# Introduction to Kernel methods

Chiranjib Bhattacharyya

Machine Learning lab
Dept of CSA, IISc
chiru@csa.iisc.ernet.in
http://drona.csa.iisc.ernet.in/~chiru

13th June, 2011

# Agenda

1. Kernel Trick
   - SVM and Non-linear Classification
2. Definition of Kernel functions
3. Kernels and Hilbert Spaces
   - RKHS, Representer theorem etc
4. Kernels on Different kinds of data
   - Kernels on Strings
   - Kernels on generative models
   - Kernels on Graphs
5. Advanced Topics
   - Multiple Kernel Learning
   - Learning Kernels from Similarity functions

# Introduction

- Data: Protein structures, Images, videos,
  Algorithms: Classifiers, PCA
  Most algorithms may require vectorial description which maynot be easily available. Kernels can help bridge the gap.
- Kernels are similarity measures
- Kernels can help integrate different sources of data

# PART 1: KERNEL TRICK

# The problem of classification

Let $D = \{(x_i, y_i) | i = 1, \ldots, m\}$ be the training dataset. where $y_i$ is the class label of an observation $x_i$

Assume that $y \in \{-1, 1\}$.

A classifier $f : \mathscr{X} \to \{-1, 1\}$. A linear classifier is of the form

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

Learning classifiers on a given dataset $D$ which has good *generalization* abilities is one of the important problems in Machine Learning.

# Review of C-SVM

$$min_{\mathbf{w},b} C \sum_{i=1}^{m} max(1 - y_i(\mathbf{w}^\top x_i + b), 0) + \frac{1}{2}\|w\|^2$$

## C-SVM formulation

$$\text{maximize}_\alpha \; -\frac{1}{2}\sum_{ij} \alpha_i \alpha_j y_i y_j x_i^\top x_j + \sum_{i=1}^{m} \alpha_i$$

$$\text{subject to } \; 0 \leq \alpha_i, \sum_i \alpha_i y_i = 0$$

At optimality $\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i x_i$

$$f(x) = sign(\sum_{i=1}^{m} \alpha_i y_i x_i^\top x + b)$$

# C-SVM in feature spaces

Let us work with a feature map, $\Phi(x)$ Solving a SVM problem in feature space turns out to be

$$\text{maximize}_\alpha \; -\frac{1}{2}\sum_{ij}\alpha_i\alpha_j y_i y_j \Phi(x_i)^\top \Phi(x_j) + \sum_{i=1}^{m}\alpha_i$$

$$\text{subject to } \; 0 \leq \alpha_i, \sum_i \alpha_i y_i = 0$$

and our classifier is

$$f(x) = sign(\sum_{i=1}^{m}\alpha_i y_i \Phi(x_i)^\top \Phi(x) + b)$$

The dot product between any pair of examples computed in the feature space be denoted by

$$K(x,z) = \Phi(x)^\top \Phi(z)$$

# C-SVM in feature spaces

Let us work with a feature map, $\Phi(x)$ Solving a SVM problem in feature space turns out to be

$$\text{maximize}_\alpha \; -\frac{1}{2}\sum_{ij}\alpha_i\alpha_j y_i y_j K(x_i,x_j) + \sum_{i=1}^{m}\alpha_i$$

$$\text{subject to } \; 0 \leq \alpha_i, \sum_i \alpha_i y_i = 0$$

and our classifier is

$$f(x) = sign(\sum_{i=1}^{m}\alpha_i y_i K(x_i,x) + b)$$

The dot product between any pair of examples computed in the feature space be denoted by

$$K(x,z) = \Phi(x)^\top \Phi(z)$$

# Kernel Fisher Discriminant(Mika et al. 1999)

Let $X = [x_1, \ldots, x_{m_+}]$ and $Z = [z_1, \ldots, z_{m_-}]$ where $x_i, z_j \in \mathbb{R}^d$

$$max_w \frac{\{w^\top (\mu_1 - \mu_2)\}^2}{w^\top S_w w}$$

$$\mu_1 = \frac{1}{m_+} \sum_i x_i = \frac{1}{m_+} Xe \; \mu_2 = \frac{1}{m_-} \sum_i z_i = \frac{1}{m_-} Ze$$

($e$ is a vector of ones)

$$S_w = \frac{1}{m_1} \sum_{i=1}^{m_+} (x_i - \mu_+)(x_i - \mu_+)^\top + \frac{1}{m_-} \sum_{j=1}^{m_-} (z_j - \mu_-)(z_j - \mu_-)^\top$$

$\forall w \in \mathbb{R}^d$ there exists unique $w_\perp$ such that

$$w = [X\, Z]\alpha + w_\perp \;\; w_\perp^\top x_i = w_\perp^\top z_j = 0$$

# Kernel Fisher Discriminant(Mika et al. 1999)

$$w^\top \mu_+ = \alpha^\top [X^\top X \, Z^\top X] e \; w^\top \mu_- = \alpha^\top [X^\top Z \, Z^\top Z] e$$

$$(X^\top Z)_{ij} = x_i^\top z_j$$

can be kernelized, i.e. $x_i^\top z_j$ can be replaced by kernel functions $K(x_i, z_j)$.

# Kernel Fisher Discriminant(Mika et al. 1999)

$$w^\top \mu_+ = \alpha^\top [X^\top X \, Z^\top X] e \; w^\top \mu_- = \alpha^\top [X^\top Z \, Z^\top Z] e$$

$$(X^\top Z)_{ij} = x_i^\top z_j$$

can be kernelized, i.e. $x_i^\top z_j$ can be replaced by kernel functions $K(x_i, z_j)$. It is possible to compute Fisher Discriminant in feature space

$$max_\alpha \frac{\{\alpha^\top (k_1 - k_2)\}^2}{\alpha^\top C \alpha}$$

where $k_1, k_2$ and $C$ are suitably defined.

# Principal Component Analysis(PCA)

Let $X = [x_1, \ldots, x_m]$. Finding directions of maximum variance can be very informative(see Jollife 2002).
Using sample covariance the direction of maximum variance, $v$, is given by

$$\frac{1}{m} XX^\top v = \lambda v$$

(assuming that $Xe = 0$)
Let $\alpha = X^\top v$ then $v = \frac{1}{\lambda} \sum_{i=1}^{m} \alpha_i x_i$ indicating that $v$ is in the linear span of $\{x_1, \ldots, x_m\}$.

$$\frac{1}{m} XX^\top X\alpha = \lambda X\alpha$$

leading to the following eigenvalue problem

$$\frac{1}{m} \mathbf{K}\alpha = \lambda \alpha$$

where $(\mathbf{K})_{ij} = (X^\top X)_{ij} = x_i^\top x_j$. Replacing $x_i^\top x_j$ by $\Phi(x_i)^\top \Phi(x_j)$ one can do PCA in feature spaces (Scholkopf et al. 1996)

# An example

- Let $x \in \mathbb{R}^2$ and $\Phi(x) = [x_1^2 \; x_2^2 \; \sqrt{2}x_1x_2]^\top$

$$K(x,z) = \Phi(x)^\top \Phi(z) = x_1^2 z_1^2 + 2x_1x_2z_1z_2 + x_2^2 z_2^2 = < x,z >^2$$

- If $K(x,z) = (x^\top z)^r$ is a dot product in a $\binom{d+r-1}{r}$ feature space corresponding to $x, z \in \mathbb{R}^d$.
- If $d = 256, r = 4$, the feature space size is $6,35,376$.
- However if we know $K$ one can still solve the SVM formulation without explicitly evaluating $\Phi$

# In the sequel

- conditions on $K$ will be discussed
- $K$ for various data-types
- Advanced topics in Kernel design

# Kernel function

## Kernel function

$K : \times \mathscr{X} \rightarrow \mathbb{R}$ is a Kernel function if

$$K(x,z) = K(z,x) \text{ symmetric}$$

$$K \text{is positive semidefinite, i.e.} \forall n, x_1, \ldots, x_n \in \mathscr{X},$$

the matrix $K_{ij} = K(x_i, x_j)$ is psd

Recall that a $\mathbf{K} \in \mathbb{R}^{d \times d}$ is psd if $u^\top \mathbf{K} u \geq 0$ for all $u \in \mathbb{R}^d$.

# Examples of Kernel function

$$K(x,z) = \phi(x)^\top \phi(z) \text{ where } \phi : \mathscr{E} \to \mathbb{R}^d$$

$K$ is symmetric i.e. $K(x,z) = K(z,x)$

# Examples of Kernel function

$$K(x,z) = \phi(x)^\top \phi(z) \text{ where } \phi : \mathscr{E} \to \mathbb{R}^d$$

$K$ is symmetric i.e. $K(x,z) = K(z,x)$
Positive Semidefinite:
Let $D = \{x_1, x_2, \ldots, x_n\}$ be set of arbitrarily chosen $n$ elements of $\mathscr{E}$.
Define

$$\mathbf{K}_{ij} = \phi(x_i)^\top \phi(x_j)$$

For any $u \in \mathbb{R}^n$ it is straightforward to see that

$$u^\top \mathbf{K} u = \|\phi(D)u\|_2^2 \geq 0 \;\; \phi(D) = [\phi(x_1), \ldots, \phi(x_n)]$$

# Examples of Kernel functions

$$K(x,z) = x^\top z \qquad \Phi(x) = x$$

$$K(x,z) = (x^\top z)^r \qquad \Phi_{t_1 t_2 \ldots t_d}(x) = \sqrt{\frac{r!}{t_1! t_2! \ldots . t_d!}} x_1^{t_1} x_2^{t_2} \ldots x_d^{t_d}$$

$$\sum_{i=1}^{d} t_i = r$$

$$K(x,z) = e^{-\gamma \|x-z\|^2}$$

# Kernel Construction

Let $K_1$ and $K_2$ be two valid kernels.

$$K(x,y) = \phi(x)^\top \phi(y)$$

$$K(u,v) = K_1(u,v)K_2(u,v)$$

$$K = \alpha K_1 + \beta K_2 \quad \alpha,\beta \geq 0$$

$$\hat{K}(x,y) = \frac{K(x,y)}{\sqrt{K(x,x)}\sqrt{K(y,y)}}$$

# Kernel Construction

Let $K_1$ and $K_2$ be two valid kernels.

$$K(x,y) = \phi(x)^\top \phi(y)$$

$$K(u,v) = K_1(u,v)K_2(u,v)$$

$$K = \alpha K_1 + \beta K_2 \quad \alpha,\beta \geq 0$$

$$\hat{K}(x,y) = \frac{K(x,y)}{\sqrt{K(x,x)}\sqrt{K(y,y)}}$$

$$K(x,y) = x^\top y$$

$$K(x,y) = (x^\top y)^i$$

$$K(x,y) = \lim_{N\to\infty} \sum_{i=0}^{N} \frac{(x^\top y)^i}{i!} = e^{x^\top y}$$

$$\hat{K}(x,y) = e^{-\frac{1}{2}\|x-y\|^2}$$

# Kernel function and feature map

A theorem due to Mercer guarantees a feature map for symmetric, psd kernel functions.
Loosely stated
For a symmetric kernel $K : \mathscr{X} \times \mathscr{X} \rightarrow \mathbb{R}$, there exists an expansion $K(x,z) = \Phi(x)^\top \Phi(z)$ iff

$$\int_{\mathscr{X}} g(x)g(z)K(x,z)dxdz \geq 0$$

# PART 3: Kernels and Hilbert spaces

# What is a Dot product(aka Inner Product)

Let $\mathscr{X}$ be a vector space.

## What is a Dot product

$$\text{Symmetry} \quad < u,v >=< v,u > \quad u,v \in \mathscr{X}$$

$$\text{Bilinear} \quad < \alpha u+\beta v,w >= \alpha < u,w > +\beta < v,w > \quad u,v,w, \in \mathscr{X}$$

$$\text{Positive Semidefinite} \quad < u,u > \geq 0 \; u \in \mathscr{X}$$

$$< u,u >= 0 \text{ iff } u = 0$$

## Norm

$$\|x\| = \sqrt{\langle x,x \rangle}$$

$$\|x\| = 0 \implies x = 0$$

# Examples of Dot products

$$\mathscr{X} = \mathbb{R}^n, < u, v >= u^\top v$$

$$\mathscr{X} = \mathbb{R}^n, < u, v >= \sum_{i=1}^{n} \lambda_i u_i v_i \ \lambda_i \geq 0$$

$$\mathscr{X} = L_2(X) = \{f : \int_X f(x)^2 dx < \infty\}$$

$$f, g \in \mathscr{X} \ < f, g >= \int_X f(x) g(x) dx$$

# Cauchy Schwartz inequality

## Cauchy Schwartz inequality

Let $\mathscr{X}$ be an *inner product space*.

$$|\langle x, y\rangle| \leq \|x\|\|y\| \; \forall \, x, y \in \mathscr{X}$$

and equality holds iff $x = \alpha z$ for some scalar $\alpha$

**Proof:** $\forall \alpha \in \mathbb{R} \; \|x - \alpha z\|^2 \geq 0$

$$\|x\|^2 - 2\alpha\langle x, z\rangle + \alpha^2\|z\|^2 \geq 0 \forall \alpha$$

Let $\alpha = \frac{\langle x,z\rangle}{\|z\|^2}$ and the inequality follows by taking square roots. The claim about equality follows from the definition of norm.

# Hilbert Space: Basic facts

Defn: A Inner product space $(\mathscr{H}, \langle \cdot, \cdot \rangle_{\mathscr{H}})$ is a Hilbert Space if it is separable and complete.
We will denote the norm as $\| \cdot \|_{\mathscr{H}}$. The orthogonal complement of $M$, where $M \subset \mathscr{H}$ be a subspace of $\mathscr{H}$ is defined as
$M^{\perp} = \{z | \langle x, z \rangle_{\mathscr{H}} = 0, \ \forall x \in M\}$

## Hilbert space Projection theorem

Let $M$ be a subspace of Hilbert space $\mathscr{H}, \langle \cdot, \cdot \rangle_{\mathscr{H}}$. For every $x \in \mathscr{H}$ the following holds

- There exists an unique $\Pi_M(x) \in M$ such that $\Pi_M(x) = \text{argmin}_{z \in M} \|x - z\|_{\mathscr{H}}$
- $x - \Pi_M(x) \in M^{\perp} \ \langle z, x - \Pi_M(x) \rangle_{\mathscr{H}} = 0 \ \forall z \in M$
- $\|x\|^2_{\mathscr{H}} = \|\Pi_M(x)\|^2_{\mathscr{H}} + \|y\|^2_{\mathscr{H}}$ where

$$x = \Pi_M(x) + y \text{ where } y \in M^{\perp}$$

# Reproducing kernel Hilbert Space(RKHS)

Let $K$ be any kernel function. Consider the following set

$$\mathscr{H} = \{f | f(.) = \sum_{i=1}^{m} \alpha_i K(.,x_i) \forall x_i \in \mathscr{X}, m \in N\}$$

## Dot product

For any $f, g \in \mathscr{H}$,

$$f(.) = \sum_{i=1}^{m_1} \alpha_i K(.,x_i) \ , \ g(.) = \sum_{i=1}^{m_2} \beta_j K(.,x_j)$$

$$\langle f, g \rangle_{\mathscr{H}} = \sum_{i=1}^{m_1} \sum_{j=1}^{m_2} \alpha_i \beta_j K(x_i, x_j)$$

Is it a dot product?

# Reproducing kernel Hilbert Space(RKHS)

As $K$ is symmetric, $\langle f, g \rangle_{\mathscr{H}} = \langle g, f \rangle_{\mathscr{H}}$

$$\langle f(.), f(.) \rangle = \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j K(x_i, x_j)$$

Recall that $\mathbf{K}$ is a psd matrix if $K$ is kernel function and so

$$\langle f(.), f(.) \rangle_{\mathscr{H}} \geq 0$$

## Reproducible Property

for any $f \in \mathscr{H}$

$$f(x) = \sum_{i=i}^{m} \alpha_i K(x, x_i) = \langle \sum_{i=1}^{m} \alpha_i K(., x_i), K(., x) \rangle = \langle f(.), K(., x) \rangle$$

Applying C-S inequality

$$|f(x)| \leq \sqrt{\langle f, f \rangle_{\mathscr{H}}} \sqrt{K(x, x)}$$

# Representer theorem

## Representer theorem

Let $K$ be a valid kernel defined on $\mathscr{X}$ and $\mathscr{H}$ be the corresponding RKHS. Let $\Omega$ be an increasing function. The optimization problem

$$\min_{g \in \mathscr{H}} G(g) = \sum_{i=1}^{m} l(g(x_i), y_i) + \Omega(\|g\|^2_{\mathscr{H}})$$

is solved when $g^* = \sum_{i=1}^{m} \alpha_i K(., x_i)$

# Representer theorem

## Representer theorem

Let $K$ be a valid kernel defined on $\mathscr{X}$ and $\mathscr{H}$ be the corresponding RKHS. Let $\Omega$ be an increasing function. The optimization problem

$$\min_{g \in \mathscr{H}} G(g) = \sum_{i=1}^{m} l(g(x_i), y_i) + \Omega(\|g\|^2_{\mathscr{H}})$$

is solved when $g^* = \sum_{i=1}^{m} \alpha_i K(.,x_i)$

Proof: Let $M = \{\sum_{i=1}^{m} \alpha_i K(.,x_i)\ i = 1,\ldots,m\}$. Clearly $M$ is a subspace of $\mathscr{H}$. Take any $g \in \mathscr{H}$.

$$g(x_i) = \langle g, K(.,x_i)\rangle = \langle g_M + g_{per}, K(.,x_i)\rangle$$

$$= \langle g_M, K(.,x_i)\rangle + \langle g_{per}, K(.,x_i)\rangle = \langle g_M, K(.,x_i)\rangle = g_M(x_i)$$

As $\Omega$ is an increasing function, $\Omega(\|g\|^2_{\mathscr{H}}) \geq \Omega(\|g_M\|^2_{\mathscr{H}})$

# Back to C-SVM formulation

Given a Kernel function $K$ defined on $\mathscr{X}$ one can create a RKHS

$$\mathscr{H} = \{\sum_{i=1}^{n} \beta_i z_i | z_i \in \mathscr{X}, n \in N\}$$

The C-SVM problem can be posed as

$$min_{g \in \mathscr{H}, b \in \mathbb{R}} i \sum_{i=1}^{m} \underbrace{max(0, 1 - y_i(g(x_i) + b))}_{l(g(x_i), y_i)} + \|g\|^2_{\mathscr{H}}$$

Classifier: $f(x) = sign(g(x) + b)$

# Kernel C-SVM

- Using the representer theorem one can deduce that at optimality $g(.) = \sum_{i=1}^{m} \gamma_i K(., x_i)$
- Plugging this into the optimization problem mentioned in the previous slide recovers the C-SVM formulation in the feature space.
- The Trace of the Kernel matrix plays an important role in the generalization abilities

# Computation in feature spaces using dot products

$$K(x,z) = \langle \Phi(x), \Phi(z) \rangle \quad \|\Phi(x)\| = \sqrt{\langle \Phi(x), \Phi(x) \rangle} = \sqrt{K(x,x)}$$

## Normalized Kernel

$$\hat{\Phi}(x) = \frac{\Phi(x)}{\|\Phi(x)\|} \quad \hat{K}(x,z) = \langle \hat{\Phi}(x)^\top \hat{\Phi}(z) \frac{K(x,z)}{\sqrt{K(x,x)K(z,z)}}$$

Distance between feature vectors

$$\|\Phi(x) - \Phi(z)\|^2 = \langle \Phi(x) - \Phi(z), \Phi(x) - \Phi(z) \rangle$$

$$= K(x,x) + K(z,z) - 2K(x,z)$$

If $\Phi$ is normalized $K(x,x) = 1$ then $\|\Phi(x) - \Phi(z)\|^2 = 2 - 2K(x,z)$
$K(x,z)$ can be understood as a measure of similarity between $x$ and $z$.

# Computation in feature spaces using dot products

Computing norms of linear combination of feature maps

$$\|\sum_{i=1}^{m} \alpha_i \Phi(x_i)\|^2 = \langle \sum_{i=1}^{m} \alpha_i \Phi(x_i), \sum_{j=1}^{m} \alpha_j \Phi(x_j) \rangle = \sum_{i=1,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j)$$

Let $\bar{\Phi} = \frac{1}{m} \sum_{i=1}^{m} \Phi(x_i)$.

$$\|\bar{\Phi}\|^2 = \langle \bar{\Phi}, \bar{\Phi} \rangle = \frac{1}{m^2} \sum_{i,j=1}^{m} K(x_i, x_j)$$

$$\Phi_{new}(x) = \Phi(x) - \bar{\Phi}$$

$$K_{new}(x,z) = K(x,z) - \frac{1}{m} \sum_{i=1}^{m} K(x, x_i) - \frac{1}{m} \sum_{i=1}^{m} K(z, x_i) + \frac{1}{m^2} \sum_{i,j=1}^{m} K(x_i, x_j)$$

# Computation in feature spaces using dot products

Assume that

$$S_1 = \{x_1, \ldots, x_{m_+}\} \text{ positive class}$$

$$S_2 = \{z_1, \ldots, z_{m_-}\} \text{ negative class}$$

Let $\bar{\Phi}_1$ be defined on $S_1$ and $\bar{\Phi}_2$ be defined on $S_2$

$$f(x) = sign(\|\Phi(x) - \bar{\Phi}_1\|^2 - \|\Phi(x) - \bar{\Phi}_2\|^2)$$

# Computation in feature spaces using dot products

Assume that

$$S_1 = \{x_1, \ldots, x_{m_+}\} \text{ positive class}$$

$$S_2 = \{z_1, \ldots, z_{m_-}\} \text{ negative class}$$

Let $\bar{\Phi}_1$ be defined on $S_1$ and $\bar{\Phi}_2$ be defined on $S_2$

$$f(x) = sign(\|\Phi(x) - \bar{\Phi}_1\|^2 - \|\Phi(x) - \bar{\Phi}_2\|^2)$$

$$= sign(\frac{1}{m_+}\sum_{i=1}^{m_+} K(x, x_i) - \frac{1}{m_-}\sum_{i=1}^{m_-} K(x, z_i) - b)$$

# Summary

- reviewed the notion of dot products
- seen examples of how one can do computations in feature space
- reviewed kernel functions

# PART 4: Kernels on Different Data types

# Similarity between Strings

- Impetus from Biology
- *Insertion*, *Deletion* and *Substitution*

Let $x =$ singapore and $y =$ ingpore

s i n g a p o r e
- i n g - p o r e

Global alignment Needleman Wunsch $O(|x||y|)$

p o r e

is the largest common subsequence.
Local alignment
Smith Waterman $O(|x||y|)$

Both algorithms gives a score, which is used heavily in computational biology.

# Probabilistic approaches for modelling sequences

- Probabilistic approaches can be an interesting alternative (Durbin et al. 2000)
- HMMs have been used very successfully for modelling sequences
- A very special HMM model called Profile HMM was invented for probabilistically aligning multiple sequences

# Spectrum Kernels(Leslie et al. 2002)

- Consider all $k-$ length contiguous subsequences that it contains
- Create a feature map indexed by all possible substrings (”k-mers”) from the alphabet of amino acids
- Dimension of feature space $|\Sigma|^k, \Sigma = 20$

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| s | i | n | g | a | p | o | r | e |
| s | i | n | | | | | | |
| | i | n | g | | | | | |
| | | n | g | a | | | | |
| | | | g | a | p | | | |
| | | | | a | p | o | | |
| | | | | | p | o | r | |
| | | | | | | o | r | e |

# Spectrum kernel(Leslie et al. 2002)

Let $x =$ singapore and $u$ be a substring of $x$ of length 3($u$ is a $3-mer$)

**feature map**

$$\phi_3(x) \in \mathbb{R}^l \; \phi_3(x)_u = \#\text{ occurrences of } u \text{ in } x$$

$\phi_3(x) \in \mathbb{R}^l \;\; l = 26 \times 3$

# Spectrum kernel(Leslie et al. 2002)

Let $x =$ singapore and $u$ be a substring of $x$ of length 3($u$ is a $3-mer$)

## feature map

$$\phi_3(x) \in \mathbb{R}^l \; \phi_3(x)_u = \#\text{ occurrences of } u \text{ in } x$$

$\phi_3(x) \in \mathbb{R}^l \;\; l = 26 \times 3$

| aaa | ... | sin | ing | nga | gap | apo | por | ore | ... | zzz |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | ... | 1 | 1 | 1 | 1 | 1 | 1 | 1 | ... | 0 |

$$K(x,z) = \phi_3(x)^\top \phi_3(z)$$

# Allowing for mismatches

For each $u$ form a $r$ mismatch neighbourhood, $N_r(u)$

$$N_{3,1}(\text{sin}) = \{\alpha\text{in}, \text{s}\beta\text{n}, \text{si}\gamma\}$$

where $\alpha, \beta, \gamma \in A$ and $\alpha \neq s, \beta \neq i, \gamma \neq n$

$$\phi_{3,1}(u) = (I_{v \in N_{3,1}(u)})_{v \in 3-mer}$$

$$\Phi_{3,1}^{mis} = \sum_{3-mer} \sum_{u \in x} \phi_{3,1}(u)$$

Computing $K(x,z)$ can be solved in $O\left(k^{r+1}|\Sigma|^r \underbrace{(|x|+|z|)}_{linear}\right)$

# Variations on the theme (Leslie and Kuang, 2003)

- Introduce gaps
- probabilistic substitutions
- Wild cards

# What is a Generative model

$$P(X = x|\theta)$$

Hidden Markov models

# What is a Generative model

$$P(X = x|\theta)$$

Hidden Markov models
Given $D = \{x_1, \ldots, x_m\}$ Maximum likelihood estimate is given by

$$\hat{\theta} = \text{argmax}_\theta \sum_{i=1}^{n} logP(x_i|\theta)$$

# Fisher kernel(Jaakkola and Haussler 1999)

## Fisher information matrix

$$F(\theta) = \int_x U(x,\theta)U(x,\theta)^\top P(x|\theta)dx$$

$$U(x,\theta) = \nabla_\theta log P(x|\theta)$$

## Fisher Kernel

$$K_f(x,z) = U(x,\hat{\theta})^\top F^{-1}(\hat{\theta})U(x,\hat{\theta})$$

# Fisher kernel(Jaakkola and Haussler 1999)

Computing $F$ is hard but can be approximated $F \approx I$
If $x \sim$ Exponential family

$$logP(x|\theta) = s(x) + \Psi(\theta)$$

$s(x)$ sufficient statistics

$$K_f(x,z) = s(x)^\top s(z) + c$$

# Graphs play a very important role in Computational Biology

- Ideal to model molecules
- Protein-protein interaction networks
- metabolic networks

See (Vishwanathan et al. 2010, Gartner et al. 2003, Kashima et al. 2003)

# Diffusion Kernels(Kondor and Lafferty 2002)

Let $\mathscr{X} = \{1, \ldots, m\}$ and there be some associated edges between them. The adjacency matrix of the resulting graph be $A$.

## Diffusion Kernel

$$K = \lim_{s \to \infty} (I + \frac{\beta}{s} H)^s$$

$H = A - D$, where $D$ is diagonal with $d_{ii} = \sum_j a_{ij}$. $K$ is positive definite and Symmetric. Computation is $O(m^3)$

# Diffusion Kernels(Kondor and Lafferty 2002)

Let $\mathscr{X} = \{1, \ldots, m\}$ and there be some associated edges between them. The adjacency matrix of the resulting graph be $A$.

## Diffusion Kernel

$$K = \lim_{s \to \infty} (I + \frac{\beta}{s} H)^s$$

$H = A - D$, where $D$ is diagonal with $d_{ii} = \sum_j a_{ij}$. $K$ is positive definite and Symmetric. Computation is $O(m^3)$

$$\lim_{s \to \infty} (1 + \frac{\beta}{s} x)^s = e^{\beta x}$$

$K = e^{\beta H} = \sum_{i=1}^{m} v_i e^{\beta \lambda_i} v_i$ where $(\lambda_i, v_i)$ are the (eigen-value, eigen-vector) of $H$

# Diffusion Kernels(Kondor and Lafferty 2002)

Sometimes can be computed in closed form for special graphs e.g. complete graphs

$$K(i,j) = \begin{cases} \frac{1+(m-1)e^{-m\beta}}{m} & i = j \\ \frac{1-e^{-m\beta}}{m} & i \neq j \end{cases}$$

Has a very interesting analogue with Diffusion equation in physics.

# From Similarity measures to Kernels

In many applications Similarity measures are available. Let $s : \mathscr{X} \times \mathscr{X} \rightarrow \mathbb{R}$ be a similarity measure.

## Kernel from Empirical Kernel Map

$$K(x,z) = \phi(x)^\top \phi(z) = \sum_{i=1}^{l} s(x,t_i)s(z,t_i)$$

$\phi(x,t_i) = [s(x,t_1)\ldots,s(x,t_l)]^\top$(Tsuda,1999) is called the empirical kernel map. The elements $t_1,\ldots,t_l$ are pre-specified and are called templates

Found useful in remote protein Homology detection (Liao and Noble, 2002).

# Other domains

- Kernels on Probability distributions (See Kondor et al. 2004)
- An interesting set of Kernels based on Binet-Cauchy theorem was proposed by (Vishwanathan and Smola, 2004) useful for dynamical systems
- Exploiting Semigroups, Kernels on measures was derived in (Cuturi et al. 2005)

# PART 5: Advanced Topics

# Recap of SVMs

- On a dataset $D = \{(x_i, y_i) | i = 1, \ldots, m\}$ SVMs solve the following problem

$$\Gamma(K) = \max_\alpha \sum_i \alpha_i - \frac{1}{2} \alpha^\top YKY\alpha \quad (1)$$

$$0 \leq \alpha_i \leq C \qquad \sum_i \alpha_i y_i = 0 \quad (2)$$

  where $K_{ij} = k(x_i, x_j)$ is the kernel function defined on examples $x_i$ and $x_j$.

- The final classifier is $y = \text{sign}(\sum_i \alpha_i y_i k(x, x_i) + b)$

# Recap of SVMs

- Does not scale well
- The function $\Gamma(K)$ is a pointwise maximum of a set of functions and hence is convex
- If the maximization in $\alpha$ is not unique then $\Gamma(K)$ is not differentiable.
- $\Gamma(K)$ maynot be differentiable, but *subgradients* exist.
- Let us relax the problem a little and say that $\mu_i \geq 0$

# On the Problem of Kernel Design

- Learn an optimal Kernel from a library of kernel functions
- Learn an optimal kernel from several similarity functions
- Find a *good* classifier when $K$ is uncertain
- All three problems gives rise to extremely interesting optimization problems

# MKL is a Semi-definite Programming problem

$$\begin{aligned} \min_z \quad & c^\top z \\ \text{s.t.} \quad & F(z) = \sum_{i=1}^{l} z_i F_i \succeq 0 \\ & Bz = d \end{aligned}$$

$z \in \mathbb{R}^l$ and $F_i = F_i^\top \in \mathbb{R}^{m \times m}$.
$F(z)$ is positive semidefinite
Instance of a convex optimization problem. Can be solved by interior point methods

# Learning a linear combination of Multiple Kernels

Let $\{K_1, \ldots, K_n\}$ be a given library of kernels. Given a training set of $m$ examples, each $K_i = K_i^\top \in \mathbb{R}^{m \times m}$

Find a good classifier using a kernnel function of the form $\sum_{i=1}^{l} \mu_i K_i$ such that $\Gamma(K)$ is minimized(Lanckriet et al. 2004) The problem

requires $K \succeq 0$ and $trace(K) = c$(for theoretical guarantees)

# MKL formulation

## SDP formulation

$$min_{\mu,t,\lambda,\nu>0} \quad t \tag{3}$$

$$\begin{pmatrix} \sum_{i=1}^{l} \mu_i Y K_i Y^\top & e+\nu+\lambda y \\ e+\nu+\lambda y & t \end{pmatrix} \succeq 0 \tag{4}$$

$$\sum_{i=1}^{n} \mu_i K_i \succeq 0 \tag{5}$$

# Reformulation of MKL

The SDP problem can be recast as QCQPs

$$max_{\alpha,t} \quad \alpha^\top e - ct \tag{6}$$

$$s.t. \quad \alpha^\top Y K_i Y \alpha \leq r_i t \;\; i = 1,\ldots,l \tag{7}$$

$$\alpha^\top y = 0 \;\; 0 \leq \alpha \leq C \tag{8}$$

where $r_i = trace(K_i)$
QCQPs are instances of SOCPs

$$min_z \quad c^\top z \tag{9}$$

$$\|A_i z + b_i\|_2 \leq c_i^\top z + d_i \tag{10}$$

where $A_i \in \mathbb{R}^{n_i \times l}, b_i, c_i, c, z \in \mathbb{R}^l, d_i \in \mathbb{R}$

# Equivalence with Block L1 requlariza tion

Bach et al. (2004) showed that the QCQP formulation is equivalent to

$$min_{w,b,\xi} \quad \frac{1}{2}\left(\underbrace{\sum_{i=1}^{l} d_i \|w_i\|}_{\text{Block L1}}\right)^2 + C\sum_{i=1}^{m}\xi_i \tag{11}$$

$$s.t. \quad y_i(\sum_j w_j^\top \phi_j(x_i) + b) \geq 1 - \xi_i \; \forall \, i = \{1,\ldots,m\} \quad \xi_i \geq 0 \tag{12}$$

for proper choice of $d_i$

Block L1 norm promotes sparsity i.e. most of $\mu_i = 0$

# Research issues

- Scalable solutions to MKL
- Non-sparse formulations

# Efficient algorithms for MKL

## A trick

Let $\gamma \in \mathbb{B}_n = \{\gamma \in \mathbb{R}^n | \gamma_i \geq 0, \ \sum_{i=1}^n \gamma_i = 1\}$ For any $a_i \in \mathbb{R}$, $i = 1, \ldots, n$

$$(\sum_{i=1}^{n} |a_i|)^2 \leq \sum_{i=1}^{n} \frac{a_i^2}{\gamma_i}$$

This implies that

$$\left( \sum_{i=1}^{n} \|w_i\| \right)^2 \leq \sum_{i=1}^{n} \frac{1}{\gamma_i} \|w_i\|^2$$

where $\gamma$ lies in a probability simplex
Can be helpful in reformulating the L1 formulation.

# Solving MKL by reusing SVM solvers

The following problem is equivalent to the Block L1 formulation(Rakotomamonjy et al. 2007)
$S_m = \{\alpha | 0 \leq \alpha_i \leq C, \alpha^\top y = 0\}$ and $\mathbb{B} = \{\mu | 0 \leq \mu_i \sum_{i=1}^{l} \mu_i = 1\}$

$$min_{\mu \in \mathbb{B}} J(\mu) \quad \left(= max_{\alpha \in S_m} \alpha^\top e - \frac{1}{2} \sum_{i=1}^{l} \mu_i \alpha^\top Y K_i Y \alpha\right) \tag{13}$$

## A gradient descent algorithm-iteration

1.) Solve SVM problem with Kernel $K = \sum_{i=1}^{l} \mu_i K_i$

2.) Differentiate $J$ w.r.t $\mu$ and update $\mu$

See also Sonnenberg et al. 2006

# Motivation for Non-sparse MKL setting

- In many situations sparsity maybe suboptimal
- (Kloft et al. 2009) one can pursue general norms
- (Nilsback and Zisserman, 2006) points out that in object categorization tasks, employing a few of the feature descriptors or employing a canonical combination of them often leads to sub-optimal solutions.
- Simply using a sparsity criterion or a non-sparsity criterion is not going to work

# Motivation for Non-sparse MKL setting

- In object categorization (Nilsback and Zisserman, 2006) study several descriptors related to shape, color and texture. They studied a total of seven descriptors to classify flowers.
- MKL learning is of extreme importance to object categorization tasks.
- Goal is to design a suitable regularization term which could select all the descriptors and yet be able to generalize well

# Hierarchical feature space

Training dataset: $\mathscr{D} = \{(\mathbf{x}_i, y_i),\ i = 1, \ldots, m \mid \mathbf{x}_i \in \mathscr{X},\ y_i \in \{-1, 1\}\}$.
Let each individual example be described by a concatenation of features

$$\mathbf{x} = \left\{ \begin{array}{l} \left\{ \begin{array}{l} \mathbf{x}_{j1} \\ \vdots \\ \mathbf{x}_{jk} \\ \vdots \\ \mathbf{x}_{jn_j} \end{array} \right. \\ \vdots \end{array} \right.$$

where $\mathbf{x}_{jk} \in \mathbb{R}^{d_{jk}}$ is a vector of attributes which correspond to the $j,k$th block. Let $K_{jl}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_{1jl}^\top \mathbf{x}_{2jl}$

# Regularizing $\mathbf{w}$

$$y = \text{sign}\left(\sum_{j=1}^{n}\sum_{k=1}^{n_j} w_{jk}^{\top}\mathbf{x}_{jk} + b\right)$$

- Learning objective: $J(\mathbf{w}) = \Omega(\mathbf{w}) + cL(\mathscr{D})$

## A new regularization term

$$\Omega(\mathbf{w}) = \frac{1}{2}\left\{\sum_{j=1}^{n}\left\{\sum_{k=1}^{n_j}\|\mathbf{w}_{jk}\|_2\right\}^{2q}\right\}^{\frac{1}{q}}$$

for $q \geq 1$

- Block $L_1$ norm is given by $q = 1$, and $n = 1$.

# A general formulation

We consider the following formulation

$$\min_{\mathbf{w}_{jk},b,\xi_i} \quad \frac{1}{2}\left[\sum_j \left(\sum_{k=1}^{n_j} \|\mathbf{w}_{jk}\|_2\right)^{2q}\right]^{\frac{1}{q}} + C\sum_i \xi_i$$

$$\text{s.t.} \quad y_i\left(\sum_{j=1}^{n}\sum_{k=1}^{n_j} \mathbf{w}_{jk}^\top \mathbf{x}_{jki} - b\right) \geq 1-\xi_i,\ \xi_i \geq 0\ \forall\ i \qquad (14)$$

If each group contains just one element then one generalizes the MKL setting

We present a convergent algorithm, $O(m^2n^2 \ln n_t/\varepsilon^2)$,
$m =$ no of examples, $n =$ no of groups, $n_t = max_j n_j$ which solves such a problem for arbirary $q \geq 1$.

# Learning kernels from similarity functions

- Often similarity functions are more readily available than kernels.
- Sequence Similarity and Structure Similarity is readily available for Protein structures
- Classifying protein structures is an important problem where both sequence and structure are important

# Problem Definition

Given: a set of $m$ training examples with
similarity matrices $S_1, S_2, \ldots, S_n \in \mathbb{R}^{m \times m}$,
and class labels $\mathbf{y} \in \{+1, -1\}^m$.

Task: learn a kernel matrix $K$, such that SVM performs well on test data.

# Related work

$$\min_{K \succeq 0} \Gamma(K) + \rho \|K - S\|_F^2 . \tag{15}$$

- It is a minimax problem

$$min_K max_\alpha f(\alpha, K)$$

- First studied in (Luss & Aspremont, 2007) and solved using Analytic center cutting plane method.
- (Chen et al et al, 2008) formulated the above problem as Quadratically Constrained Linear Program.
- (Chen et al. 2009) used an alternate loss function $\|K - S\|_F$ which led to Second Order Cone Program.

# Related work

- Exploits strong duality

$$min_K max_\alpha f(K,\alpha) = max_\alpha min_K f(K,\alpha)$$

- For Frobenius Norm Loss one could solve $min_K f(K,\alpha)$ in a closed form

$$K^*(\alpha) = \left( S + \frac{1}{4\rho}(Y\alpha\alpha^\top Y) \right)_+$$

(Luss & Aspremont 2007)

- One can solve the resulting problem

$$max_\alpha f(K^*(\alpha),\alpha)$$

by ACCP procedure

- Will not apply to other losses

# Single Kernel based Formulation

$$\min_{K} \left[ \Gamma(K) + \rho \sum_{l=1}^{n} L_l (K - S_l) \right],$$

$$s.t. \qquad K \succeq 0, \;\; \text{trace}(K) = \tau.$$

$L_l(\cdot)$ is a convex, sub-differentiable loss function.

Examples:

1] $L_l(K - S_l) = \sum_i \sum_j |K(i,j) - S_l(i,j)|$

2] $L_l(K - S_l) = \|K - S_l\|_F$

3] $L_l(K - S_l) = \sum_i \sum_j [K(i,j) - S_l(i,j)]^2$

# Recall: Multiple Kernel Learning

$$\Gamma(K_1, \ldots, K_n) = \max_{\gamma \in \triangle} \max_{\alpha \in A_m} \left[ \alpha^\top \mathbf{1} - \sum_{l=1}^{n} \frac{\alpha^\top Y K_l Y \alpha}{2\gamma_l} \right],$$

$$\mathbb{B}_n = \left\{ \gamma \in \mathbb{R}^n \;\middle|\; \gamma \geq 0,\ \gamma^\top \mathbf{1} = 1 \right\}.$$

Optimal kernel: $K^* = \sum_{l=1}^{n} \frac{1}{\gamma_l^*} K_l$

# Multiple Kernel based Formulation

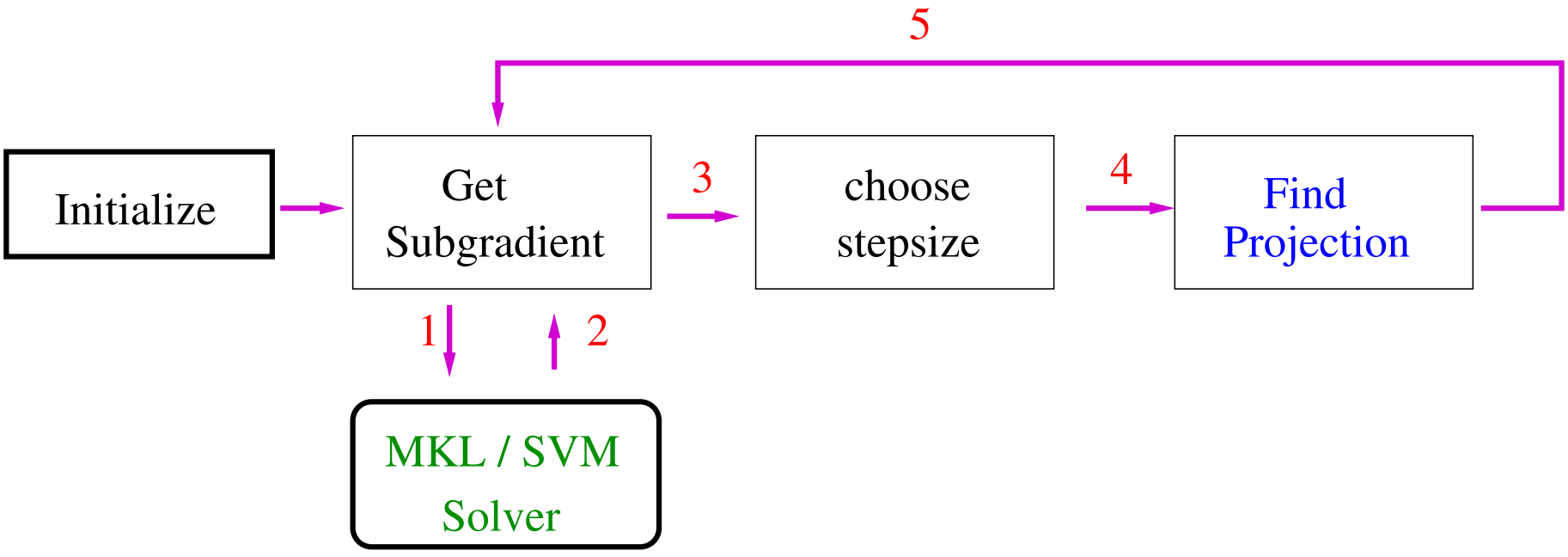
$$\min_{K_1,\ldots,K_n} \left[ \Gamma(K_1,\ldots,K_n) + \rho \sum_{l=1}^{n} L_l(K_l - S_l) \right],$$

$$s.t. \quad K_l \succeq 0, \ \mathrm{trace}(K_l) = c, \ l = 1,\ldots,n.$$

# Ideas behind the algorithm

- Solve the inner maximization over $\alpha$ for a fixed Kernel or Kernels by solving an SVM or MKL algorithm
- Solve the outer minimization by SubGradient projection.
- We choose Mirror Descent for the Subgardient projection step

# Mirror Descent based Algorithm



# MD Step 1: Subgradient Computation

- Obtain $(\alpha^*, \gamma^*)$ by solving the no n-sparse MKL.
- Calculate subgradient of $\Gamma$ as

$$\Gamma'(K_1, \ldots, K_n) = \left(\Gamma'_1, \ldots, \Gamma'_n\right),$$

$$\Gamma'_l := -\frac{1}{2\,\gamma_l^*}\, Y\alpha^*\, \alpha^{*\top} Y.$$

- Compute $L'_l$ - a subgradient of $L_l$.

# MD Step 2: Compute Projection

- Eigen decomposition:
  $\Gamma'_l(K_1^{(t)}, \ldots, K_n^{(t)}) + \rho L'_l(K_l^{(t)} - S_l)$

  $= \quad V_l^{(t)}\, diag([d_{1,l}^{(t)} \ldots d_{n,l}^{(t)}])\; V_l^{(t)\top}.$

- $\lambda_{i,l}^{(t+1)} := \dfrac{c\,\lambda_{i,l}^{(t)} \exp(-\eta_t\, d_{i,l}^{(t)})}{\sum_{j=1}^{n} \lambda_{j,l}^{(t)} \exp(-\eta_t\, d_{j,m}^{(t)})}, i = 1, \ldots, n.$

- $K_l^{(t+1)} := V_l^{(t)}\, diag([\lambda_{1,l}^{(t+1)} \ldots \lambda_{n,l}^{(t+1)}])\; V_l^{(t)\top}.$

# Algorithmic Convergence

$F^{(t)}$ : objective function value at $t$-th iteration.

$F^*$ : optimal objective value.

$\mathrm{Lip}(F)$ : Lipschitz constant of objective function .

*If the algorithm is initialized with* $K_l^{(1)} = \frac{c}{m} I$, $\forall n$ *and the stepsizes are chosen as* $\eta_t = \frac{1}{Lip(F)} \sqrt{\frac{2 \log m}{n t}}$

*then* $\min_{1 \le t \le T} F^{(t)} - F^* \le c\, n\, Lip(F) \sqrt{\frac{2 \log m}{T}}$.

# Algorithmic Complexity

$O\left(\frac{n^2 \log m}{\varepsilon^2}\right)$ iterations to reach $\varepsilon$-accurate solution.

At every iteration key comutations are:

- Eigen decomposition of $n$ matrices of dimension $m \times m$.
- Solving SVM / MKL with $m$ training examples and $n$ kernels.

# Contributions

- Note that $max_{\alpha} f(K, \alpha)$ is equivalent to solving an SVM. Unlike the state of the art we intend to solve the origninal problem

$$min_{K} max_{\alpha} f(K, \alpha)$$

- proposed three formulations which could be solved in an iterative fashion
- Each iteration is no expensive than solving an MKL/SVM
- Leads to scalable solution
- State of the art seems to be specific for the frobenius loss andcannot handle the general case
- Previous work could not handle multiple similarity matrices,

# References

*Kernel methods in Computational Biology*
Scholkopf et al. 2004
*Kernel methods for Pattern Analysis*
John Shawe Taylor and N. Cristanini
*Learning with Kernels*
Scholkopf and Smola 2002