# Fisher Discriminant Analysis

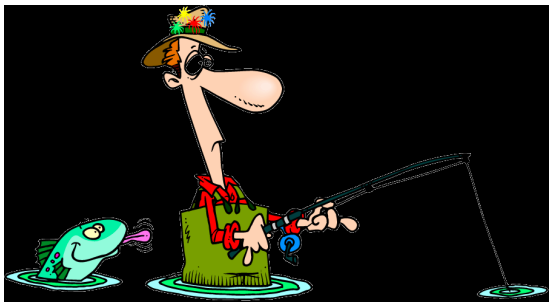


# Contents

- What is Fisher's discriminant?
- Kernelizing Fisher's discriminant
- Results

# Some history

- Introduced in 1998 by Tommi Jaakkola
- Named in honor of Sir Ronald Fisher who was an English statistician, evolutionary biologist and geneticist (analysis of variance, maximum likelihood)





# Fisher discriminant analysis

The Fisher discriminant is a classification function

$$f(x) = sgn(, \phi(x)\rangle + b)$$

where the weight $w$ is chosen to maximize the quotient

$$J(x) = \frac{(\mu_w^+ - \mu_w^-)^2}{(\sigma_w^+)^2 + (\sigma_w^-)^2}$$

where $\mu_w^+$ is the mean of the projection of the positive examples onto the direction $w$, $\mu_w^-$ the mean for the negative examples, and $\sigma_w^+$, $\sigma_w^-$ the corresponding standard deviations.

# FLD with ref to Mika's paper – Fisher Discriminant Analysis with Kernels

+1 $\mathcal{X}_1 = \{\boldsymbol{x}_1^1, \ldots, \boldsymbol{x}_{\ell_1}^1\}$ $\quad\quad$ $\mathcal{X}_2 = \{\boldsymbol{x}_1^2, \ldots, \boldsymbol{x}_{\ell_2}^2\}$ -1

$$\mathcal{X} = \mathcal{X}_1 \cup \mathcal{X}_2 = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell\}$$

$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^T S_B \boldsymbol{w}}{\boldsymbol{w}^T S_W \boldsymbol{w}}$$

$$S_B := (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^T \text{ and}$$
$$S_W := \sum_{i=1,2} \sum_{\boldsymbol{x} \in \mathcal{X}_i} (\boldsymbol{x} - \boldsymbol{m}_i)(\boldsymbol{x} - \boldsymbol{m}_i)^T$$

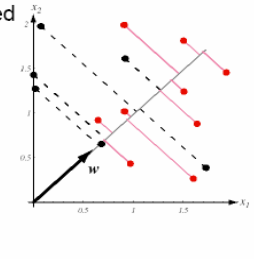
$$\boldsymbol{m}_i := \frac{1}{\ell_i} \sum_{j=1}^{\ell_i} \boldsymbol{x}_j^i$$

Different notation …. same concept

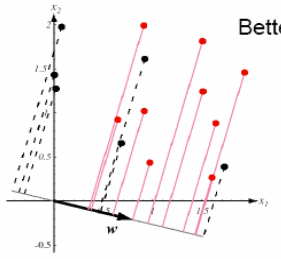
P. Agius – Spring 2008

# Fisher Liner Discriminant: two-dimensional example

Classes mixed



Better Separation



Projection of same set of two-class samples onto two different lines in the direction marked $w$.

- Unique solution :
- Easy to compute
- Probabilistic interpretation
- Kernelizable
- Naturally extends to k-class problems

# Probabilistic Interpretation

The Fisher discriminant is the Bayes optimal classifier for two normal distributions with equal covariance.

$$p(y|\omega,\sigma,x) \propto \exp\left(\frac{-1}{\sigma^2}(\omega^T x - y)^2\right) \qquad p(\omega) \propto \exp(-c\cdot\omega^T\omega)$$

Fisher discriminant analysis can be shown to:

$$\begin{aligned} &\underset{\omega}{\text{maximize}} \ \sum_i \log p(y_i,\omega|x_i,\sigma) \\ &\text{subject to} \ \sum_{i\in\{i|y_i=1\}}(\omega^T x_i - y_i)=0 \\ &\qquad\qquad \sum_{i\in\{i|y_i=-1\}}(\omega^T x_i - y_i)=0 \end{aligned} \quad \equiv \quad \begin{aligned} &\underset{\omega,\delta}{\text{minimize}} \ \|\delta\|_2^2 + c\|\omega\|_2^2 \\ &\text{subject to} \ \ \delta_i = \omega^T x_i - y_i \ , \forall i \\ &\qquad \sum_{i\in\{i|y_i=1\}}\delta_i = 0 \qquad \sum_{i\in\{i|y_i=-1\}}\delta_i = 0 \end{aligned}$$

# Colonel Trick

From the theory of reproducing kernels, we know that *w* must lie in the *span* of the training examples. Therefore, we can write *w* as

$$\boldsymbol{w} = \sum_{i=1}^{\ell} \alpha_i \Phi(\boldsymbol{x}_i)$$

Recall that

$$m_i^{\Phi} := \tfrac{1}{\ell_i}\textstyle\sum_{j=1}^{\ell_i} \Phi(\boldsymbol{x}_j^i)$$

Therefore

$$\begin{aligned} \boldsymbol{w}^T \boldsymbol{m}_i^{\Phi} &= \frac{1}{\ell_i}\sum_{j=1}^{\ell}\sum_{k=1}^{\ell_i} \alpha_j \,\mathrm{k}(\boldsymbol{x}_j, \boldsymbol{x}_k^i) \\ &= \boldsymbol{\alpha}^T \boldsymbol{M}_i \qquad \text{where} \quad (\boldsymbol{M}_i)_j := \tfrac{1}{\ell_i}\textstyle\sum_{k=1}^{\ell_i} \mathrm{k}(\boldsymbol{x}_j, \boldsymbol{x}_k^i) \end{aligned}$$

Recall $$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^T S_B^\Phi \boldsymbol{w}}{\boldsymbol{w}^T S_W^\Phi \boldsymbol{w}}$$

The numerator and denominator now become

$$\boldsymbol{w}^T S_B^\Phi \boldsymbol{w} = \boldsymbol{\alpha}^T M \boldsymbol{\alpha} \qquad \boldsymbol{w}^T S_W^\Phi \boldsymbol{w} = \boldsymbol{\alpha}^T N \boldsymbol{\alpha}$$

$$M := (\boldsymbol{M}_1 - \boldsymbol{M}_2)(\boldsymbol{M}_1 - \boldsymbol{M}_2)^T \qquad N := \sum_{j=1,2} K_j (I - \mathbf{1}_{\ell_j}) K_j^T$$

$$(\boldsymbol{M}_i)_j := \frac{1}{\ell_i} \sum_{k=1}^{\ell_i} \mathrm{k}(\boldsymbol{x}_j, \boldsymbol{x}_k^i) \qquad (K_j)_{nm} := \mathrm{k}(\boldsymbol{x}_n, \boldsymbol{x}_m^j)$$

Therefore $$J(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^T M \boldsymbol{\alpha}}{\boldsymbol{\alpha}^T N \boldsymbol{\alpha}}$$



# Kernel Fisher Discriminant

Solve the problem

$$J(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^T M \boldsymbol{\alpha}}{\boldsymbol{\alpha}^T N \boldsymbol{\alpha}}$$

by finding the leading eigenvector of $N^{-1}M$

**One regularization method** that leads to more stable results when solving for the leading eigenvector involves replacing N by

$$N_\mu := N + \mu I$$

where I=identity matrix and $\mu$ is a positive parameter inducing a positive semi-definite matrix

# Another Regularized Formulation

Choose $w$ that solves the following optimization problem:

$$\max_w J(w) = \frac{(\mu_w^+ - \mu_w^-)^2}{(\sigma_w^+)^2 + (\sigma_w^-)^2 + \lambda||w||^2}$$

Kernel Methods for Pattern Analysis, Ch 5



# Results – Mika paper

| | RBF | AB | $AB_R$ | SVM | KFD |
|---|---|---|---|---|---|
| Banana | **10.8±0.6** | 12.3±0.7 | *10.9±0.4* | 11.5±0.7 | **10.8±0.5** |
| B.Cancer | 27.6±4.7 | 30.4±4.7 | 26.5±4.5 | *26.0±4.7* | **25.8±4.6** |
| Diabetes | 24.3±1.9 | 26.5±2.3 | 23.8±1.8 | *23.5±1.7* | **23.2±1.6** |
| German | 24.7±2.4 | 27.5±2.5 | 24.3±2.1 | **23.6±2.1** | *23.7±2.2* |
| Heart | 17.6±3.3 | 20.3±3.4 | 16.5±3.5 | **16.0±3.3** | *16.1±3.4* |
| Image | 3.3±0.6 | **2.7±0.7** | **2.7±0.6** | *3.0±0.6* | 4.8±0.6 |
| Ringnorm | 1.7±0.2 | 1.9±0.3 | *1.6±0.1* | 1.7±0.1 | **1.5±0.1** |
| F.Sonar | 34.4±2.0 | 35.7±1.8 | 34.2±2.2 | **32.4±1.8** | *33.2±1.7* |
| Splice | *10.0±1.0* | 10.1±0.5 | **9.5±0.7** | 10.9±0.7 | 10.5±0.6 |
| Thyroid | 4.5±2.1 | *4.4±2.2* | 4.6±2.2 | 4.8±2.2 | **4.2±2.1** |
| Titanic | 23.3±1.3 | *22.6±1.2* | *22.6±1.2* | **22.4±1.0** | 23.2±2.0 |
| Twonorm | 2.9±0.3 | 3.0±0.3 | *2.7±0.2* | 3.0±0.2 | **2.6±0.2** |
| Waveform | 10.7±1.1 | 10.8±0.6 | **9.8±0.8** | *9.9±0.4* | *9.9±0.4* |

RBF = RBF classifier
AB = Adaboost, ABR = regularized Adaboost
KFD = Kernel Fisher Discriminant

Using the Fisher kernel method to detect remote protein homologies
(Jaakkola, 1999)

Goal:

- Detect remote protein homologies
- Sequence-based tools
- SCOP database
- Use HMM for each family/superfamily

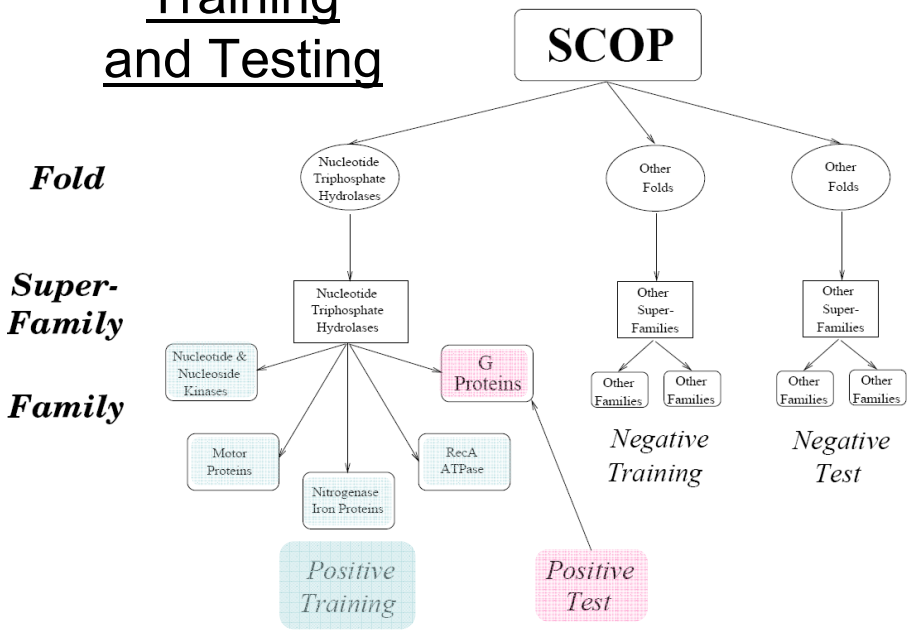


# Jaakkola 1999

Method:

- Derive a kernel function using HMMs
- Kernel function specifies a similarity score for any pair of protein sequences (in contrast to the HMM likelihood which is a measure of the closeness of a sequence to the HMM model)
- Derive a Fisher score

# Training and Testing






# Methodology details

$X = [x_1, ..., x_n]$ is a protein sequence (amino acids).

Build a HMM $H_1$ for a particular family of proteins

Then $P(X|H_1)$ is the probability of X under H1 and $P(X|H_0)$ is the probability of X under the null model.

The score used in the database search is defined by the log likelihood of X

$$\begin{aligned}\mathcal{L}(X) &= \log \frac{P(X|H_1)P(H_1)}{P(X|H_0)P(H_0)} \\ &= \log \frac{P(X|H_1)}{P(X|H_0)} + \log \frac{P(H_1)}{P(H_0)}\end{aligned}$$

Positive L(x) → x is a member of the family

# The Fisher kernel method

$$\begin{aligned}\mathcal{L}(X) &= \log P(H_1|X) - \log P(H_0|X) \qquad (2)\\ &= \sum_{i:X_i \in H_1} \lambda_i K(X, X_i) - \sum_{i:X_i \in H_0} \lambda_i K(X, X_i)\end{aligned}$$

The sign of the discriminant function determines the assignment of the sequences into hypothesis classes.

The Fisher score is $U_X = \nabla_\theta \log P(X|H_1, \theta)$ (Set of model parameters)

where Ux is the derivative of the loglikelihood for sequence X with respect to a particular parameter (hence determines contribution of parameter)

The kernel function is then defined to be

$$K(X, X') = e^{-\frac{1}{2\sigma^2}(U_X - U_{X'})^T (U_X - U_{X'})}$$



# Summary of the method

- Derive HMM trained on positive examples
- Use HMM to map a sequence X to a feature vector (its Fisher score)
- Compute the kernel function
- The resulting discriminant function is:

$$\mathcal{L}(X) = \sum_{i:X_i \in H_1} \lambda_i K(X, X_i) - \sum_{i:X_i \in H_0} \lambda_i K(X, X_i),$$

This is the SVM-Fisher method

# Other comparative methods

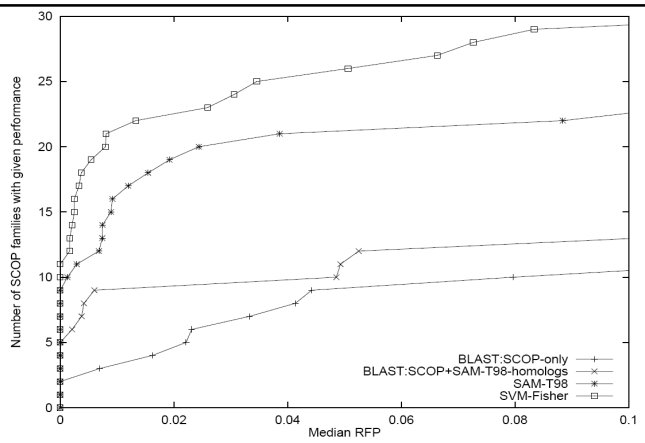
- BLAST – family pairwise search; database is queried with each positive training sequence
- SAMT-98 – builds HMM for a SCOP domain sequence, then iteratively selects positive training examples from among potential homologs and refines the model
- BLAST + SAM
- SVM-Fisher



# Results

| Sequence | BLAST | B-Hom | S-T98 | SVM-F |
|---|---|---|---|---|
| 5p21 | 0.043 | 0.010 | 0.001 | 0.000 |
| 1guaA | 0.179 | 0.031 | 0.000 | 0.000 |
| 1etu | 0.307 | 0.404 | 0.428 | 0.038 |
| 1hurA | 0.378 | 0.007 | 0.007 | 0.000 |
| 1eft(3) | 0.431 | 0.568 | 0.041 | 0.051 |
| 1dar(2) | 0.565 | 0.391 | 0.289 | 0.019 |
| 1tadA(2) | 0.797 | 0.330 | 0.004 | 0.000 |
| 1gia(2) | 0.867 | 0.421 | 0.017 | 0.000 |

Table 1: Rate of false positives for *G proteins* family. BLAST = BLAST:SCOP-only, B-Hom = BLAST:SCOP+SAMT-98-homologs, S-T98 = SAMT-98, and SVM-F = SVM-Fisher method.




RFP = Rate of False Positives

Figure 2: Here we compare the overall performance for the four methods on the 33 test families. For each family we computed the median RFP for that family, as shown in Table 2. Values for the median RFP are shown on the X-axis. On the Y-axis we plot the number of SCOP families, out of the 33 families that we tested, for which the given method achieves that median RFP performance or better.

# Future work and other

- Which sequences should be used to estimate the HMM parameters?
- Which sequences should be left for the discriminative function?
- Can build generic models by building HMMs on short amino acid sequences that map to conserved regions in the proteins
- Extend the method to identify multiple domains within large protein sequences
- Fisher's discriminant for multiple classes (Ch 4, Chris Bishop)