ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

inf | Informatik
Computer Science

Improved Functional Prediction of Proteins by Learning Kernel Combinations in Multilabel Settings

Volker Roth, Bernd Fischer
ETH Zurich, Institute of Computational Science

Overview

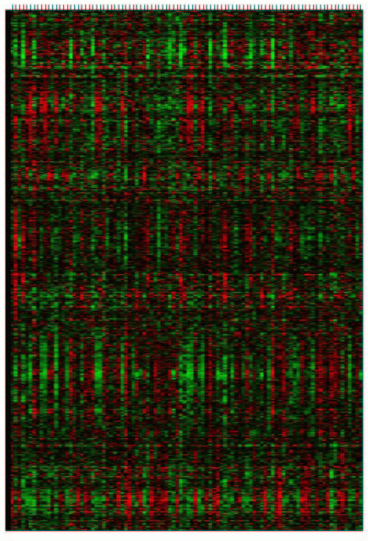
- Genes/proteins characterized by different measurements
  $\Rightarrow$ **kernels** for combining **heterogeneous data**.

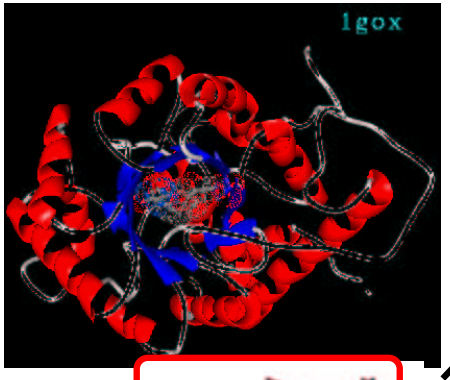
- Proteins can have multiple functions
  $\Rightarrow$ **multilabel classifier** which learns kernel combinations

- MIPS functional catalogue: hierarchy of protein functions
  $\Rightarrow$ **kernelHMM: hierarchical multilabel classifier**.

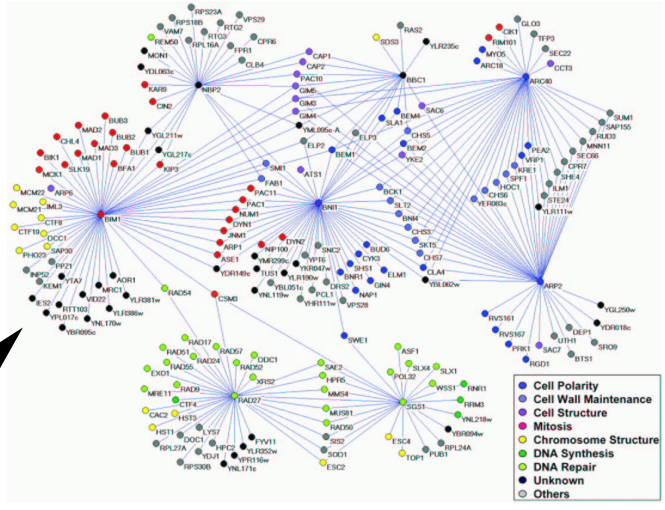
Measurement of heterogeneous Data

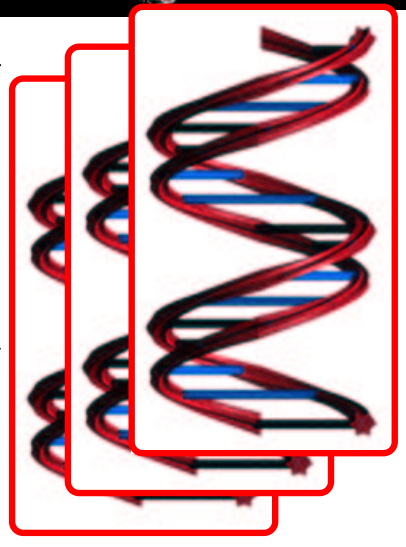
Gene expression



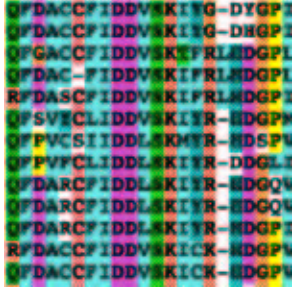


Genetic interactions

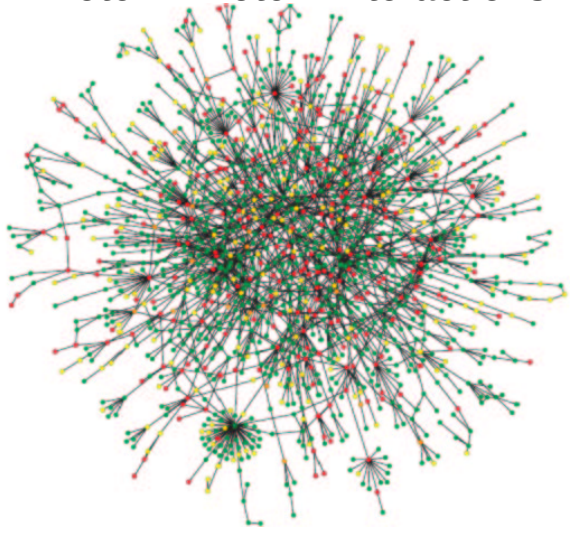




Protein Sequences



Protein–Protein Interactions



Goal: Data-fusion, "Common Language" $\rightsquigarrow$ **Mercer Kernels**

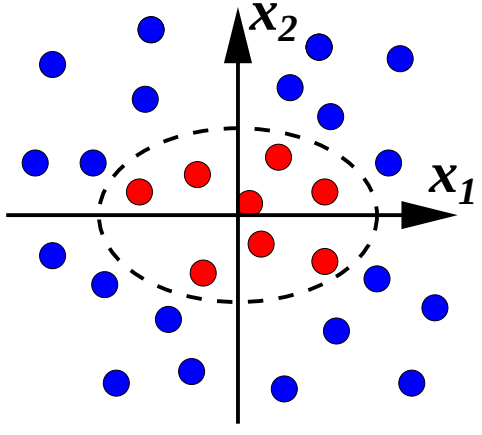
Mercer Kernels

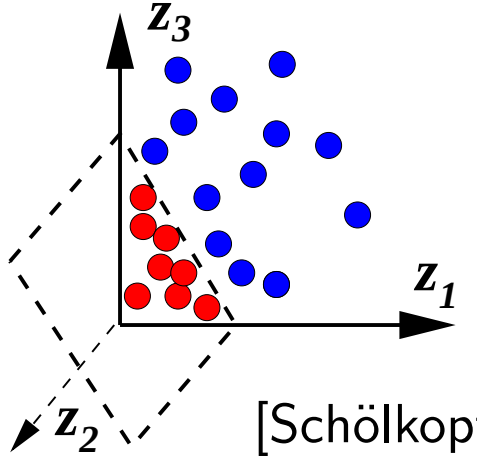
- **Support Vector Machines**: Non-linear learning machines

- "Classical View": vectorial data.
  Mapping from **input space** to **feature space**

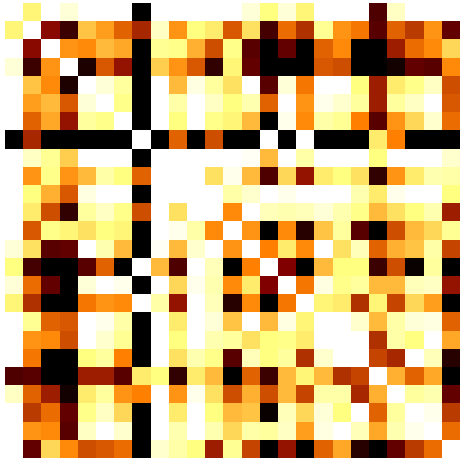
$$\phi : \mathbb{R}^2 \to \mathbb{R}^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3)$$
$$= (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\phi}(\boldsymbol{x}) \cdot \boldsymbol{\phi}(\boldsymbol{x}')$$
$$= (\boldsymbol{x} \cdot \boldsymbol{x}')^2$$



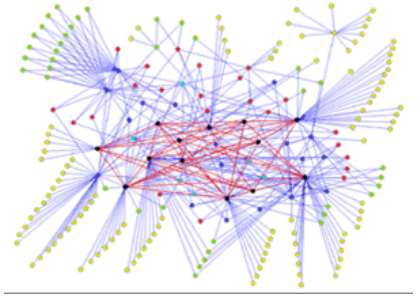


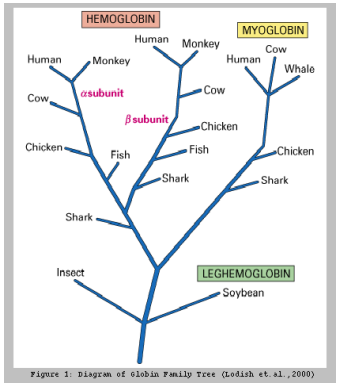
[Schölkopf, 2000]



Non-vectorial Data

- Bioinformatics: many **non-vectorial** data-types:
  - Interaction graphs





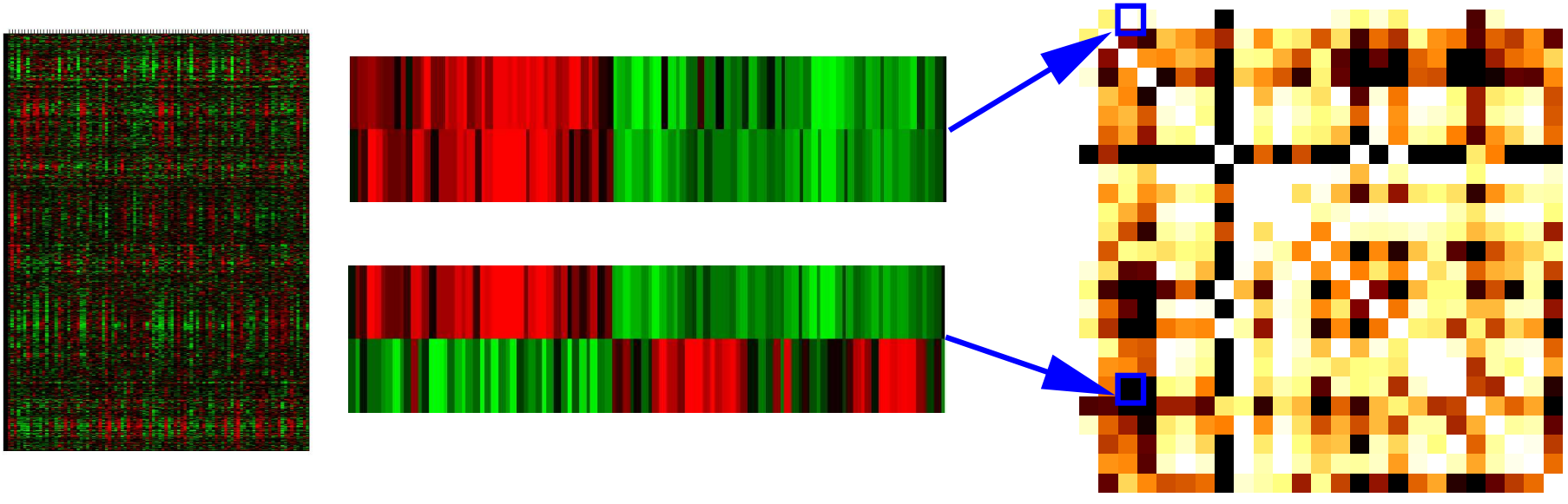
  - Phylogenetic trees
  - Strings GSAQVKGHGKKVADALTNAVAHV

- **Observation:** each $n \times n$ kernel matrix contains **dot products** $\rightsquigarrow$ **pairwise similarities**

- **Data fusion:** Data of each type are converted into a kernel $\Rightarrow$ "common language" for heterogeneous data.

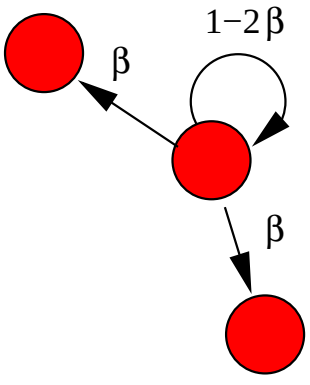
RBF-Kernels for Expression Data

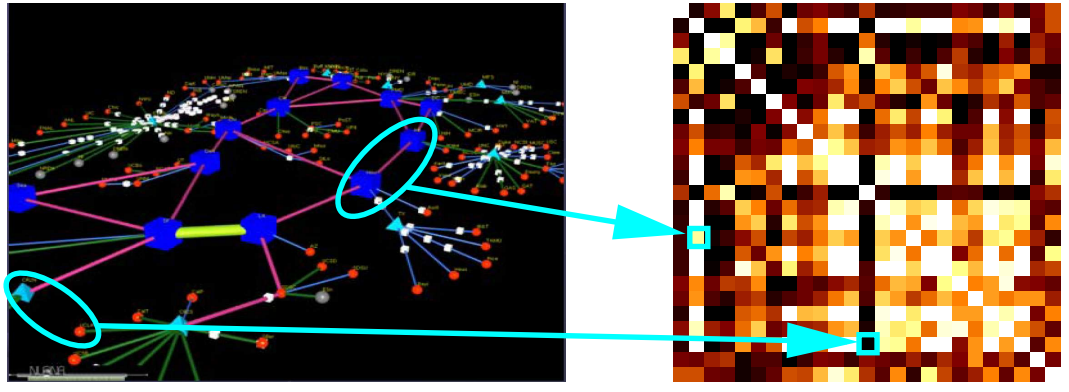
- **For each gene:** vector of expression values under different experimental conditions
- "classical" RBF-Kernel $k(x_1, x_2) = \exp(-\sigma \|x_1 - x_2\|^2)$



Diffusion-Kernels for Interaction Graphs

- A: Adjacency matrix, D: node degrees, $L = D - A$.
- $K := \frac{1}{Z(\beta)} \exp(-\beta L)$ with transition probability β.
- **Physical interpretation (random walk):** next node randomly picked among neighboring nodes.
- Self-transition probabilities: $1 - d_i\beta$



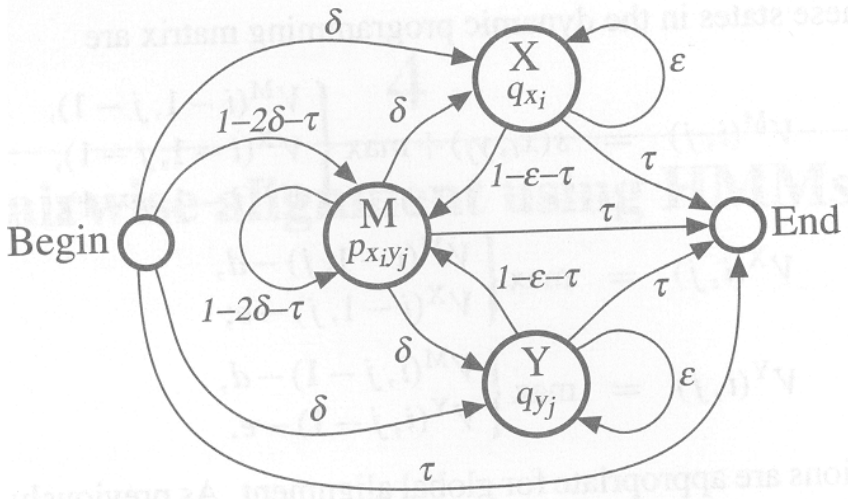


- K_{ij}: Probability to walk from i to j.

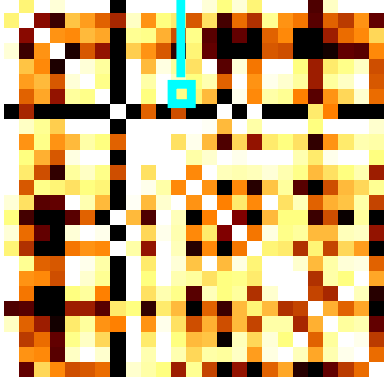
(Kondor and Lafferty, ICML 2002)

Alignment kernels for Sequences

Sequence alignment via **Pair HMMs** $\Rightarrow$ Mercer kernel (Watkins, 2000)




```
HBA_HUMAN   GSAQVKGHGKKVADALTNAVAHV---D--DMPNALSALSDLHAHKL
            ++ ++++H+ KV    + +A   ++             +L+ L+++H+ K
LGB2_LUPLU NNPELQAHAGKVFKLVYEAAIQLQVTGVVVTDATLKNLGSVHVSKG
```

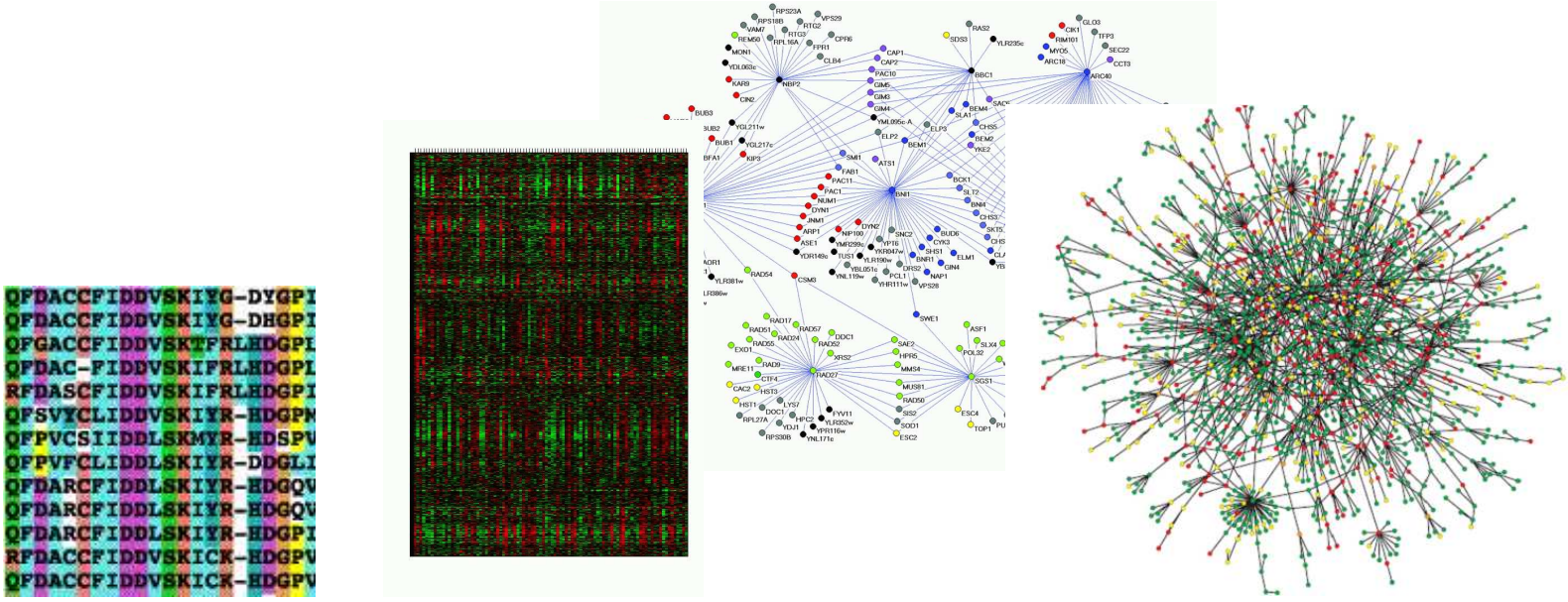


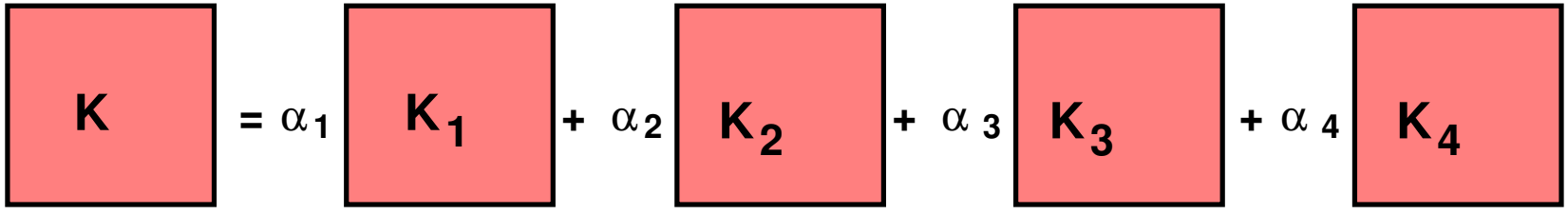
Combination of heterogeneous Data

Addition of kernels $\Rightarrow$ **new kernel**:

$$k_1(x,y) = \phi_1(x) \cdot \phi_1(y),$$
$$k_2(x,y) = \phi_2(x) \cdot \phi_2(y)$$
$$\Rightarrow k' = k_1 + k_2 = \begin{pmatrix}\phi_1(x)\\ \phi_2(x)\end{pmatrix} \cdot \begin{pmatrix}\phi_1(y)\\ \phi_2(y)\end{pmatrix}$$

$\Rightarrow$learn weighted combination of m kernels:





Functional Classification of Yeast Proteins

- **Goal:** learn classifier that assigns each yeast protein to **one or several** functional classes
  $\Rightarrow$ **hierarchical learning problem with multiple labels**

- Yeast has $\approx$ 6000 genes, for $\approx$ 3500 the function(s) is/are "known".

Functional Classification: *MIPS comprehensive yeast genome database* `www.mips.gsf.de`

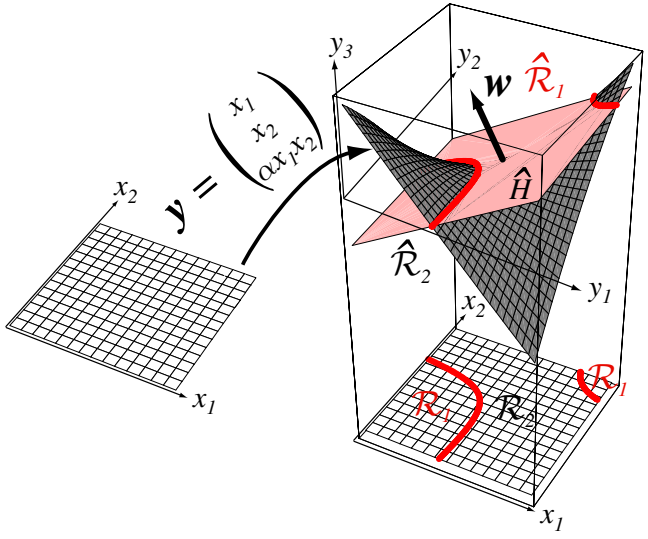
	Category		Category
1	metabolism	10	cellular transport
2	energy	11	cellular communication
3	storage	12	cell rescue, defense
4	cell cycle and DNA processing	13	interaction with cellular environment
5	transcription	14	interaction with environment(systemic)
6	protein synthesis	15	transposable elements
7	protein fate	16	cell fate
8	protein with binding function	17	development
9	protein activity regulation	18	biogenesis of cellular components

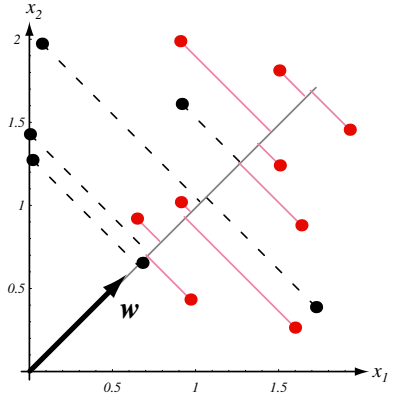
Designing the Classifier: Wishlist

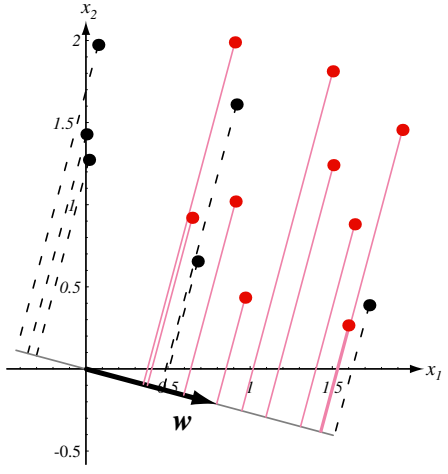
- **Data-fusion:** sparse combinations of kernel matrices.
- **Multiple protein functions**: consistent handling of multiple classes with **multilabels**;
- **Efficiency:** many objects, many kernels, out-of-core kernel matrices...
- **Structured outputs:** consistent handling of **hierarchical multilabels**.

Designing the Classifier: NKDA

- Method: multilabel version of **Nonlinear Kernel Discriminant Analysis** with built-in **feature selection**
  $\rightsquigarrow$ subset of optimal kernel weights.

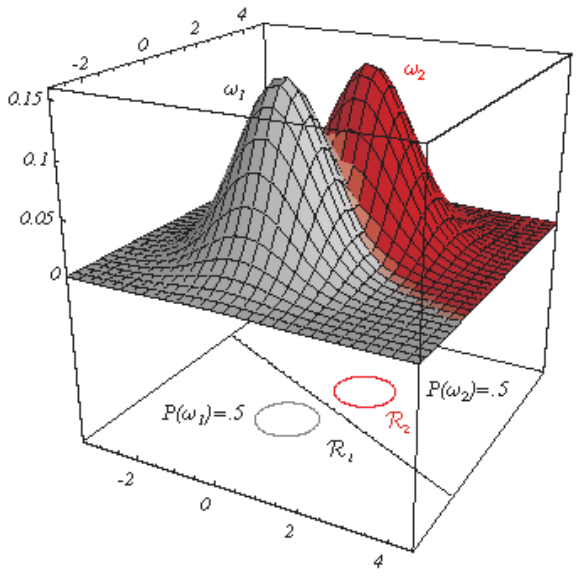


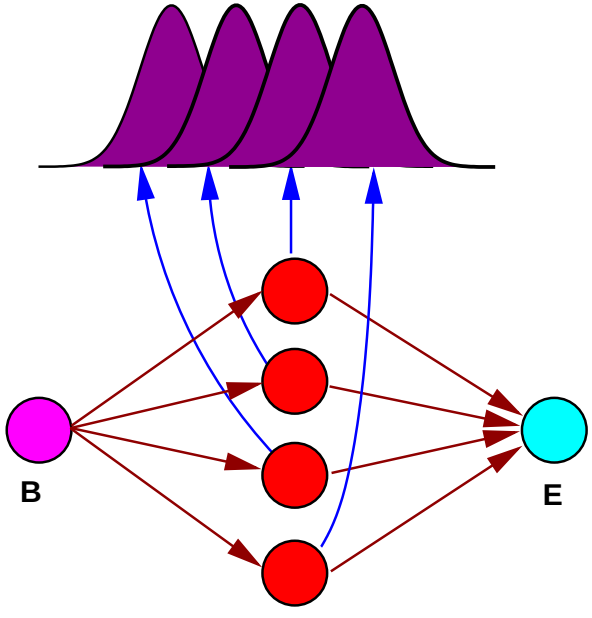




Designing the Classifier: multilabels

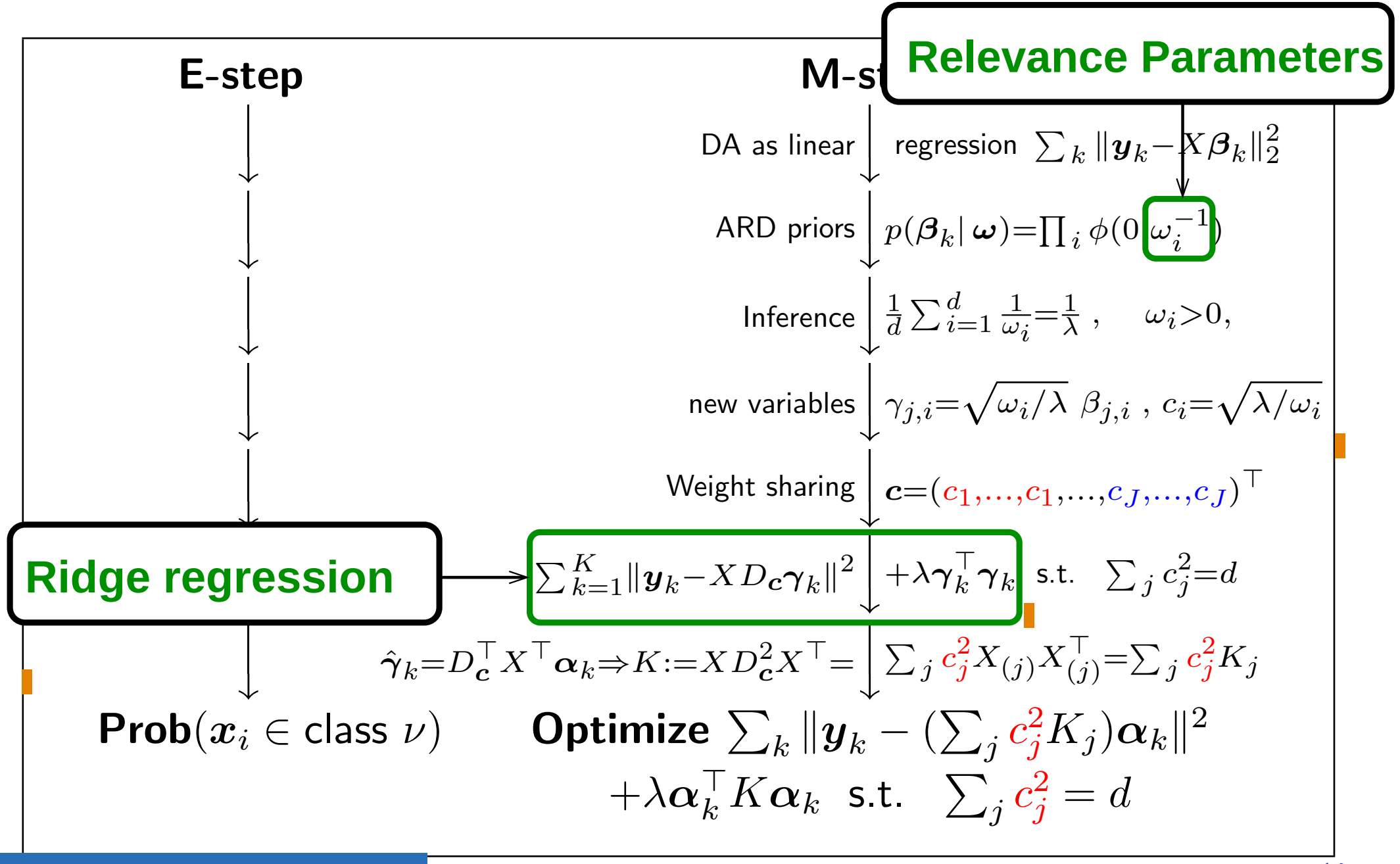
- Unsupervised LDA: **Mixture DA** [Hastie 96].
  Observation: LDA finds maximum likelihood solution for Gaussian mixture model with "pooled" covariance





- **Unsupervised NKDA** $\rightsquigarrow$ Gaussian mixtures in kernel space
  $\rightsquigarrow$ EM algorithm

Convex Kernel Combinations: Modified EM



Nonlinear KDA with Feature Selection

- Core of the algorithm: **Ensemble of linear systems** with different right-hand sides:

$$(\sum_{j=1}^{m} c_j^2 K_j + \lambda I)\boldsymbol{a}_k = \boldsymbol{y}_k$$

- Approximation: **block conjugate gradients** (Dubrulle 01)
- For fixed number of iterations: $\Rightarrow O(n^2)$ algorithm
- (Lanckriet et al. 2004): **quadratically constrained QP**: $O(n^{4.5})$ algorithm and $O(n^3)$ approximation.
  (Sonnenburg et al., 2006): **semi-infinite LP**:
  Complexity unclear, seems to work well.

Multilabel Classification as Special Case

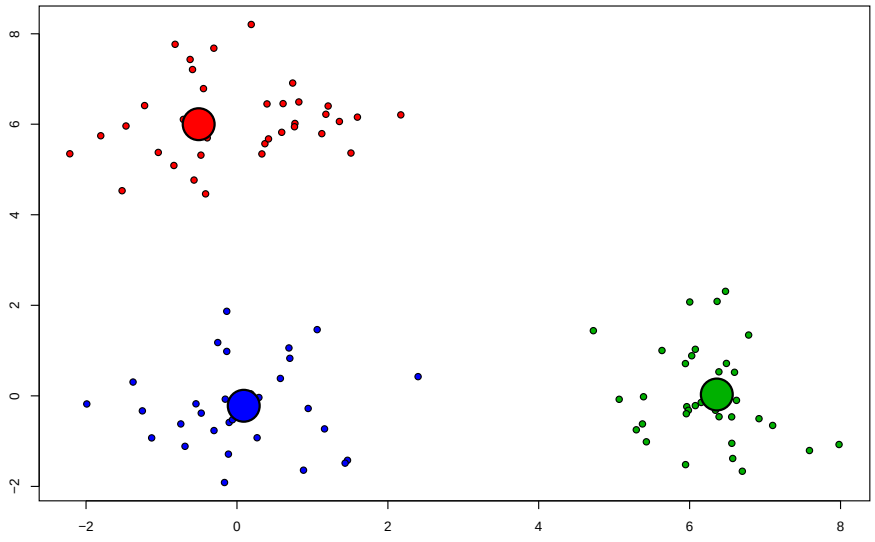
- Object $\boldsymbol{x}_i$ can have **multiple** labels $\rightsquigarrow$ **label set** Y_i
- **Probabilistic treatment**: object is generated from mixture of class-specific distributions.
  $\rightsquigarrow$ multilabels modeled as class-assignment probabilities

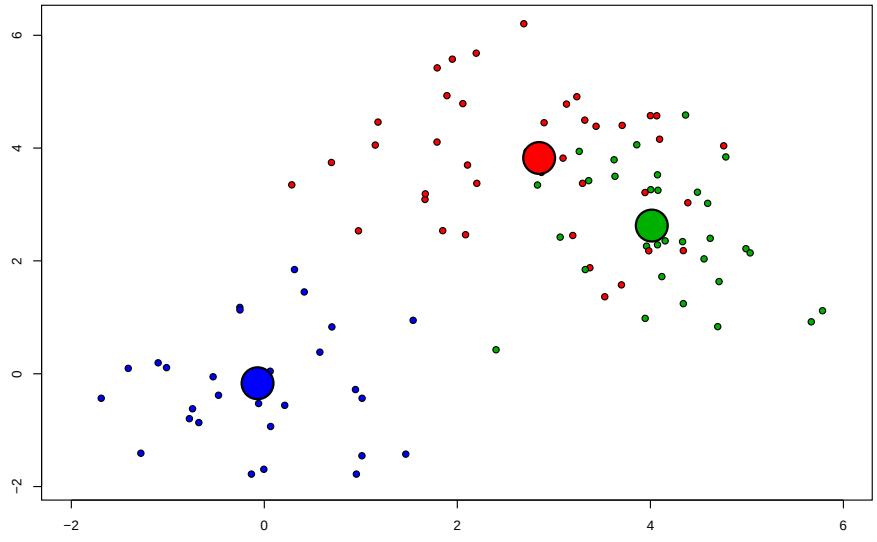
$$p(C_j|\boldsymbol{x}_i) = 1/|Y_i|.$$

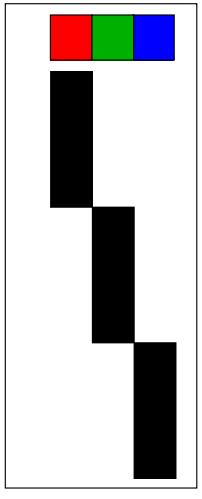
- **Run one single M-step**.
- **Inference for x_*:**
  $\rightsquigarrow$ assign object to k most probable classes such that

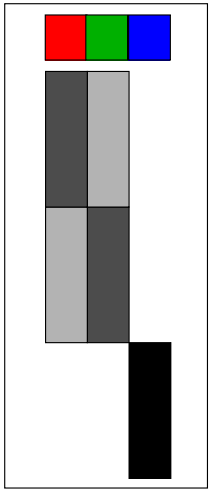
$$\sum_{j=1}^{k} p^{\text{ordered}}(C_j|\boldsymbol{x}_*) > 1/2.$$

Effect of Multiple Labels

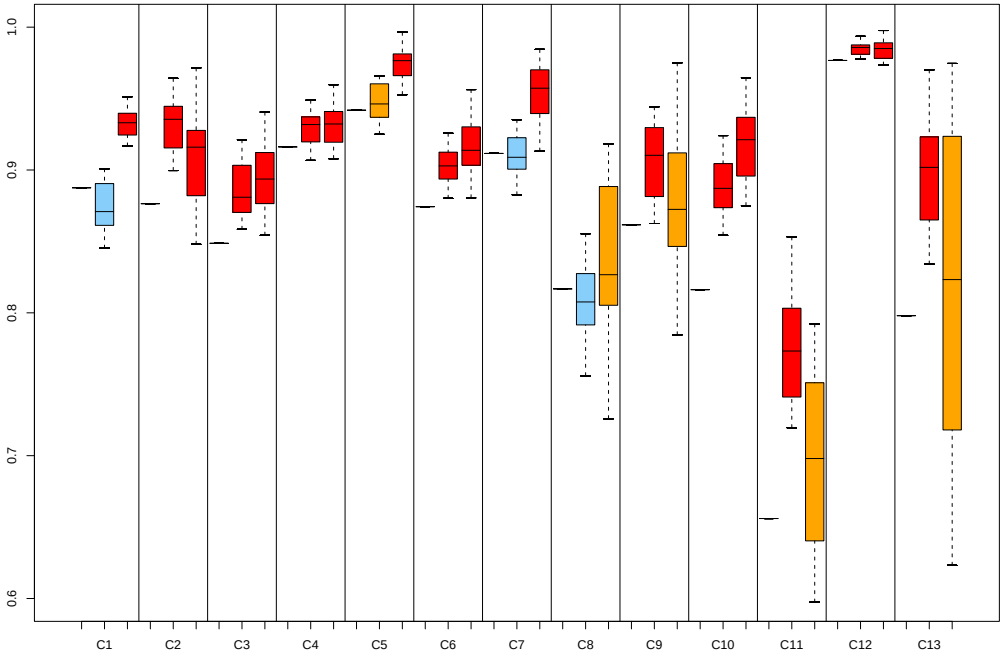




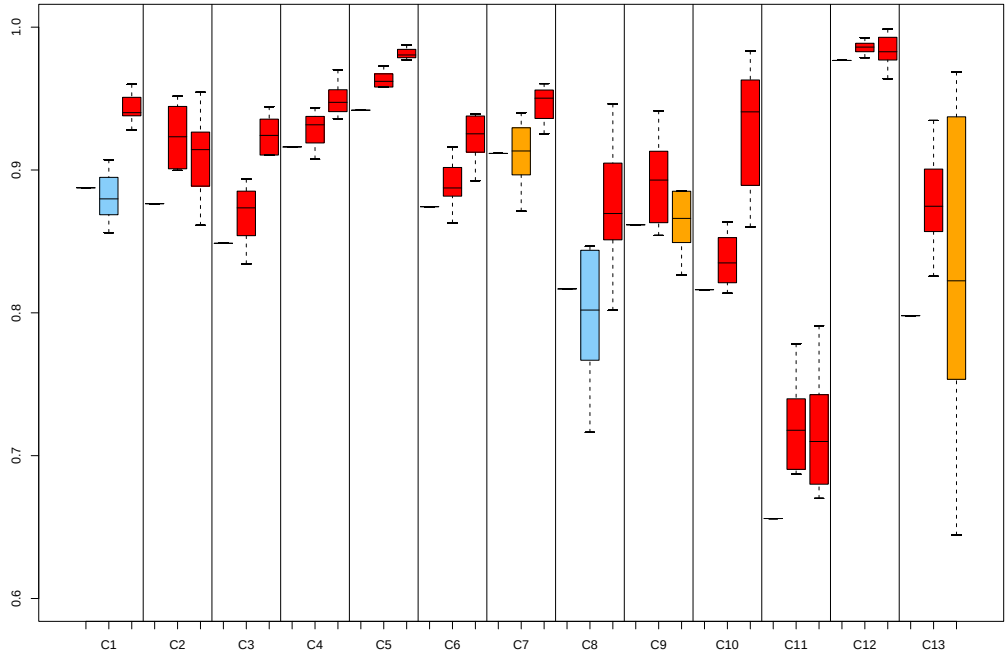




Results



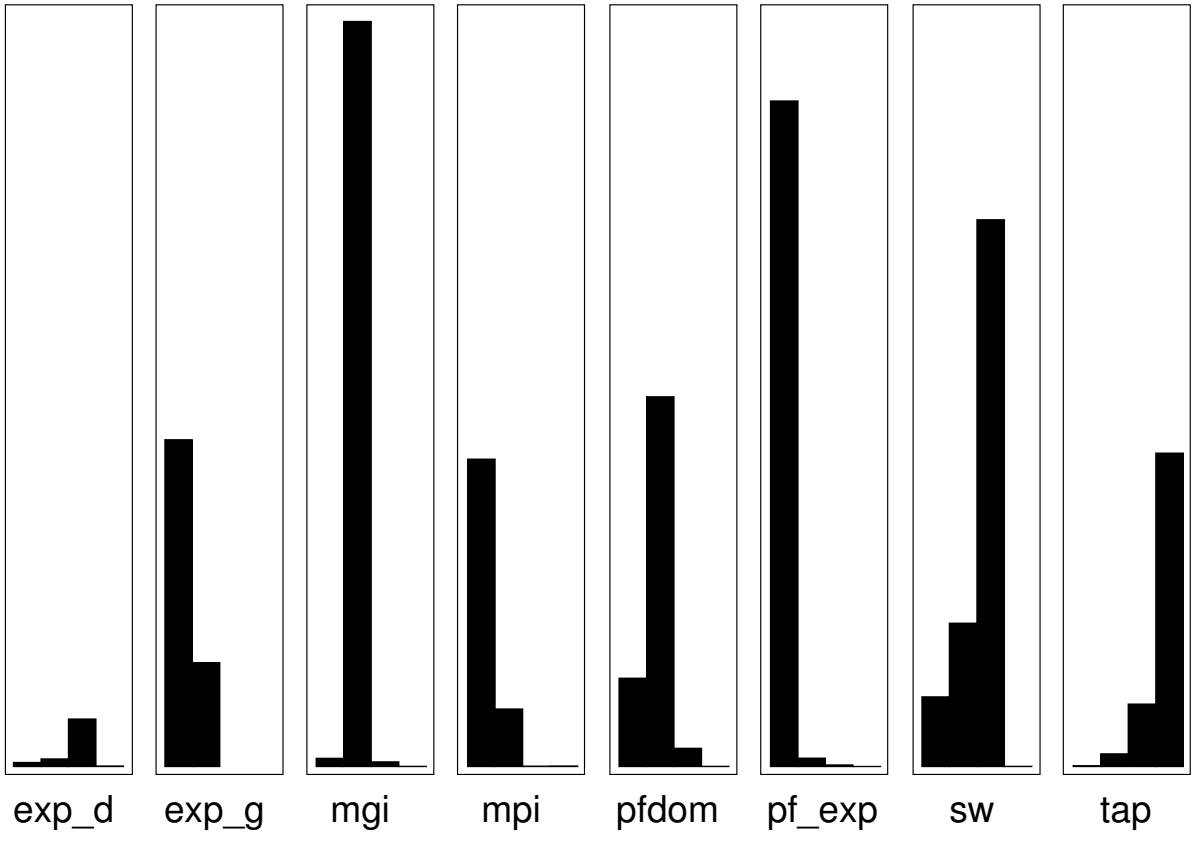
Original 8 kernels



Extended kernel set (32)

- "correct" probabilistic handling of **multiple labels**.
- use of 32 kernels (instead of 8). Newest version: 120 kernel matrices: block-CG with optimized hard-disk access.

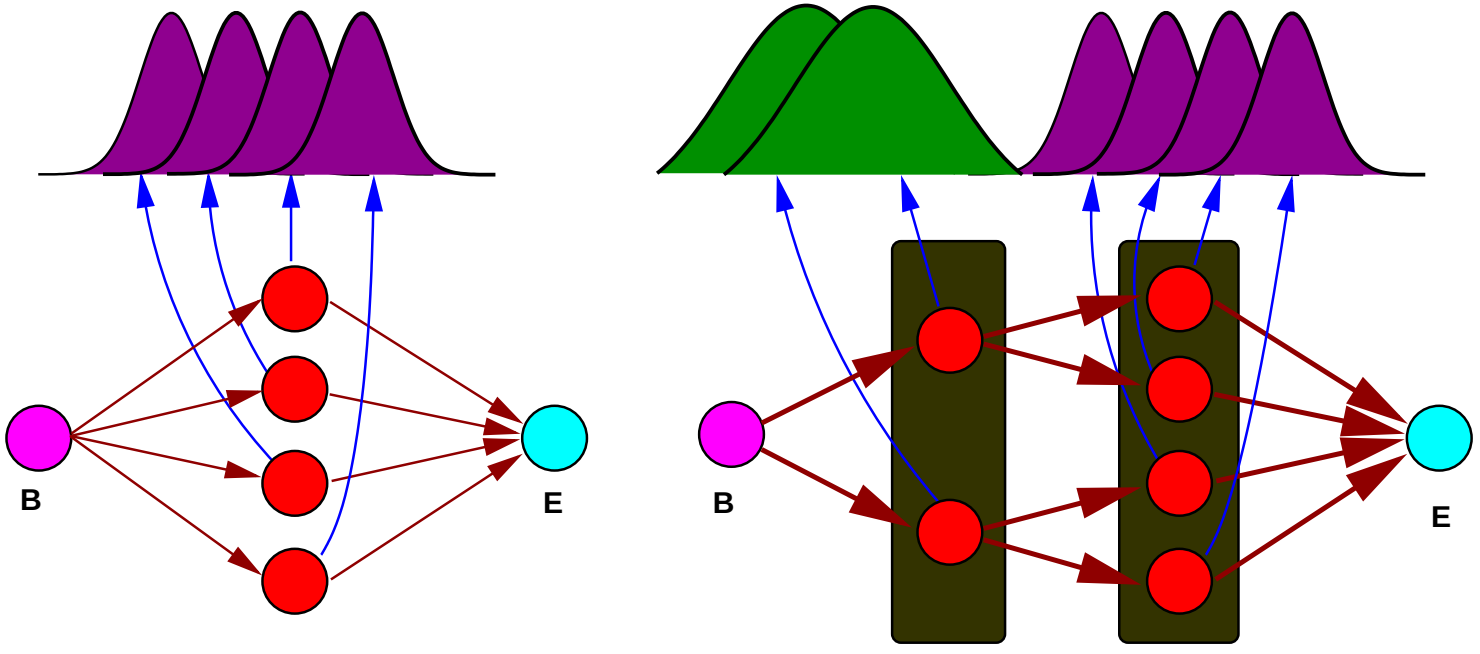
Learned Kernel Weights



- **Surprising observation:** genetic interactions are most important...but only RBF variants!

Hierarchical Extension: The Kernel HMM

Hierarchical classification: multiple layers

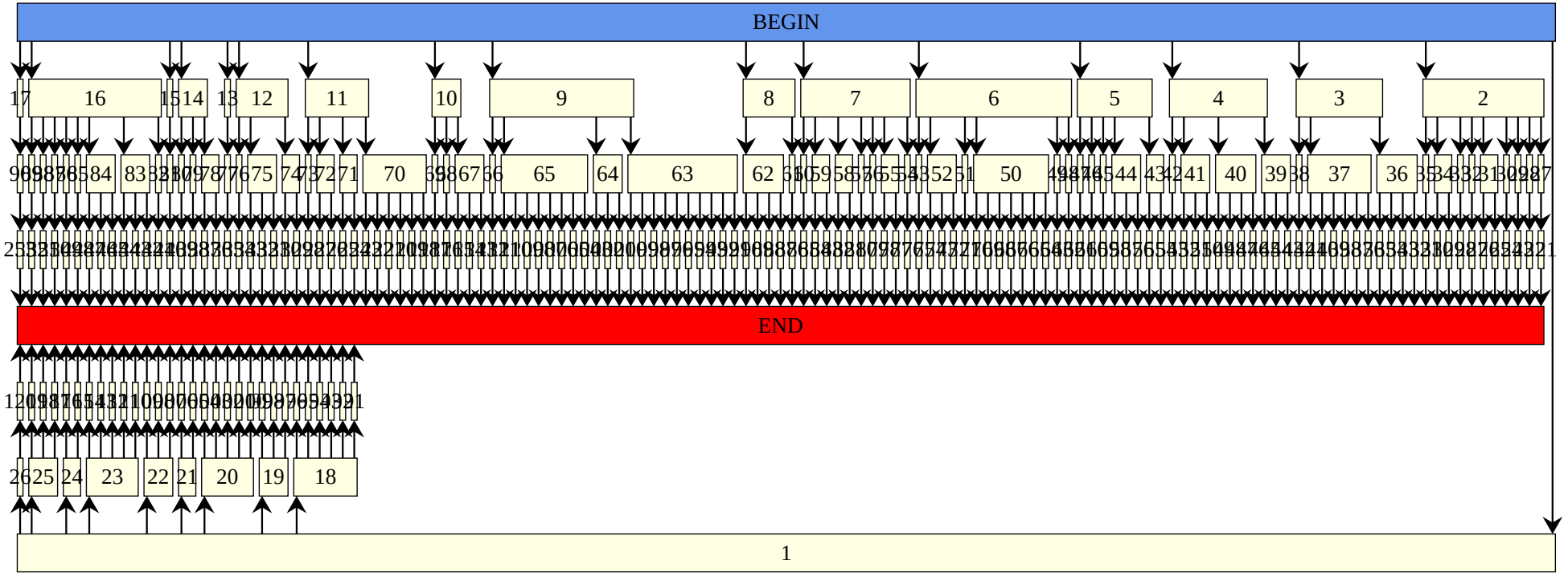


Tree-trellis variant of Viterbi algorithm:

$\Rightarrow k$ **most probable paths.**

Consistent handling of multiple hierarchical labels

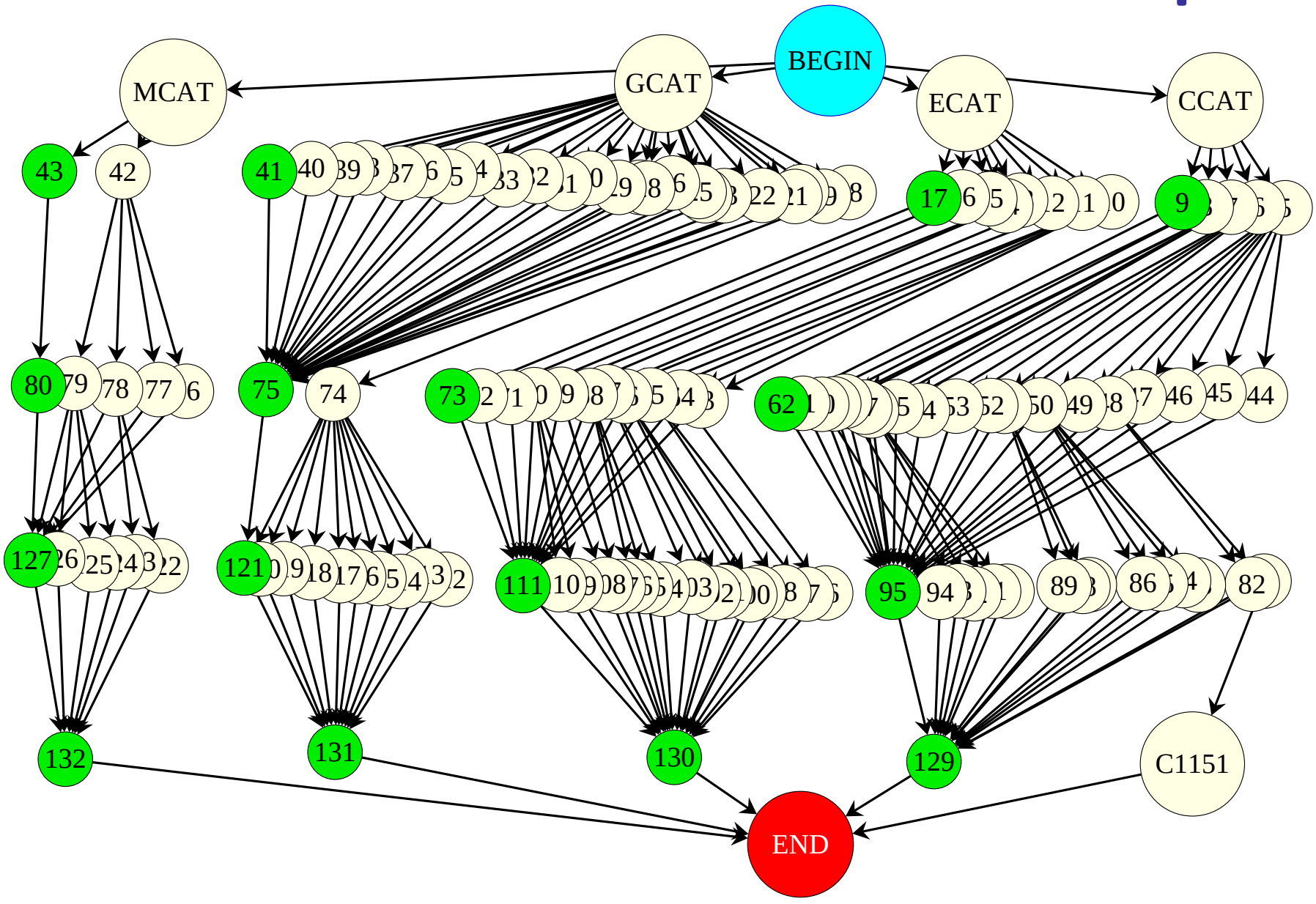
The Kernel HMM for Yeast



FunCat 2.0 hierarchy (MIPS).

$\Rightarrow$ increased performance in deeper levels of hierarchy .

Summing Up

- Kernels as **"common language"** for heterogeneous data
- Aggregation of kernels $\Rightarrow$ new kernel
- **Goal:** train classifier and find optimal kernel weights
- **Model:** Nonlinear Kernel Discriminant Analysis:
  - probabilistic treatment of multiple labels
  - efficient algorithm
  - excellent prediction of protein functions
  - "building block" in HMM $\Rightarrow$ hierarchical classification

Document Classification: RCV1 corpus