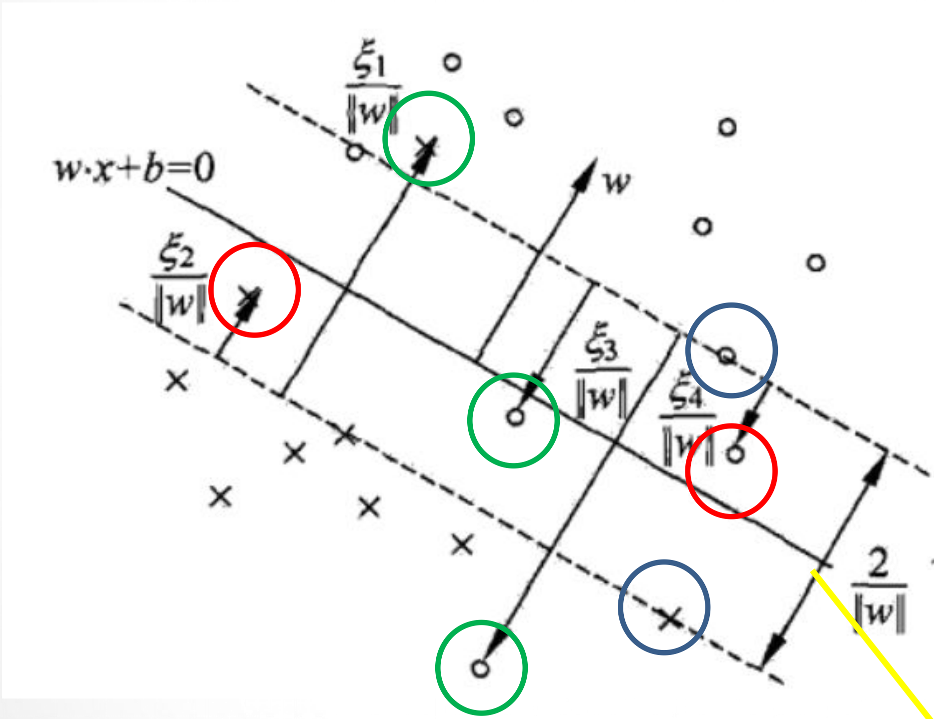
Dual Problem:

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$

$$0 \leqslant \alpha_i \leqslant C, \quad i = 1,2,\cdots,N$$

Contact Me
Email:691582372@qq.com Tel:18936465392



$$\alpha_i^* < C \,,\ \xi_i = 0$$

$$\alpha_i^* = C \,,\ 0 < \xi_i < 1$$

$$\alpha_i^* = C \,,\ \xi_i > 1$$

$$\alpha_i^* = C \,,\ \xi_i = 1$$

Contact Me
Email:691582372@qq.com Tel:18936465392

The Algorithm of SVM

First:

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$

$$0 \leqslant \alpha_i \leqslant C, \quad i = 1, 2, \cdots, N$$

Second:

$$w^* = \sum_{i=1}^{N}\alpha_i^* y_i x_i$$

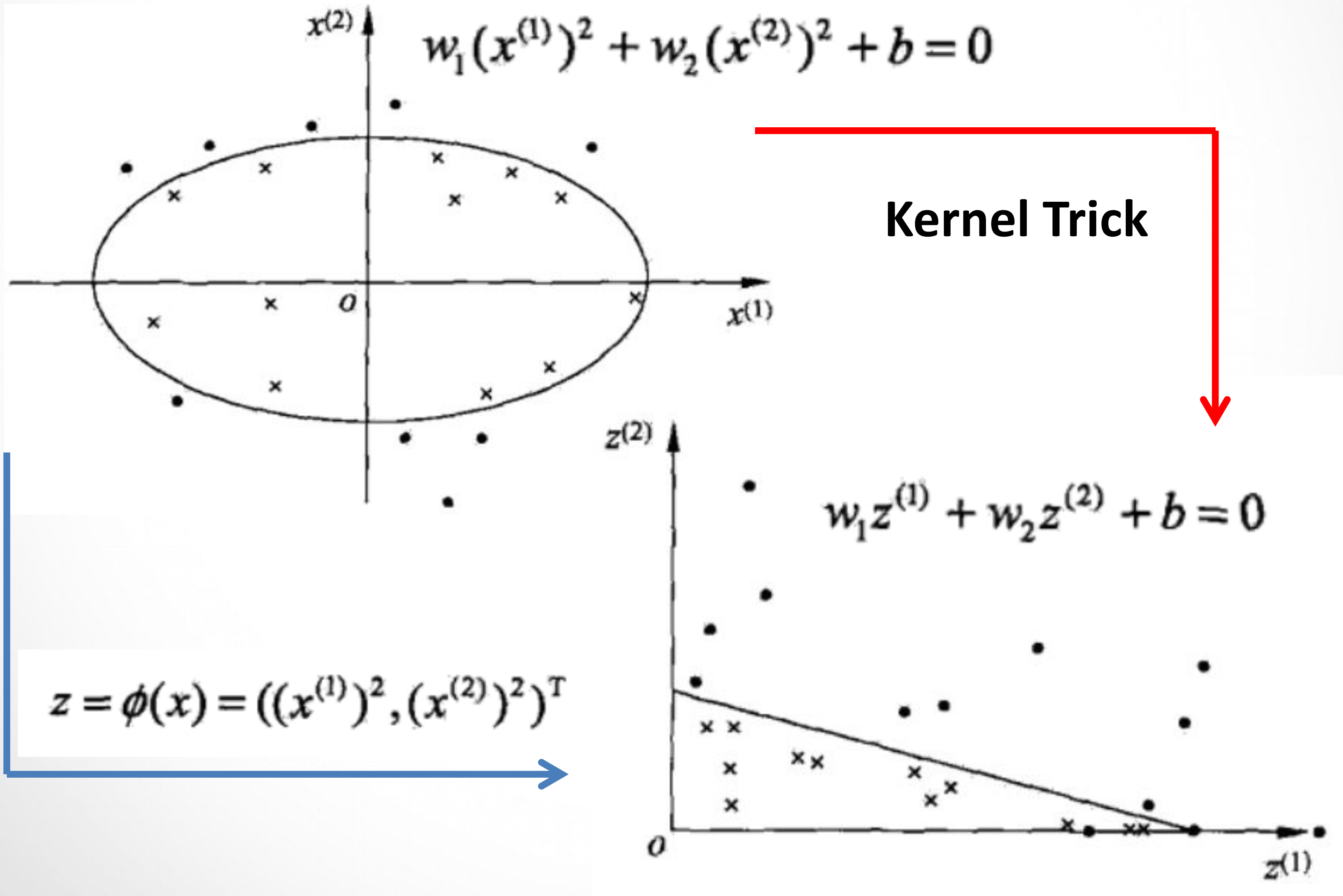
$$b^* = y_j - \sum_{i=1}^{N} y_i \alpha_i^* (x_i \cdot x_j)$$

Third:

$$w^* \cdot x + b^* = 0$$

$$f(x) = \operatorname{sign}(w^* \cdot x + b^*)$$

Contact Me
Email:691582372@qq.com Tel:18936465392



Contact Me
Email:691582372@qq.com Tel:18936465392

Kernel Function:

$\phi(x): \mathcal{X} \rightarrow \mathcal{H}$ (Hilbert Space)

$K(x,z) = \phi(x) \cdot \phi(z)$

Application in SVM:

$$x_i \cdot x_j \xrightarrow{\text{substitute}} K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$$

Dual Problem: $W(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{N}\alpha_i$

$$f(x) = \mathrm{sign}\left(\sum_{i=1}^{N_s} a_i^* y_i \phi(x_i) \cdot \phi(x) + b^*\right) = \mathrm{sign}\left(\sum_{i=1}^{N_s} a_i^* y_i K(x_i, x) + b^*\right)$$

Contact Me
Email:691582372@qq.com Tel:18936465392

The Algorithm of Non-linear SVM

First:

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{N}\alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$

$$0 \leqslant \alpha_i \leqslant C, \quad i = 1, 2, \cdots, N$$

Second:

$$b^* = y_j - \sum_{i=1}^{N}\alpha_i^* y_i K(x_i \cdot x_j)$$

Third:

$$f(x) = \operatorname{sign}\left(\sum_{i=1}^{N}\alpha_i^* y_i K(x \cdot x_i) + b^*\right)$$

Contact Me
Email:691582372@qq.com Tel:18936465392

The Kernel Trick Summary

- Any algorithm that only depends on dot products can benefit from the kernel trick
- This way, we can apply linear methods to vectorial as well as non-vectorial data
- Think of the kernel as a nonlinear similarity measure
- Examples of common kernels
  - Polynomial $k(x,x') = (\langle x,x' \rangle + c)^d$
  - Sigmoid $\tanh(\kappa \langle x,x' \rangle + \Theta)$
  - Gaussian $\exp(-\|x-x'\|^2/(2\,\sigma^2))$

Contact Me
Email:691582372@qq.com Tel:18936465392

Solving SVM Using Sequential Minimal Optimization

Dual Problem:

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i,x_j)-\sum_{i=1}^{N}\alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$

$$0 \leqslant \alpha_i \leqslant C, \quad i=1,2,\cdots,N$$

$\rightarrow \alpha = (\alpha_1,\alpha_2,\cdots,\alpha_N)^{\mathrm{T}}$

So we first fix $\alpha_3,\alpha_4,\cdots,\alpha_N$, and suppose the variables are α_1, α_2

Equality constraint: $\alpha_1 = -y_1\sum_{i=2}^{N}\alpha_i y_i$

Contact Me
Email:691582372@qq.com Tel:18936465392

The Sub-Problem of the Dual Problem

$$\min_{\alpha_1,\alpha_2} \quad W(\alpha_1,\alpha_2)=\frac{1}{2}K_{11}\alpha_1^2+\frac{1}{2}K_{22}\alpha_2^2+y_1y_2K_{12}\alpha_1\alpha_2$$

$$-(\alpha_1+\alpha_2)+y_1\alpha_1\sum_{i=3}^{N}y_i\alpha_iK_{i1}+y_2\alpha_2\sum_{i=3}^{N}y_i\alpha_iK_{i2}$$

Dual Problem:

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_jy_iy_jK(x_i,x_j)-\sum_{i=1}^{N}\alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_iy_i=0$$

$$0\leqslant\alpha_i\leqslant C\,, \quad i=1,2,\cdots,N$$

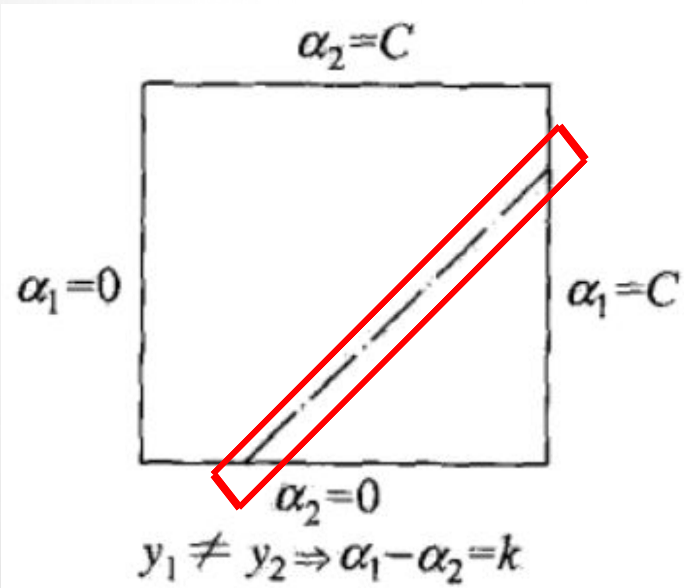
$$\text{s.t.} \quad \alpha_1y_1+\alpha_2y_2=-\sum_{i=3}^{N}y_i\alpha_i=\varsigma$$

$$0\leqslant\alpha_i\leqslant C\,, \quad i=1,2$$

$$K_{ij}=K(x_i,x_j)\,,i,j=1,2,\cdots,N$$

Contact Me
Email:691582372@qq.com Tel:18936465392

constraint condition: $\alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^{N} y_i \alpha_i = \varsigma$



Iteration

$$\alpha_1^{\text{old}}, \alpha_2^{\text{old}} \longrightarrow \alpha_1^{\text{new}}, \alpha_2^{\text{new}}$$

$$\begin{cases} L \leqslant \alpha_2^{\text{new}} \leqslant H \\ L = \max(0, \alpha_2^{\text{old}} - \alpha_1^{\text{old}}) \\ H = \min(C, C + \alpha_2^{\text{old}} - \alpha_1^{\text{old}}) \end{cases}$$

Contact Me
Email:691582372@qq.com Tel:18936465392

The error between the prediction value and true value

$$g(x)=\sum^{N}\alpha_i y_i K(x_i,x)+b$$

$$E_i=g(x_i)-y_i=\left(\sum_{j=1}^{N}\alpha_j y_j K(x_j,x_i)+b\right)-y_i,\quad i=1,2$$

α_2^{old}

$$\boxed{\alpha_2^{\text{new,unc}}=\alpha_2^{\text{old}}+\frac{y_2(E_1-E_2)}{\eta}},\quad \eta=K_{11}+K_{22}-2K_{12}=\|\Phi(x_1)-\Phi(x_2)\|^2$$

$\alpha_2^{\text{new,unc}}$

$$\boxed{\alpha_2^{\text{new}}=\begin{cases}H, & \alpha_2^{\text{new,unc}}>H\\ \alpha_2^{\text{new,unc}}, & L\leqslant\alpha_2^{\text{new,unc}}\leqslant H\\ L, & \alpha_2^{\text{new,unc}}<L\end{cases}}\qquad \alpha_1^{\text{new}}=\alpha_1^{\text{old}}+y_1y_2(\alpha_2^{\text{old}}-\alpha_2^{\text{new}})$$

α_2^{new}

Contact Me
Email:691582372@qq.com Tel:18936465392

An Important Part of the SMO Algorithm

- **How to choose the first variable(outer loop)**
  - Traverse the whole samples that satisfy the $0<\alpha_i<C$,and check whether they satisfy the KTT condition.
  - And choose the example point that most seriously violate the KTT condition as the first variable

- **How to choose the second variable(inner loop)**
  - Traverse the whole samples that satisfy the $0<\alpha_i<C$,and check whether Have the max value of $|E_1-E_2|$.
  - If the value of E1 is positive, choose the smallest value E2 , the according point is the second point. And if the value is negative, choose the biggest.

Contact Me
Email:691582372@qq.com Tel:18936465392

$$
\begin{cases}
\sum_{i=1}^{N} \alpha_i y_i K_{i1} + b = y_1 \\
b_1^{new} = \boxed{y_1 - \sum_{i=3}^{N} \alpha_i y_i K_{i1}} - \alpha_1^{new} y_1 K_{11} - \alpha_2^{new} y_2 K_{21} \\
E_1 = \sum_{i=3}^{N} \alpha_i y_i K_{i1} + \alpha_1^{old} y_1 K_{11} + \alpha_2^{old} y_2 K_{21} + b^{old} - y_1 \\
y_1 - \sum_{i=3}^{N} \alpha_i y_i K_{i1} = -E_1 + \alpha_1^{old} y_1 K_{11} + \alpha_2^{old} y_2 K_{21} + b^{old}
\end{cases}
$$

$$
b_1^{new} = -E_1 - y_1 K_{11} (\alpha_1^{new} - \alpha_1^{old}) - y_2 K_{21} (\alpha_2^{new} - \alpha_2^{old}) + b^{old}
$$

$$
E_i^{new} = \sum_S y_j \alpha_j K(x_i, x_j) + b^{new} - y_i
$$

Contact Me
Email:691582372@qq.com Tel:18936465392

SMO Algorithm

INPUT: Training database: $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

$x_i \in \mathcal{X}=\mathbf{R}^n$, $y_i \in \mathcal{Y}=\{-1,+1\}$, $i=1,2,\cdots,N$

OUTPUT: approximate solution $\hat{\alpha}$

1. Take the initial value $\alpha^{(0)}=0$, and let $k=0$.
2. Choose the optimization variable $\alpha_1^{(k)},\alpha_2^{(k)}$, then get the optimal solution $\alpha_1^{(k+1)},\alpha_2^{(k+1)}$,update the α as $\alpha^{(k+1)}$.
3. If satisfy the following condition

$$\left\{\begin{array}{l}\sum_{i=1}^{N}\alpha_i y_i=0 \\ 0\leqslant\alpha_i\leqslant C, \quad i=1,2,\cdots,N \\ y_i\cdot g(x_i)=\begin{cases}\geqslant 1, & \{x_i \mid \alpha_i=0\} \\ =1, & \{x_i \mid 0<\alpha_i<C\} \\ \leqslant 1, & \{x_i \mid \alpha_i=C\}\end{cases}\end{array}\right.$$

where $g(x_i)=\sum_{j=1}^{N}\alpha_j y_j K(x_j,x_i)+b$.

Jump to 4. Otherwise jump to 2.

4. $\hat{\alpha}=\alpha^{(k+1)}$

Contact Me
Email:691582372@qq.com Tel:18936465392

Multiple Kernel Learning

The objective in MKL :

- Learn kernel parameters
- Learn SVM parameters

Given a set of base kernels $\{K_k\}$ and corresponding feature map $\{\phi_k\}$, linear MKL aims to learn a linear combination of the base kernels as $K = \sum_k d_k K_k$, and usually the kernel weights are restricted to be non-negative .

MKL primal problem :

$$\min_{\mathbf{w},b,\boldsymbol{\xi}\geq\mathbf{0},\mathbf{d}\geq\mathbf{0}} \frac{1}{2}\sum_k \mathbf{w}_k^t\mathbf{w}_k + C\sum_i \xi_i + \boxed{\frac{\lambda}{2}(\sum_k d_k^p)^{\frac{2}{p}}}$$

$$\text{s. t. } y_i(\sum_k \sqrt{d_k}\mathbf{w}_k^t\phi_k(\mathbf{x}_i)+b) \geq 1-\xi_i$$

The regularization on the kernel weights is necessary to prevent them from shooting off to infinity

Contact Me
Email:691582372@qq.com Tel:18936465392

If we substituting $\mathbf{w}_k$ for $\sqrt{d_k}\mathbf{w}_k$

$$\min_{\mathbf{w},b,\boldsymbol{\xi}\geq\mathbf{0},\mathbf{d}\geq\mathbf{0}} \tfrac{1}{2}\sum_k \mathbf{w}_k^t\mathbf{w}_k/d_k + C\sum_i \xi_i + \frac{\lambda}{2}(\sum_k d_k^p)^{\frac{2}{p}}$$

$$\text{s. t. } y_i(\sum_k \mathbf{w}_k^t\phi_k(\mathbf{x}_i)+b) \geq 1-\xi_i$$

The Lagrange Function:

$$L = \tfrac{1}{2}\sum_k \mathbf{w}_k^t\mathbf{w}_k/d_k + \sum_i (C-\beta_i)\xi_i + \frac{\lambda}{2}(\sum_k d_k^p)^{\frac{2}{p}} - \sum_i \alpha_i[y_i(\sum_k \mathbf{w}_k^t\phi_k(\mathbf{x}_i)+b)-1+\xi_i]$$

Differentiating with respect to w, b and ξ to get the optimality conditions and substituting back results in the following intermediate saddle point problem.

$$\min_{\mathbf{d}\geq\mathbf{0}} \max_{\boldsymbol{\alpha}\in\mathcal{A}} \mathbf{1}^t\boldsymbol{\alpha} - \tfrac{1}{2}\sum_k d_k\boldsymbol{\alpha}^t H_k\boldsymbol{\alpha} + \frac{\lambda}{2}(\sum_k d_k^p)^{\frac{2}{p}}$$

$$\text{where } \mathcal{A} = \{\boldsymbol{\alpha}|0 \leq \boldsymbol{\alpha} \leq C\mathbf{1}, \mathbf{1}^t Y\boldsymbol{\alpha} = 0\}, H_k = YK_kY$$

PS: Y is a diagonal matrix with the labels on the diagonal.

Contact Me
Email:691582372@qq.com Tel:18936465392

$$L = \mathbf{1}^t\boldsymbol{\alpha} - \tfrac{1}{2}\sum_k d_k \boldsymbol{\alpha}^t H_k \boldsymbol{\alpha} + \frac{\lambda}{2}\Big(\sum_k d_k^p\Big)^{\frac{2}{p}} - \underline{\sum_k \gamma_k d_k}$$

$$\frac{\partial L}{\partial d_k} = 0 \Rightarrow \lambda\Big(\sum_k d_k^p\Big)^{\frac{2}{p}-1} d_k^{p-1} = \gamma_k + \tfrac{1}{2}\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha}$$

To eliminate d

$$\Rightarrow \lambda\Big(\sum_k d_k^p\Big)^{\frac{2}{p}} = \sum_k d_k\big(\gamma_k + \tfrac{1}{2}\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha}\big)$$

$$\Rightarrow L = \mathbf{1}^t\boldsymbol{\alpha} - \frac{\lambda}{2}\Big(\sum_k d_k^p\Big)^{\frac{2}{p}} = \mathbf{1}^t\boldsymbol{\alpha} - \frac{1}{2\lambda}\Big(\sum_k (\boxed{\gamma_k} + \boxed{\tfrac{1}{2}\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha}})^q\Big)^{\frac{2}{q}} \qquad \text{where } \frac{1}{p} + \frac{1}{q} = 1$$

$\gamma_k \geq 0$ $\qquad$ H_k is positive semi-definite, $\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha} \geq 0$

Our lp-MKL dual problem:

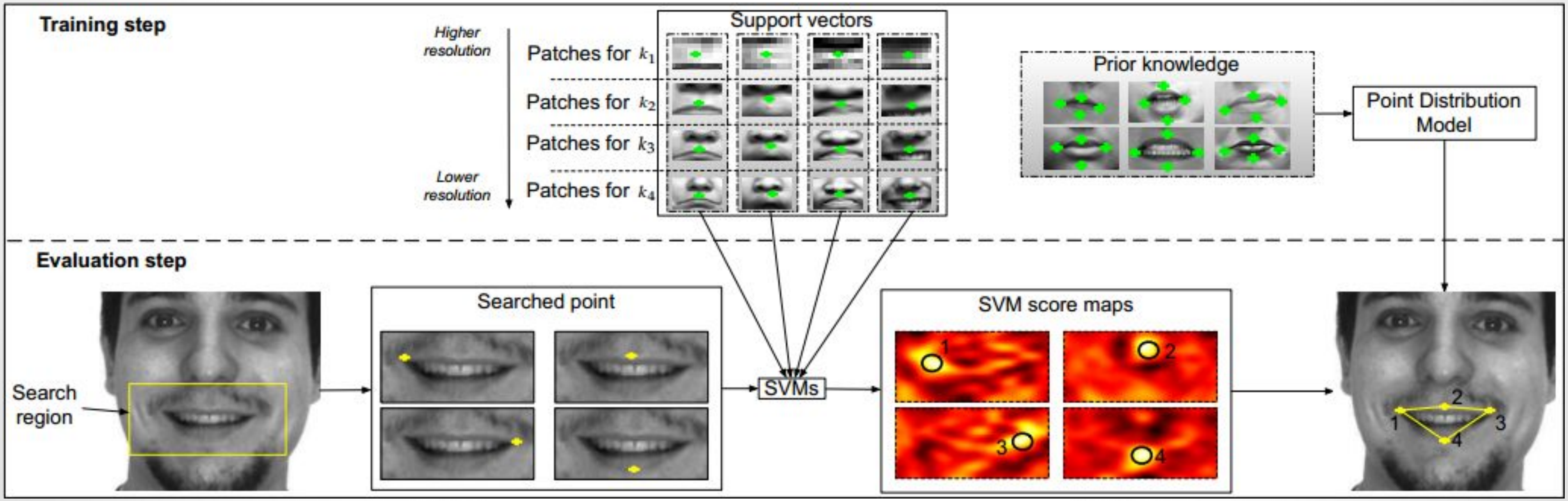
$$D \equiv \max_{\boldsymbol{\alpha}\in\mathcal{A}} \mathbf{1}^t\boldsymbol{\alpha} - \frac{1}{8\lambda}\Big(\sum_k (\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha})^q\Big)^{\frac{2}{q}}$$

where $\mathcal{A} = \{\boldsymbol{\alpha} | \mathbf{0} \leq \boldsymbol{\alpha} \leq C\mathbf{1}, \mathbf{1}^t Y\boldsymbol{\alpha} = 0\}$, $H_k = YK_kY$

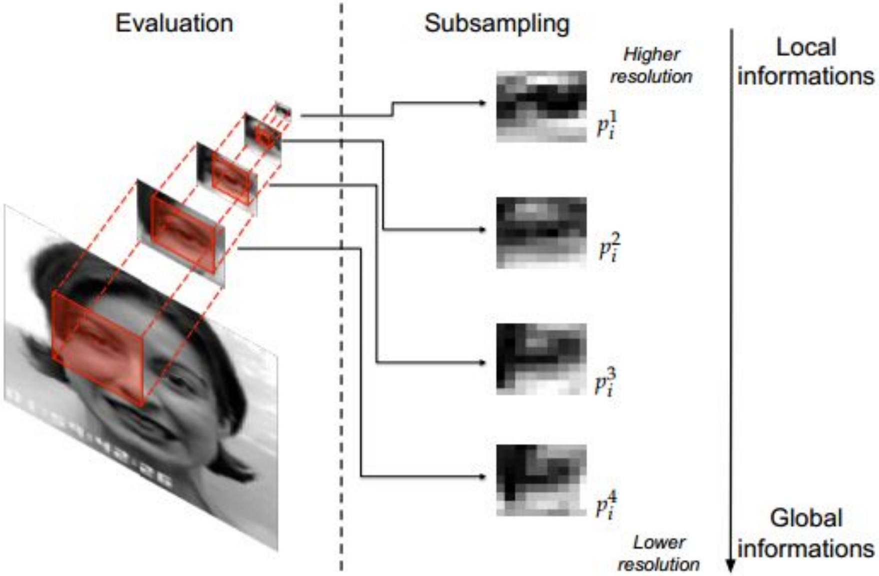
$$\rightarrow \quad d_k = \frac{1}{2\lambda}\left(\sum_k (\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha})^q\right)^{\frac{1}{q}-\frac{1}{p}} (\boldsymbol{\alpha}^t H_k \boldsymbol{\alpha})^{\frac{q}{p}}$$

Contact Me
Email:691582372@qq.com Tel:18936465392

Multiple Kernel Learning SVM for Facial Landmark Detection



Contact Me
Email:691582372@qq.com Tel:18936465392

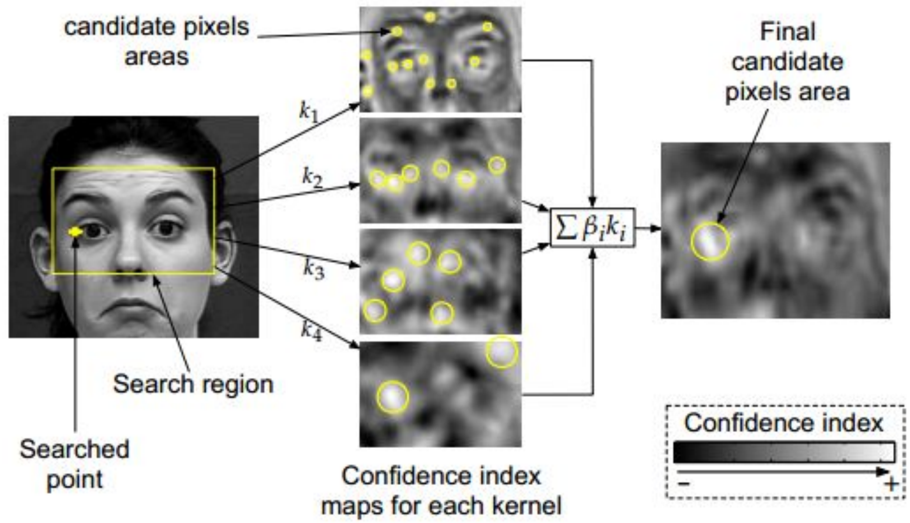


In this paper, we use multi-resolution patches extracting different level of information. For a pixel i, we take the first patch (p_i^1) large enough to encode plenty of general information.

The other patches $(p_i^2, p_i^3, \ldots, p_i^N)$ are extracted cropping a progressively smaller area giving increasingly detailed information.

Thus, high resolution patches encode local information and small details, such as canthus or pupil location, around the point. Low resolution patches, on the other hand, encode global information.

Contact Me
Email:691582372@qq.com Tel:18936465392

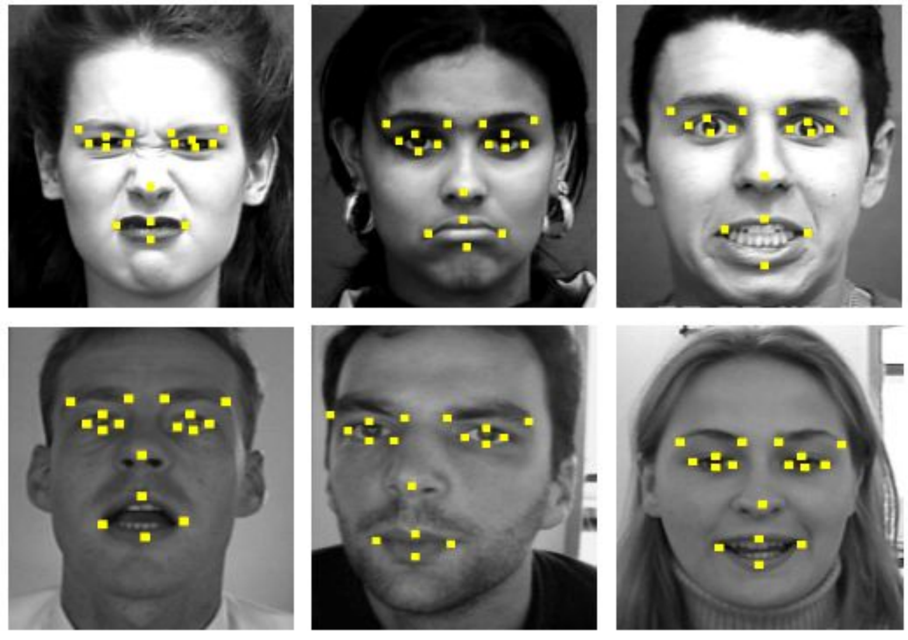


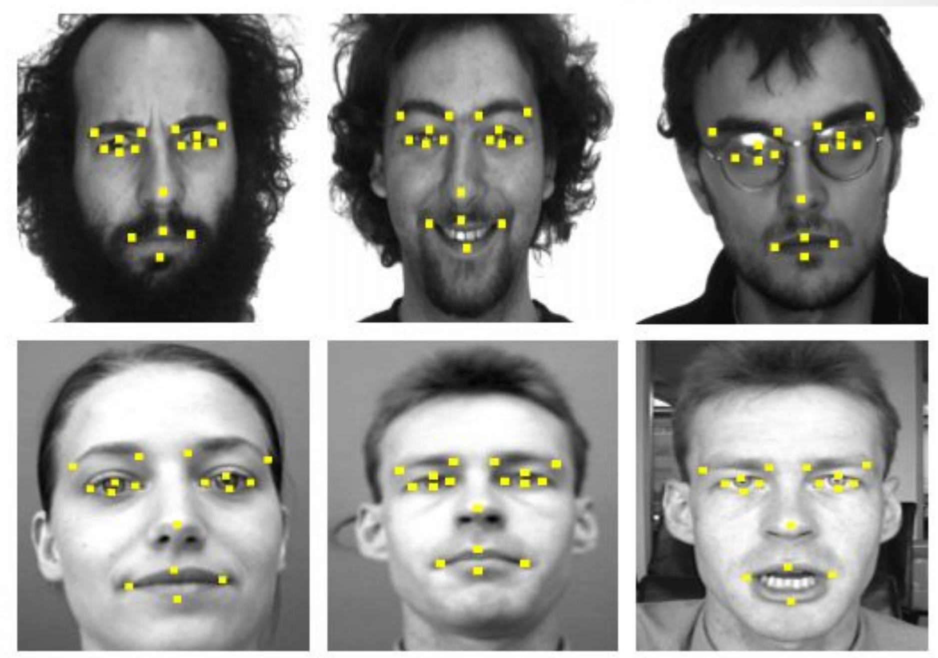
1) Training Step: Given $x_i = (p_i^1, ..., p_i^N)$ a training set of m samples associated with labels $y_i \in \{-1, 1\}$ (target or non-target), the classification function of the SVM associates a score s to a new sample (or candidate pixel) $x = (p_i^1, ..., p_i^N)$

$$s = \left(\sum_{i=1}^{m} \alpha_i k(x_i, x) + b \right) \quad (1)$$

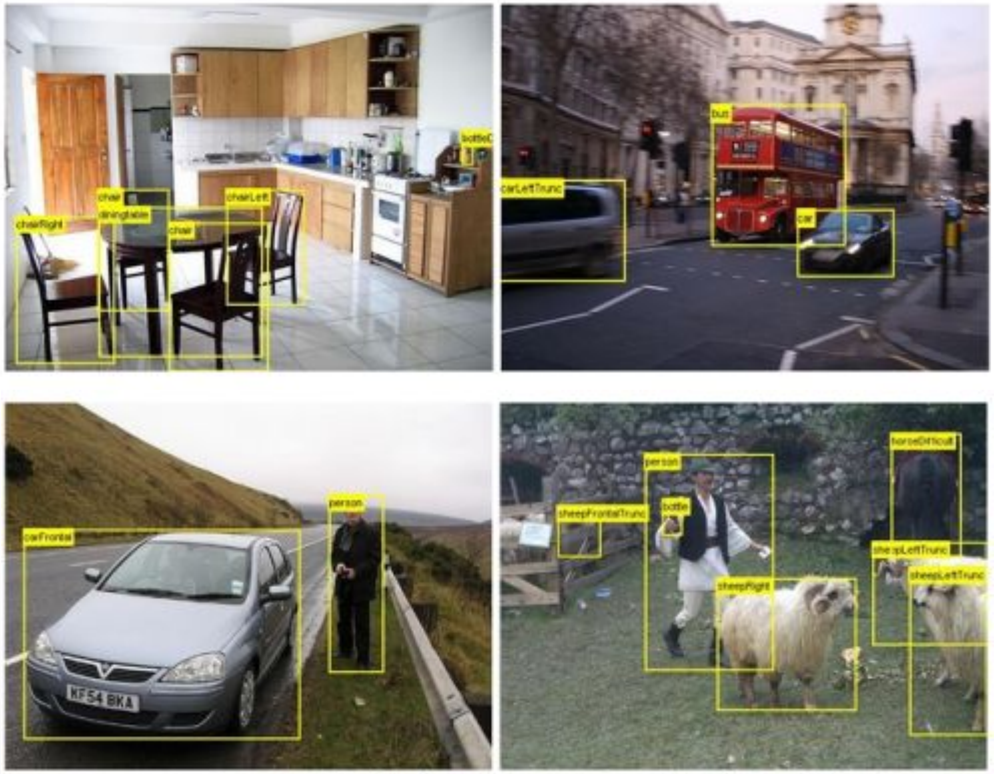
$$k(x_i, x) = \sum_{j=1}^{K} \beta_j k_j$$

$$\text{with } \beta_j \geq 0, \sum_{j=1}^{K} \beta_j = 1$$





Contact Me
Email:691582372@qq.com Tel:18936465392



the system learns its parameters from a set of training images

$I^i,\ i = 1, \ldots, N$

with known locations

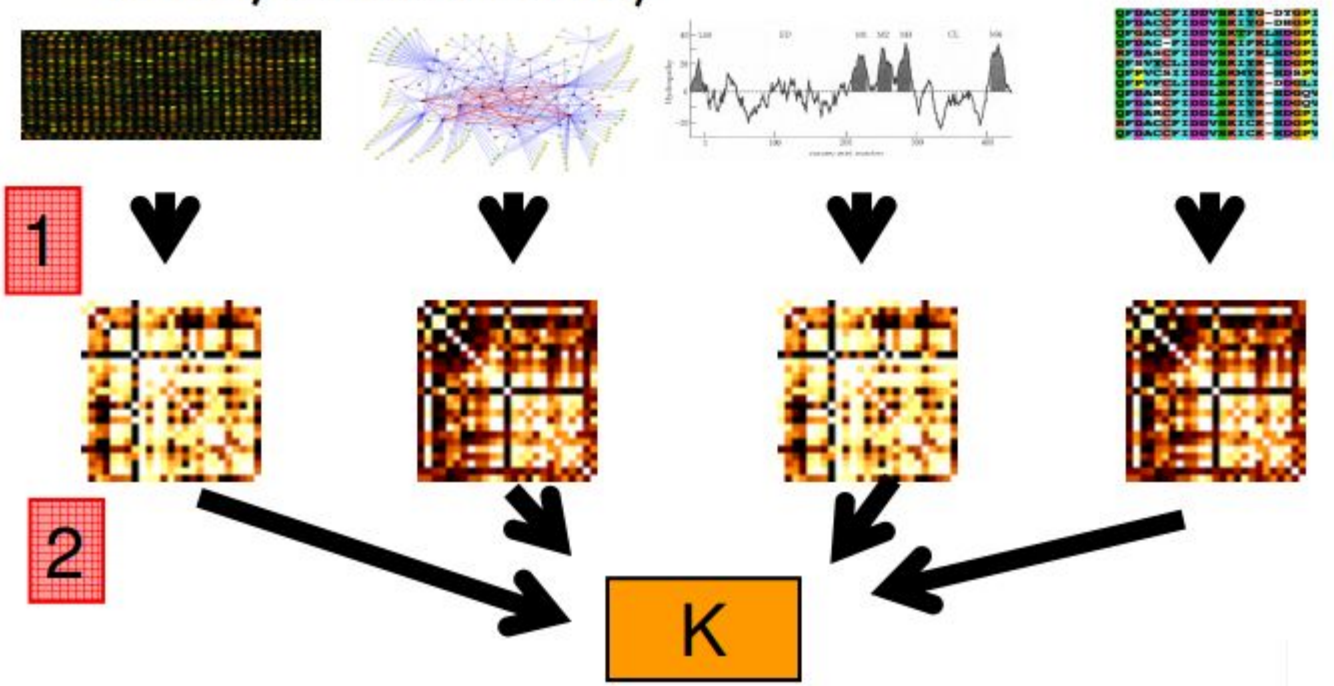
$l_1^i, \ldots, l_{n_i}^i$

class labels for the ni objects present in I^i

$$f : \mathcal{I} \times \overbrace{\mathcal{L} \times \cdots \times \mathcal{L}}^{K \text{ times}} \to \mathbb{R}^K$$

Contact Me
Email:691582372@qq.com Tel:18936465392

- Create individual kernels for each source (string kernel, diffusion kernel)



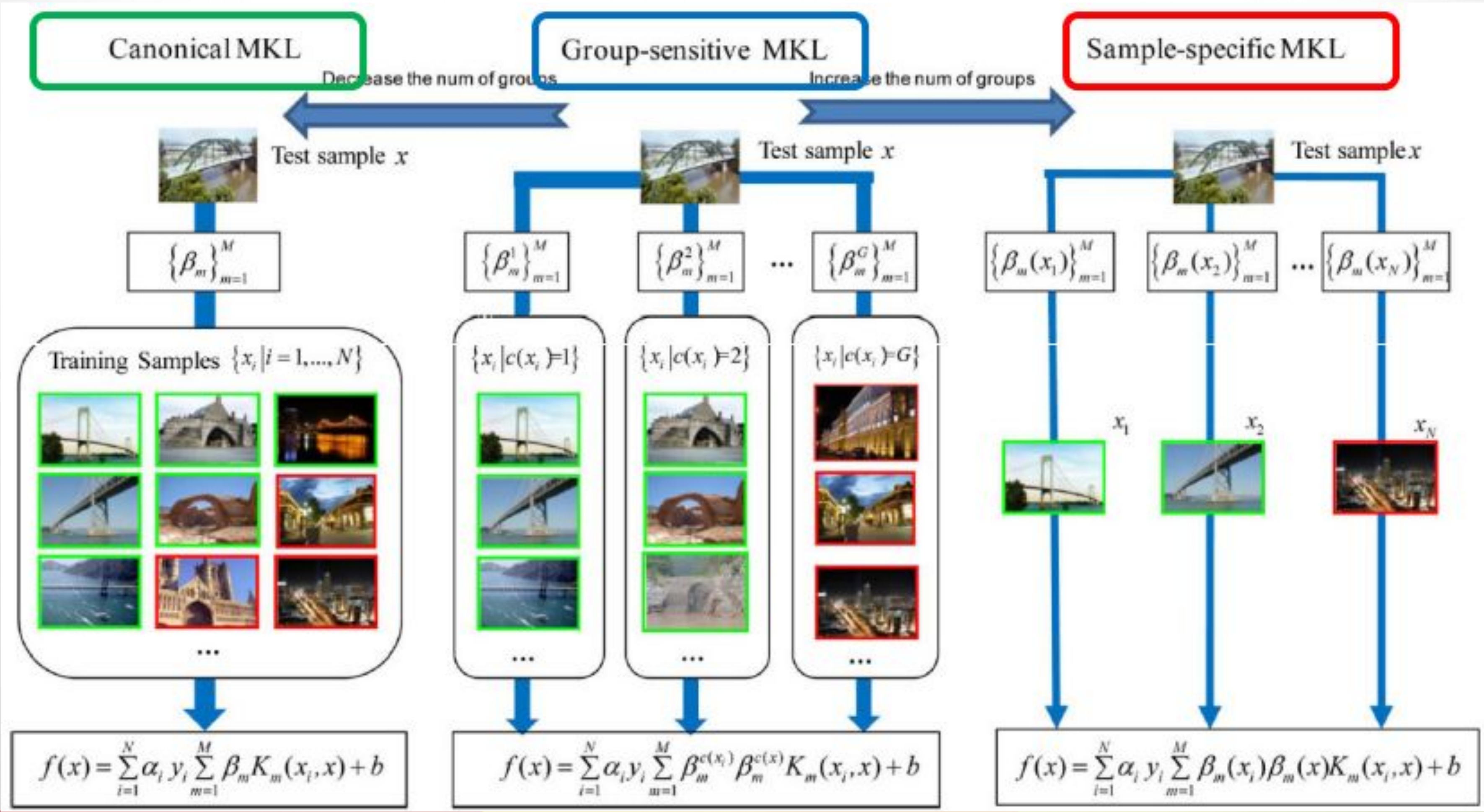
Contact Me
Email:691582372@qq.com
Tel:18936465392

- The overall MKL framework:

1. Extract features from all available sources
2. Construct kernel matrices
    1. Different features
    2. Different kernel types
    3. Different kernel parameters
3. Find the optimal kernel combination and the kernel classifier

Contact Me
Email:691582372@qq.com Tel:18936465392

Non-stationary MKL



Contact Me
Email:691582372@qq.com Tel:18936465392

Thanks for listening…

Contact Us
Lorem ipsum dolor, 03663, State, Country / P. 123 456 7890 / email@domain.com