Kernel Matrix Regression

Yoshihiro Yamanishi[1] Jean-Philippe Vert[2]

[1]Bioinformatics Center
Kyoto University

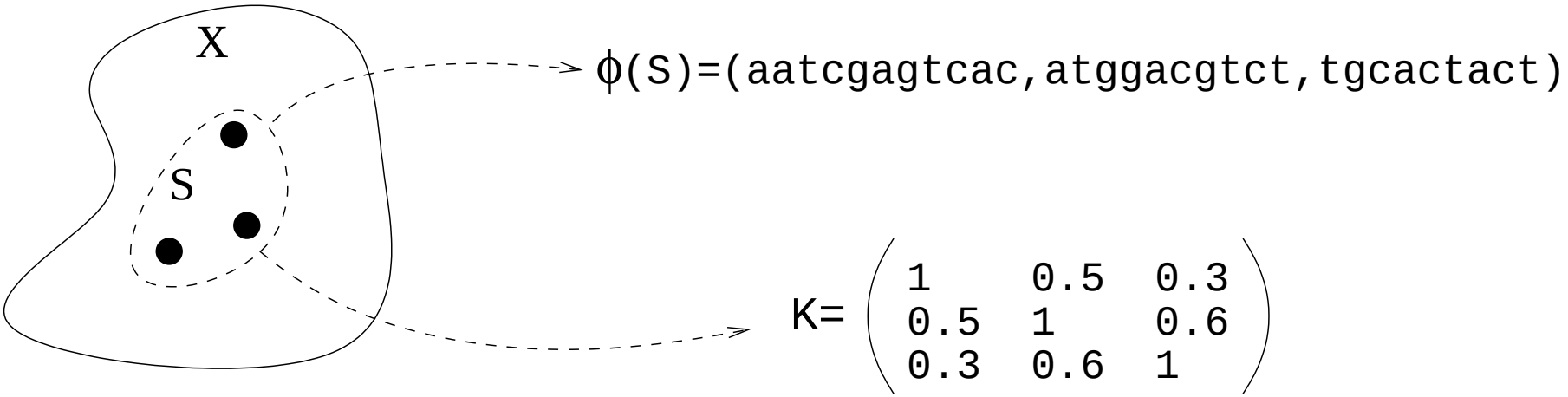
[2]Centre for Computational Biology
Ecole des Mines de Paris, ParisTech

The 12th International Conference on Applied Stochastic Models and Data Analysis (ASMDA 2007), Chania, Greece, May 29, 2007.

Outline

Outline

Kernel methods



- A positive definite kernel is a function $K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ such that any Gram matrix is positive semidefinite.
- Many algorithm for data analysis, called kernel methods, are based on p.d. kernels (SVMs, kernel PCA, kernel regression, ...)

Examples

Kernels for vectors

Classical kernels for vectors ($\mathcal{X} = \mathbb{R}^p$) include:

- The linear kernel

$$K_{lin}\left(\mathbf{x}, \mathbf{x}'\right) = \mathbf{x}^\top \mathbf{x}' \ .$$

- The polynomial kernel

$$K_{poly}\left(\mathbf{x}, \mathbf{x}'\right) = \left(\mathbf{x}^\top \mathbf{x}' + a\right)^d \ .$$

- The Gaussian RBF kernel:

$$K_{Gaussian}\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-\frac{\| \mathbf{x} - \mathbf{x}' \|^2}{2\sigma^2}\right) \ .$$

Examples

Kernels for vectors

Classical kernels for vectors ($\mathcal{X} = \mathbb{R}^p$) include:

- The linear kernel

$$K_{lin}\left(\mathbf{x}, \mathbf{x}'\right) = \mathbf{x}^\top \mathbf{x}' \ .$$

- The polynomial kernel

$$K_{poly}\left(\mathbf{x}, \mathbf{x}'\right) = \left(\mathbf{x}^\top \mathbf{x}' + a\right)^d \ .$$

- The Gaussian RBF kernel:

$$K_{Gaussian}\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-\frac{\| \mathbf{x} - \mathbf{x}' \|^2}{2\sigma^2}\right) \ .$$

Examples

Kernels for vectors

Classical kernels for vectors ($\mathcal{X} = \mathbb{R}^p$) include:

- The linear kernel

$$K_{lin}\left(\mathbf{x}, \mathbf{x}'\right) = \mathbf{x}^\top \mathbf{x}' \ .$$

- The polynomial kernel

$$K_{poly}\left(\mathbf{x}, \mathbf{x}'\right) = \left(\mathbf{x}^\top \mathbf{x}' + a\right)^d \ .$$

- The Gaussian RBF kernel:

$$K_{Gaussian}\left(\mathbf{x}, \mathbf{x}'\right) = \exp\left(-\frac{\| \mathbf{x} - \mathbf{x}' \|^2}{2\sigma^2}\right) \ .$$

P.d. kernels are inner products

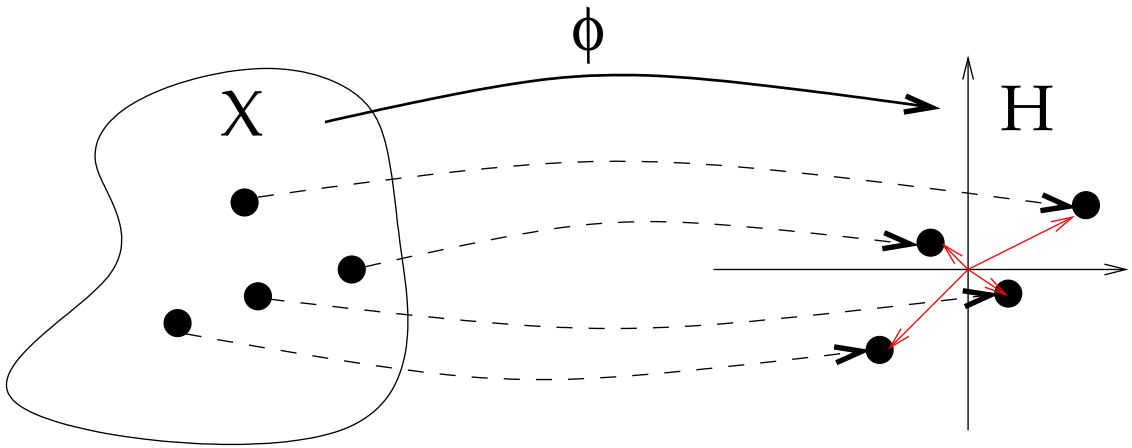
Theorem (Aronszajn, 1950)

K is a p.d. kernel on the set $\mathcal{X}$ if and only if there exists a Hilbert space $\mathcal{H}$ and a mapping

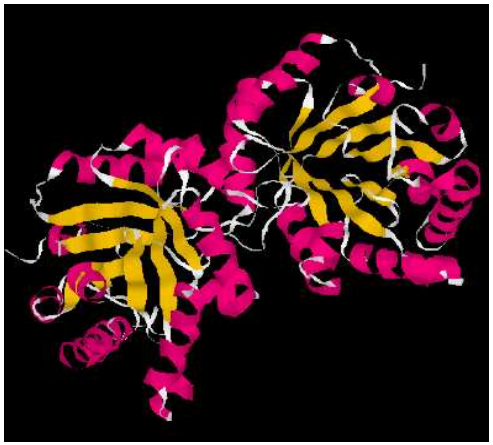
$$\Phi : \mathcal{X} \mapsto \mathcal{H} \, ,$$

such that, for any $\mathbf{x}, \mathbf{x}'$ in $\mathcal{X}$:

$$K\left(\mathbf{x}, \mathbf{x}'\right) = \left\langle \Phi\left(\mathbf{x}\right), \Phi\left(\mathbf{x}'\right) \right\rangle_{\mathcal{H}} \, .$$



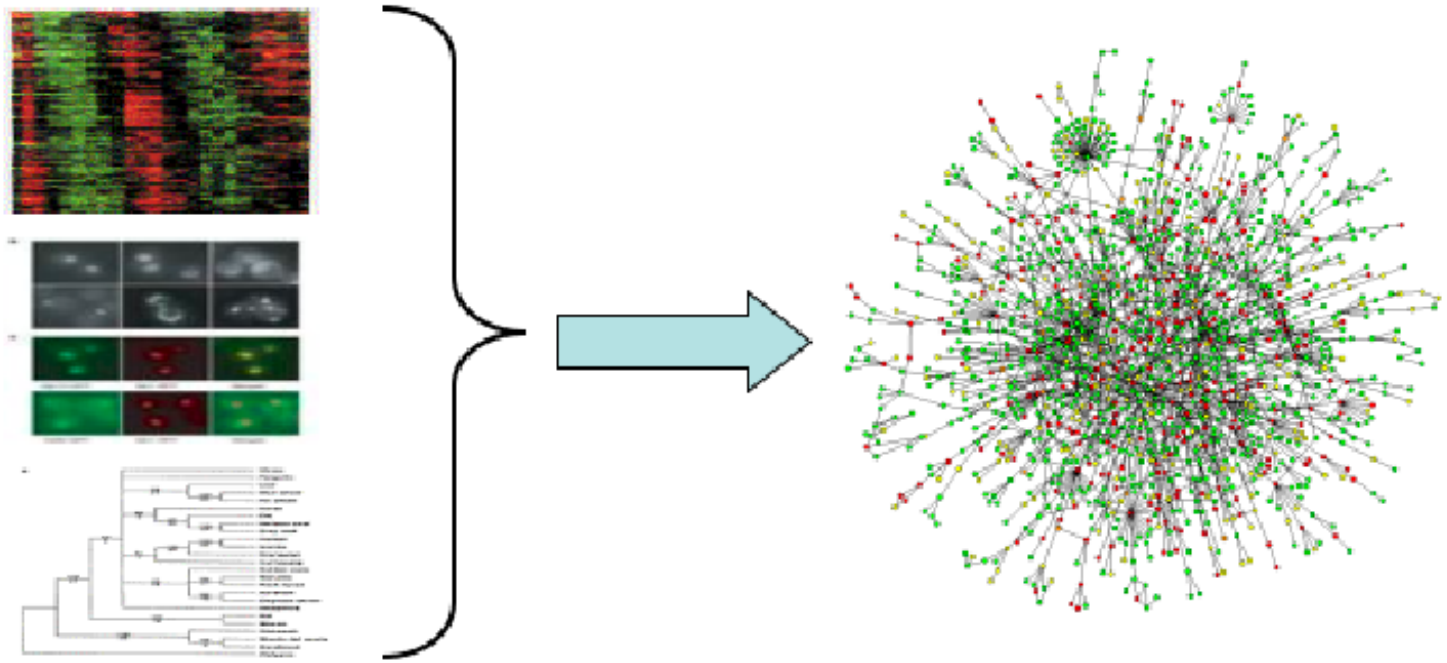
Motivation 1: “expensive” vs “cheap” kernels



FVNQHLCGSHLVEALYLVCGERGFF...

- Objective: predict the function of proteins $\Longrightarrow$ we need a kernel for proteins.
- “Cheap” kernel: all protein sequences are easily known, we easily define a kernel K_{seq} based on sequences.
- “Expensive” kernel: a few protein structures are known, we can define a kernel K_{struct} based on structures
- The problem: K_{struct} is more relevant, but not available for all proteins; K_{seq} is less relevant but known for all proteins.

Motivation 2: Graph inference



- Available genomic data = K_{input}
- Graph to predict (partially known) = K_{output} (after Hilbert embedding)
- Problem: predict missing entries in K_{output} from K_{input}

The problem

$$K = \begin{pmatrix} K_{tt} & K_{pt}^\top \\ K_{pt} & K_{pp} \end{pmatrix}, \quad G = \begin{pmatrix} G_{tt} & G_{pt}^\top \\ G_{pt} & G_{pp} \end{pmatrix},$$

- K is fully known
- G_{tt} is known, but not G_{pt} nor G_{pp}
- Goal: predict G_{pt} and G_{pp} from K and G_{tt}

Outline

Direct approach

The idea

The cheap kernel could be used in place of the expensive kernel.

Solution

Use the cheap kernel as a proxy for the expensive one:

$$G_{pt} = K_{pt} \quad G_{pp} = K_{pp}.$$

Direct approach

The idea

The cheap kernel could be used in place of the expensive kernel.

Solution

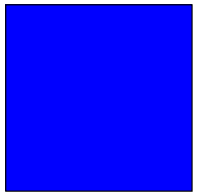
Use the cheap kernel as a proxy for the expensive one:

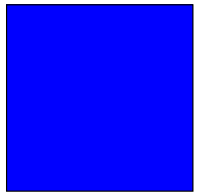
$$G_{pt} = K_{pt} \quad G_{pp} = K_{pp}.$$

Kernel CCA approach (Y. et al., 2004)

The idea

- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.

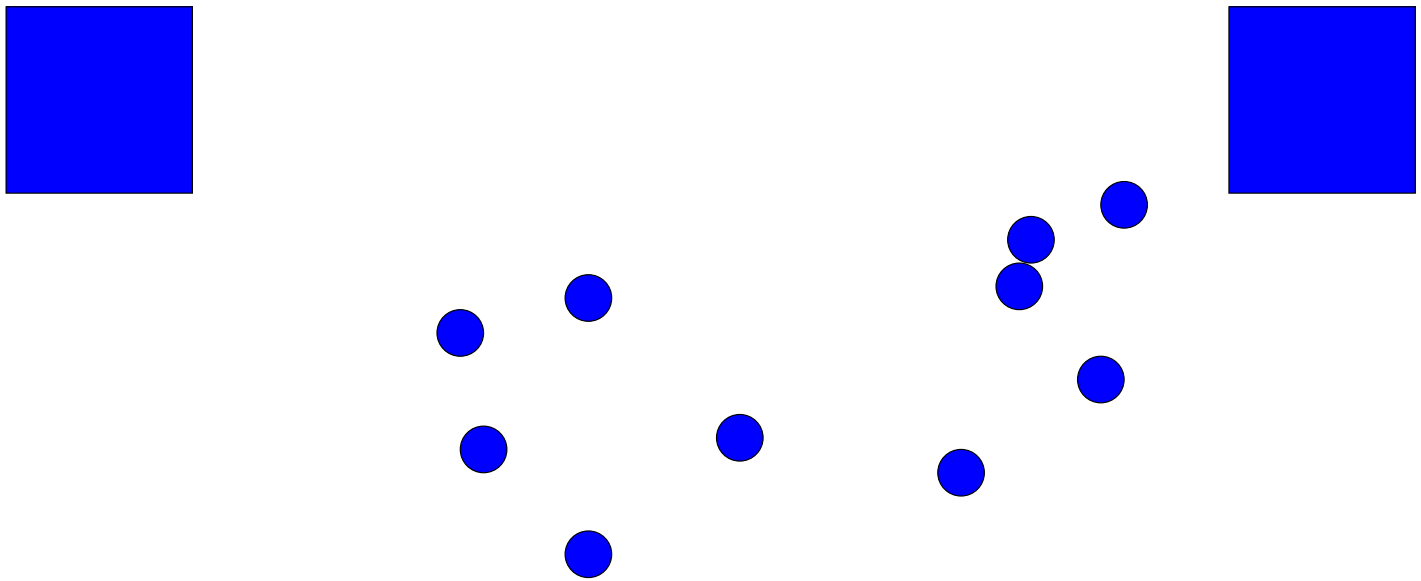




Kernel CCA approach (Y. et al., 2004)

The idea

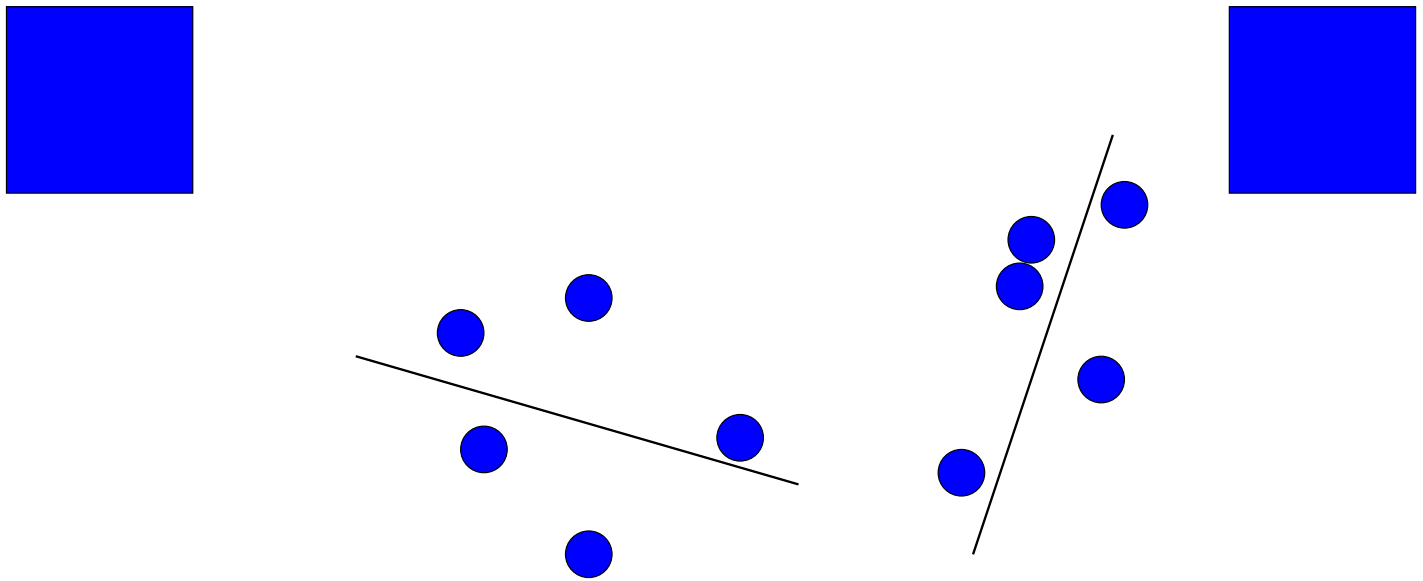
- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.



Kernel CCA approach (Y. et al., 2004)

The idea

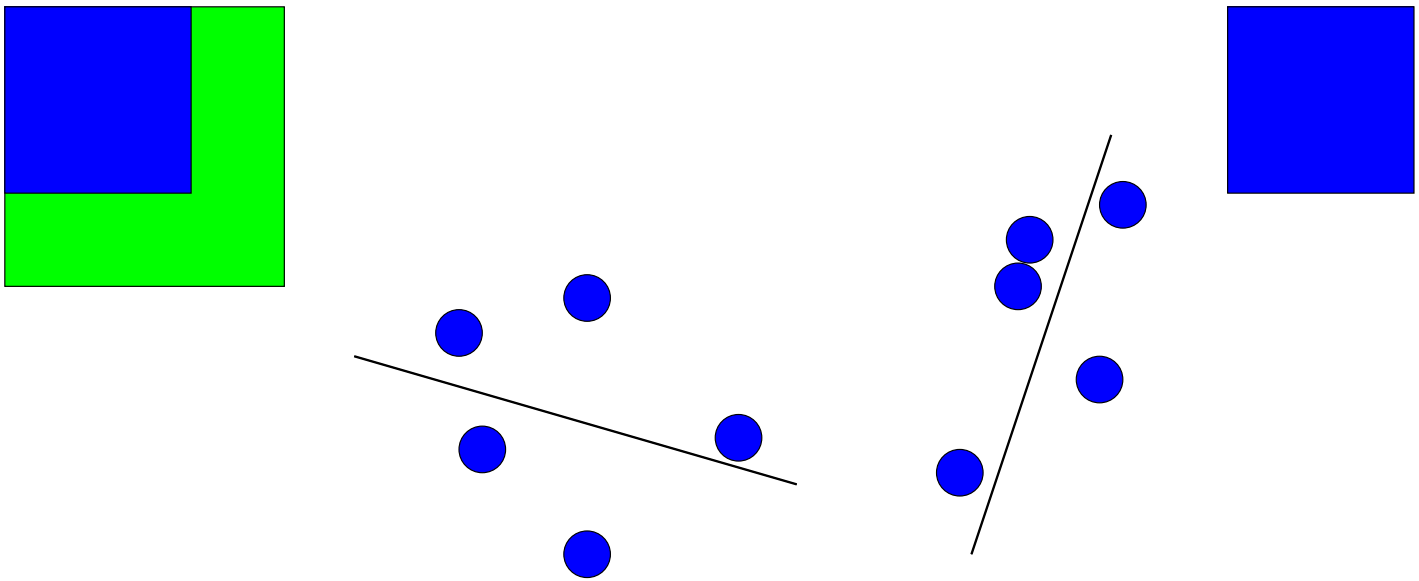
- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.



Kernel CCA approach (Y. et al., 2004)

The idea

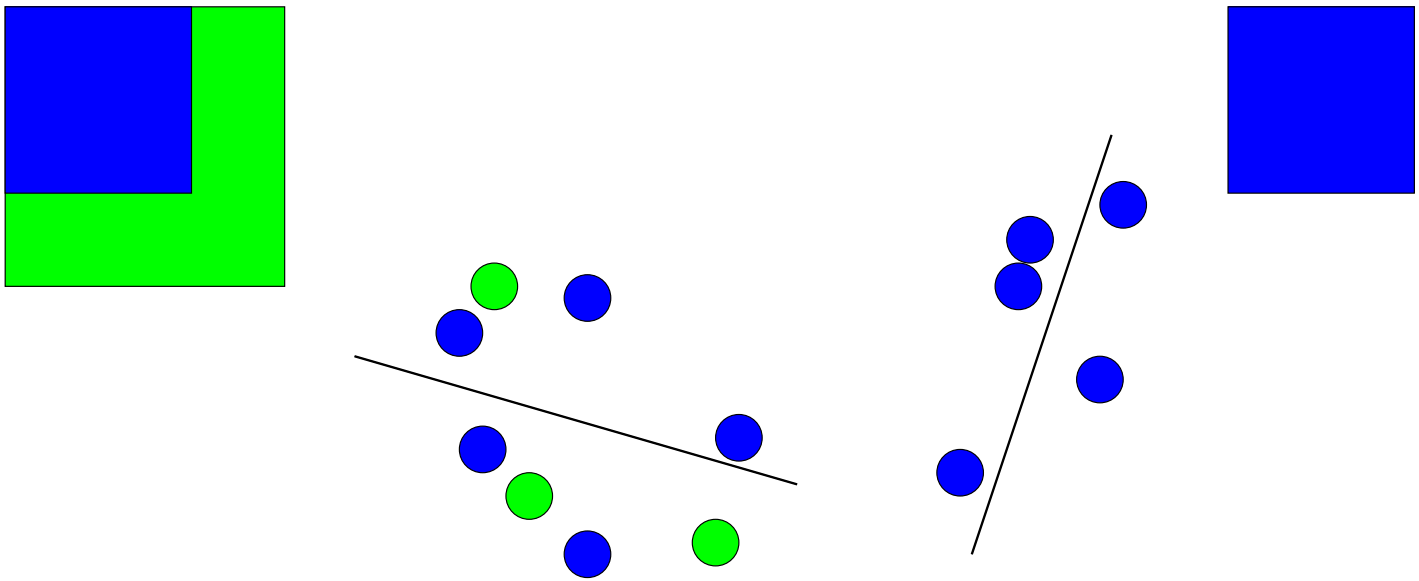
- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.



Kernel CCA approach (Y. et al., 2004)

The idea

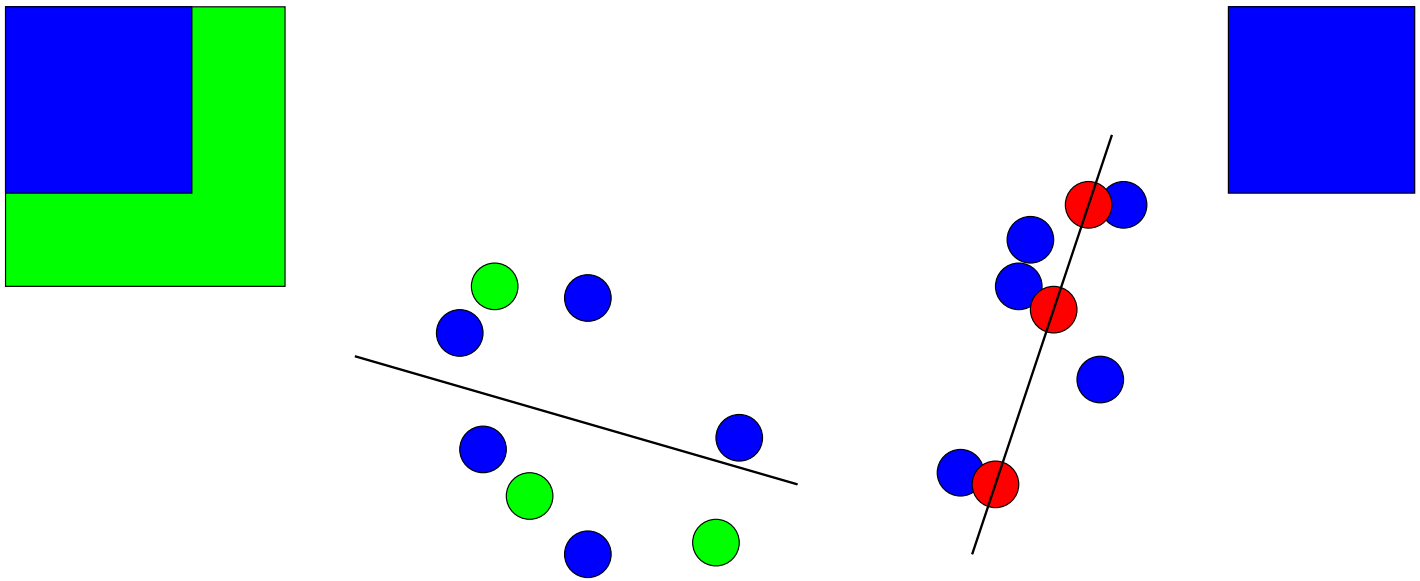
- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.



Kernel CCA approach (Y. et al., 2004)

The idea

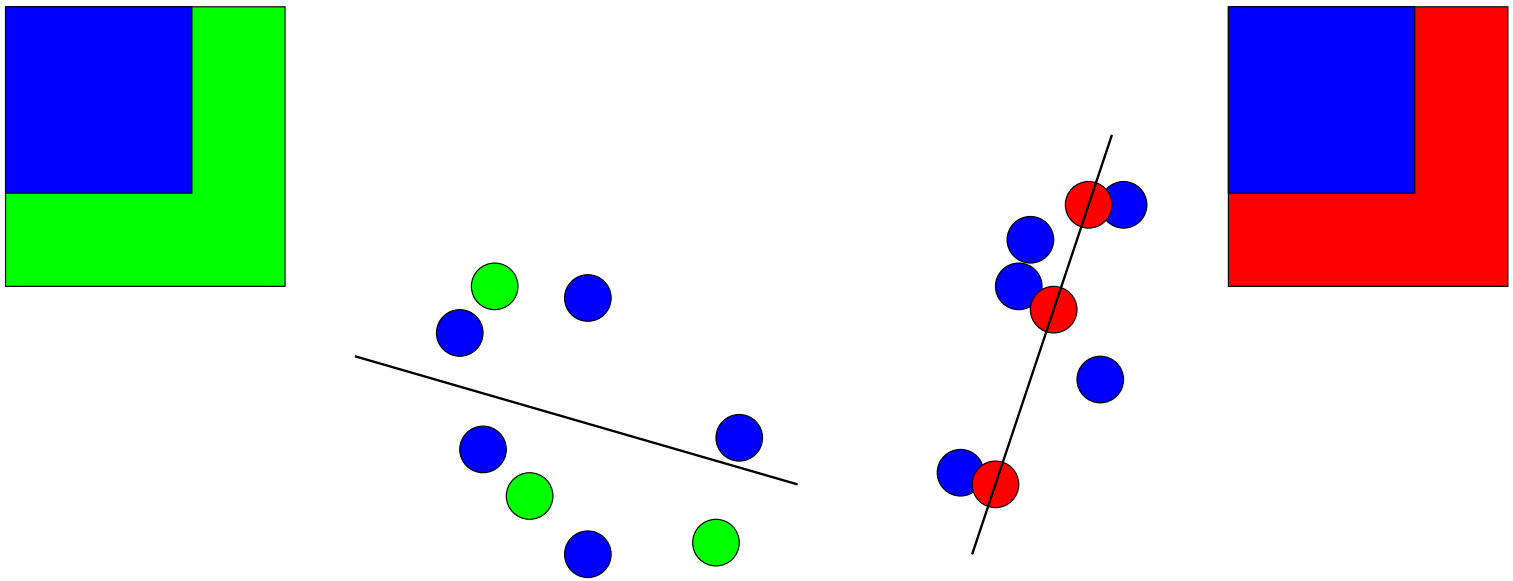
- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.



Kernel CCA approach (Y. et al., 2004)

The idea

- Consider the kernels as inner products
- Find low-dimensional projections of the t points in K_{tt} and G_{tt} with maximum correlation
- Project the last p points of K onto this subspace
- Estimate G_{tp} as the inner products in the projection from K.



Kernel matrix regression

The idea

- Directly try to predict the expensive kernel G from K, using the know part as training set.
- In order to learn a p.d. function we impose the model:

$$G(x, y) = \sum_i u_i(x) u_i(y) + \epsilon \,,$$

  for $u_i : \mathcal{X} \to \mathbb{R}$.
- We minimize the L_2 (Frobenius) norm of ϵ over the training set.

Kernel matrix regression

Solution

- We look for u_i in the RKHS of K:

$$u_i(x) = \sum_{i=1}^{n} \alpha_j^i K(x_j, x)\,.$$

- Minimization over the α's of

$$\| G_{tt} - K_{tt} A A^\top K_{tt} \|_{Fro}^2$$

gives:

$$A A^\top = K_{tt}^{-1} G_{tt} K_{tt}^{-1}\,,$$

and therefore:

$$G_{pt} = K_{pt} K_{tt}^{-1} G_{tt} \quad G_{pp} = K_{pt} K_{tt}^{-1} G_{tt} K_{tt}^{-1} K_{pt}^\top.$$

Regularized kernel matrix regression

Minimization over the α's of

$$\| G_{tt} - K_{tt} A A^\top K_{tt} \|^2_{Fro} + 2\lambda \sum_i \| u_i \|^2_{RKHS}$$

gives:

$$A A^\top = K_{tt}^{-1} \left(G_{tt} - \lambda K_{tt}^{-1} \right) K_{tt}^{-1} ,$$

and therefore:

$$G_{pt} = K_{pt} K_{tt}^{-1} \left(G_{tt} - \lambda K_{tt}^{-1} \right) \quad G_{pp} = K_{pt} K_{tt}^{-1} \left(G_{tt} - \lambda K_{tt}^{-1} \right) K_{tt}^{-1} K_{pt}^\top .$$

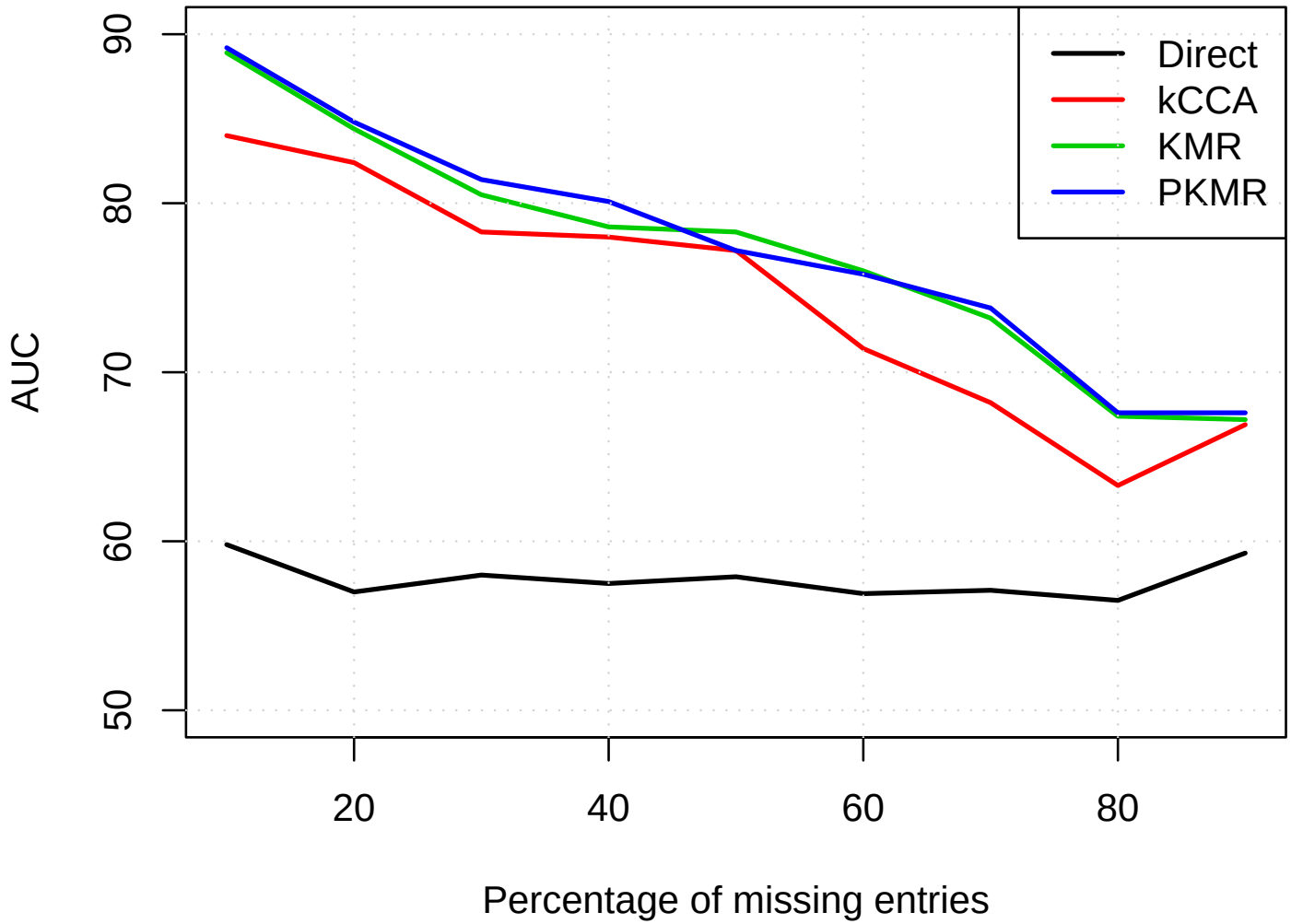
Outline

Biological network inference

Data

- Reconstruct the metabolic gene network, a biological network with 769 genes as vertices
- Use a kernel over genes deduced from gene expression data, phylogenetic profiles, cellular localization.
- The graph is embedded to a Hilbert space using a diffusion kernel
- 5-fold cross-validation, measure average AUC.

Biological network inference: Results



Outline

Conclusion

- A method to predict missing entries in a kernel Gram matrix using side information from another Gram matrix
- Objective function and regularization more adapted to the problem than kernel CCA