Scalable Matrix-valued Kernel Learning: Multivariate Regression and Granger Causality

Vikas Sindhwani
IBM Research, NY

With Minh Ha Quang (IIT, Genova) and Aurelie Lozano (IBM)

July 13, 2013
Uncertainty in Artificial Intelligence (UAI) 2013

Problem Setting

- Estimate, non-parametrically, an unknown non-linear dependency,

$$f : \mathcal{X} \mapsto \mathcal{Y},$$

  from labeled examples, where $\mathcal{Y}$ is a "structured" output space.

  - "Structure": multiple outputs; joint prediction more efficient.
  - $\mathcal{Y}$: Hilbert space structure $\langle \cdot, \cdot \rangle_{\mathcal{Y}}$, $\| \cdot \|_{\mathcal{Y}}$. Focus on $\mathcal{Y} \subseteq \mathbf{R}^n$.
  - Multivariate Regression, Multitask, Structured Output Learning.
  - Jointly learn f and the structure on $\mathcal{Y}$.

- Very natural to attempt to formulate as Tikhonov Regularization in *vector-valued* Reproducing Kernel Hilbert Spaces (RKHS):

$$\arg\min_{f \in \mathcal{H}} \|\mathbf{y}_i - f(\mathbf{x}_i)\|_{\mathcal{Y}}^2 + \lambda \|f\|_{\mathcal{H}}^2. \quad (1)$$

Challenges with vector-valued RKHS methods

- Long history : Laurent Schwartz (1964), Burbea and Masani (1984),. . . , MP(2005), but not as popular as scalar kernel methods.

- Kernel function $\overrightarrow{k}(\mathbf{x}, \boldsymbol{z})$, which encodes input and output structure, is $matrix\text{-}valued$. This makes model selection daunting.
  - By contrast, widely popular Gaussian or Polynomial scalar-valued kernels have just one hyperparameter.

- Computational Complexity: Ridge Regression in a general $\mathbf{R}^n$-valued RKHS with l labeled samples requires $O(l^3n^3)$ time and $O(l^2n^2)$ storage.

- To be able to even consider vector-valued RKHS methods for an application, we need **scalable matrix-valued kernel learning**.

Contributions and Outline

- Function estimation in vector-valued RKHS dictionaries - generalize scalar multiple kernel learning (MKL), structured sparsity algorithms.

- Full resolution of kernel learning for $separable$ matrix-valued kernels.

  – Eigendecomposition-free algorithms that orchestrate inexact solvers.

- Empirical Studies

  – Statistical effectiveness of matrix-valued kernel learning.
  – Computational effectiveness of using inexact solvers.

- Enable a new application: Non-linear Graphical Granger Causality.

- Generalization bounds based on Rademacher complexity for our vector-valued hypothesis sets (analogous to scalar MKL results).

Vector-valued RKHS [Michelli and Pontil, 2005]

- A Hilbert space $\mathcal{H}$ of functions mapping $\mathcal{X} \rightarrow \mathcal{Y}$ is a *vector-valued RKHS* if there is a function $\overrightarrow{k} : \mathcal{X} \times \mathcal{X} \rightarrow \mathcal{L}(\mathcal{Y})$ such that:

  1. For all $\mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y}$, the function

  $$\delta_{\mathbf{x},\mathbf{y}}(\cdot) = \overrightarrow{k}(., \mathbf{x})\mathbf{y} \in \mathcal{H} \tag{2}$$

  2. For all $f \in \mathcal{H}$, the **reproducing property** (RP) holds

  $$\langle f, \delta_{\mathbf{x},\mathbf{y}} \rangle_{\mathcal{H}} = \langle f(\mathbf{x}), \mathbf{y} \rangle_{\mathcal{Y}} \quad \forall \mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y} \tag{3}$$

- All niceness properties (theoretical and algorithmic) of RKHSs ultimately flow from the reproducing property.

Tikhonov Regularization in Vector-valued RKHS

$$\arg\min_{f \in \mathcal{H}_{\overrightarrow{k}}} \frac{1}{l} \sum_{i=1}^{l} \|f(\mathbf{x}_i) - \mathbf{y}_i\|_2^2 + \lambda \|f\|_{\mathcal{H}_{\overrightarrow{k}}}^2$$

- **Representer Theorem**: solution is the sum of linear transformations.

$$f(\cdot) = \sum_{i=1}^{l} \overrightarrow{k}(\cdot, \mathbf{x}_i)\boldsymbol{\alpha}_i, \quad \text{where} \quad \boldsymbol{\alpha}_i \in \mathbf{R}^n$$

- Huge RLS linear system of size $ln \times ln$:

$$\left(\overrightarrow{\mathbf{K}} + \lambda l \mathbf{I}_{nl}\right) vec(\mathbf{C}^T) = vec(\mathbf{Y}^T)$$

where with $\overrightarrow{\mathbf{K}}_{ij} = \overrightarrow{k}(\mathbf{x}_i, \mathbf{x}_j) \in \mathbf{R}^{n \times n}$ and $\mathbf{C} = [\boldsymbol{\alpha}_1 \ldots \boldsymbol{\alpha}_l]^T \in \mathbf{R}^{l \times n}$

Separable Matrix-valued Kernels

- $\overrightarrow{k}(\mathbf{x}, \boldsymbol{z}) = k(\mathbf{x}, \boldsymbol{z})\mathbf{L}$
  - k is a scalar *input kernel* function and $\mathbf{L}$ is an $n \times n$ symmetric positive-definite *output kernel* matrix.
  - Simplicity, universality, extensibility and potential for scalability.
  - We use the notation $\mathcal{H}_{k\mathbf{L}}$ for the associated RKHS
  - If $f = (f_1 \dots f_n) \in \mathcal{H}_{k\mathbf{L}}$, then each scalar component $f_i \in \mathcal{H}_k$

- Regularization: $\|f\|^2_{\mathcal{H}_{k\mathbf{L}}} = \sum_{ij} \left(\mathbf{L}^\dagger\right)_{ij} \langle f_i, f_j \rangle_{\mathcal{H}_k}$ where $f = (f_1 \dots f_n)$.
  - Suppose $\boldsymbol{G}$ is the adjacency matrix of an output similarity graph and $\mathbf{M}$ is its Graph Laplacian. Then, for $\mathbf{L} = \mathbf{M}^\dagger$,

$$\|f\|^2_{\mathcal{H}_{k\mathbf{L}}} = \frac{1}{2} \sum_{i,j=1}^{n} \|f_i - f_j\|^2_{\mathcal{H}_k} G_{ij} + \sum_{i=1}^{n} \|f_i\|^2_{\mathcal{H}_k} G_{ii}$$

Ridge Regression with Separable Matrix-valued Kernels

- Regularized Least Squares solution can be written in two ways:

$$\begin{aligned} (\mathbf{K} \otimes \mathbf{L} + \lambda l \mathbf{I}_{nl})\, vec(\mathbf{C}^T) &= vec(\mathbf{Y}^T), & (4) \\ \mathbf{KCL} + \lambda l \mathbf{C} &= \mathbf{Y}. & (5) \end{aligned}$$

  - $O(l^2 + n^2)$ storage instead of $O(l^2 n^2)$
  - $O(n^3 + l^3)$ time instead of $O(l^3 n^3)$.

- $O(l^3 + n^3)$ Sylvester solver based on Eigendecomposition:
  - $\mathbf{K} = \mathbf{TMT}^T$ where $\mathbf{M} = diag(\sigma_1 \ldots \sigma_l)$
  - $\mathbf{L} = \mathbf{SNS}^T$ where $\mathbf{N} = diag(\rho_1 \ldots \rho_n)$
  - Solution: $\mathbf{C} = \mathbf{T}\tilde{\mathbf{X}}\mathbf{S}^T \quad \text{where} \quad \tilde{\mathbf{X}_{ij}} = \frac{\left(\mathbf{T}^T\mathbf{YS}\right)_{ij}}{\sigma_i \rho_j + \lambda}$.

Output Kernel Learning

- An extended RLS problem: also optimize $\mathbf{L}$ over PSD cone.

$$\arg\min_{f \in \mathcal{H}_{k\mathbf{L}}, \mathbf{L} \in \mathcal{S}_+^n} \frac{1}{l} \sum_{i=1}^{l} \|f(\mathbf{x}_i) - \mathbf{y}_i\|_2^2 + \lambda \|f\|_{\mathcal{H}_{\vec{k}}}^2 + \rho \|\mathbf{L}\|_{fro}^2$$

- Dinuzzo et. al.'s (ICML, 2011) Block Coordinate Descent approach
  - $\mathbf{L}$ fixed: Solve Sylvester equations for f by Eigendecomposition.
  - f fixed: optimize over $\mathbf{L} \in \mathbf{R}^{n\times n}$ leading to a linear system, or over $\mathbf{L} \in \mathcal{S}^n$ leading to another Sylvester equation.

- Three issues with this approach:
  - $\mathbf{L}$ updates hold only if f is solved exactly (Eigensolver).
  - PSD constraints are not provably satisfied.
  - Particularly expensive if scalar kernel is also being optimized.

Learning over a vector-valued RKHS dictionary

- Goals: Fuller resolution of separable kernel learning problem with eigendecomposition-free scalable solvers.
- Setup a dictionary of separable matrix-valued *base* kernels $\mathcal{D}_{\mathbf{L}} = \{k_1\mathbf{L}, \ldots k_m\mathbf{L}\}$ and define a space of functions expressible as sums of component functions drawn from RKHSs in $\mathcal{D}_{\mathbf{L}}$:

$$\mathcal{H}(\mathcal{D}_{\mathbf{L}}) = \left\{ f = \sum_{j=1}^{m} f_j : f_j \in \mathcal{H}_{k_j\mathbf{L}} \right\} \tag{6}$$

- *Functional sparsity* of f: number of non-zero component functions.
- Our objective function (for large m, need RKHS structure again):

$$\arg\min_{f\in\mathcal{H}(\mathcal{D}_{\mathbf{L}}),\mathbf{L}\in\mathcal{S}_+^n(\tau)} \frac{1}{l}\sum_{i=1}^{l} \|f(\mathbf{x}_j) - \mathbf{y}_i\|_2^2 + \lambda\Omega[f], \tag{7}$$

Variationally defined Regularizers $\Omega[f]$

- l_p regularizers:

$$\Omega[f] = \|f\|_{l_p(\mathcal{H}(\mathcal{D}_{\mathbf{L}}))} = \min_{f: f=\sum_j f_j} \left\| \left(\|f_1\|_{\mathcal{H}_{k_1 \mathbf{L}}}, \ldots, \|f_m\|_{\mathcal{H}_{k_m \mathbf{L}}} \right) \right\|_p$$

  - $p \to 1$: induces *functional sparsity* (generalization of group Lasso).

  - $p \to 2$: non-sparse combinations.

- Broader class of regularizers that admit variational representations:

$$\Omega(f) = \min_{\boldsymbol{\eta} \in \mathbf{R}_+^m} \sum_{i=1}^{m} \frac{\|f_i\|^2_{\mathcal{H}_{k_i \mathbf{L}}}}{\eta_i} + \omega(\boldsymbol{\eta}) \tag{8}$$

For l_1, the auxillary function $\omega(\boldsymbol{\eta})$ is indicator function for simplex.

Learning convex combinations of base kernels

- Variational regularizers relate non-differentiable mixed norms to weighted sum of RKHS norms, which further is equivalent to learning with a single kernel given by a convex combination of base kernels.

  Proposition 1. *The function:* $\overrightarrow{k}_{\boldsymbol{\eta}} = \sum_{i=1}^{m} \eta_i \overrightarrow{k}_i$, *is the reproducing kernel of the sum space, with norm:*

$$\|f\|^2_{\mathcal{H}_{\overrightarrow{k}_{\boldsymbol{\eta}}}} = \min_{f=\sum_{j=1}^{m} f_i, f_j \in \mathcal{H}_{\overrightarrow{k}_j}} \sum_{j=1}^{m} \frac{\|f_j\|^2}{\eta_j}.$$

- Important for handling large m.
- Generalizes analogous results in the scalar MKL literature.
- Joint optimization: scalar kernel weights $\boldsymbol{\eta}$, $\mathbf{L}$ and $f \in \mathcal{H}_{k_{\boldsymbol{\eta}}\mathbf{L}}$.

Spectahedron Constraints on Output Kernel $\mathbf{L}$

- $\mathbf{L} \in \mathcal{S}_+^n(\tau)$ - a semi-definite analogue of the simplex.

$$\mathcal{S}_+^n(\tau) = \{\mathbf{X} \in \mathcal{S}_+^n | trace(\mathbf{X}) \leq \tau\}$$

- Rationale (besides low-rankness encouraged by trace norm):
  - Can use a specialized sparse SDP solver whose iterations involve computing a single extremal eigenvector of the gradient inexactly.
  - Implies that a Conjugate Gradient iterative solver for f optimization encounters numerically well-conditioned problems.
  - Trace constraint parameter naturally appears in Generalization bounds based on Rademacher complexity.

Block Coordinate Descent

- Finite dimensional version of the optimization problem:

$$\arg\min_{\mathbf{C}\in\mathbf{R}^{n\times l},\mathbf{L}\in\mathcal{S}_{+}^{n}(\tau),\boldsymbol{\eta}\in\mathbf{R}_{+}^{m}} \frac{1}{l}\|\mathbf{K}_{\boldsymbol{\eta}}\mathbf{C}\mathbf{L}-\mathbf{Y}\|_F^2$$

$$+\lambda\ trace\left(\mathbf{C}^T\mathbf{K}_{\boldsymbol{\eta}}\mathbf{C}\mathbf{L}\right)+\omega(\boldsymbol{\eta}). \tag{9}$$

  - Optimize $\mathbf{C}$ with Conjugate-Gradient Sylvester solver.
  - Optimize $\boldsymbol{\eta}$ using closed form update rules (akin to scalar MKL).
  - Optimize $\mathbf{L}$ using a specialized sparse SDP solver.

- Vector-valued Prediction function:

$$f^{\star}(\mathbf{x})=\mathbf{L}\mathbf{C}^T[k_{\boldsymbol{\eta}}(\mathbf{x},\mathbf{x}_1)\ldots k_{\boldsymbol{\eta}}(\mathbf{x},\mathbf{x}_l)]^T \tag{10}$$

Sylvester Solver based on Conjugate Gradient

- Use iterative CG solver directly on:

$$(\mathbf{K}_\eta \otimes \mathbf{L} + \lambda l \mathbf{I}_{nl})\, vec(\mathbf{C}^T) = vec(\mathbf{Y}^T) \tag{11}$$

  - can exploit warm-starts from previous solution.
  - coefficient matrix need not be materialized
  - fast matrix-vector products $O(nl(l+n))$:

$$(\mathbf{K}_\eta \otimes \mathbf{L} + \lambda l \mathbf{I}_{nl}) vec(\mathbf{C}^{(k)T}) = vec(\mathbf{K}_\eta \mathbf{C}^{(k)} \mathbf{L} + \lambda l \mathbf{C}^{(k)}) \tag{12}$$

  - can exploit structure, e.g., $\mathbf{K}$ is low-rank or sparse
  - can be used for more general problems involving $\sum_i \mathbf{K}_i \otimes \mathbf{L}_i$

CG Sylvester Solver

Proposition 2 (Convergence Rate for CG Sylvester solver). *Assume l_1 norm for Ω. Let $\mathbf{C}^{(k)}$ be the CG iterate at step k, $\mathbf{C}^\star$ be the optimal solution (at current fixed $\boldsymbol{\eta}$ and $\mathbf{L}$) and $\mathbf{C}^{(0)}$ be the initial iterate (warm-started from previous value). Then,*

$$\|\mathbf{C}^{(k)} - \mathbf{C}^*\|_F \leq 2\sqrt{\phi}\left(\frac{\sqrt{\phi}-1}{\sqrt{\phi}+1}\right)^k \|\mathbf{C}^{(0)} - \mathbf{C}^*\|_F, \tag{13}$$

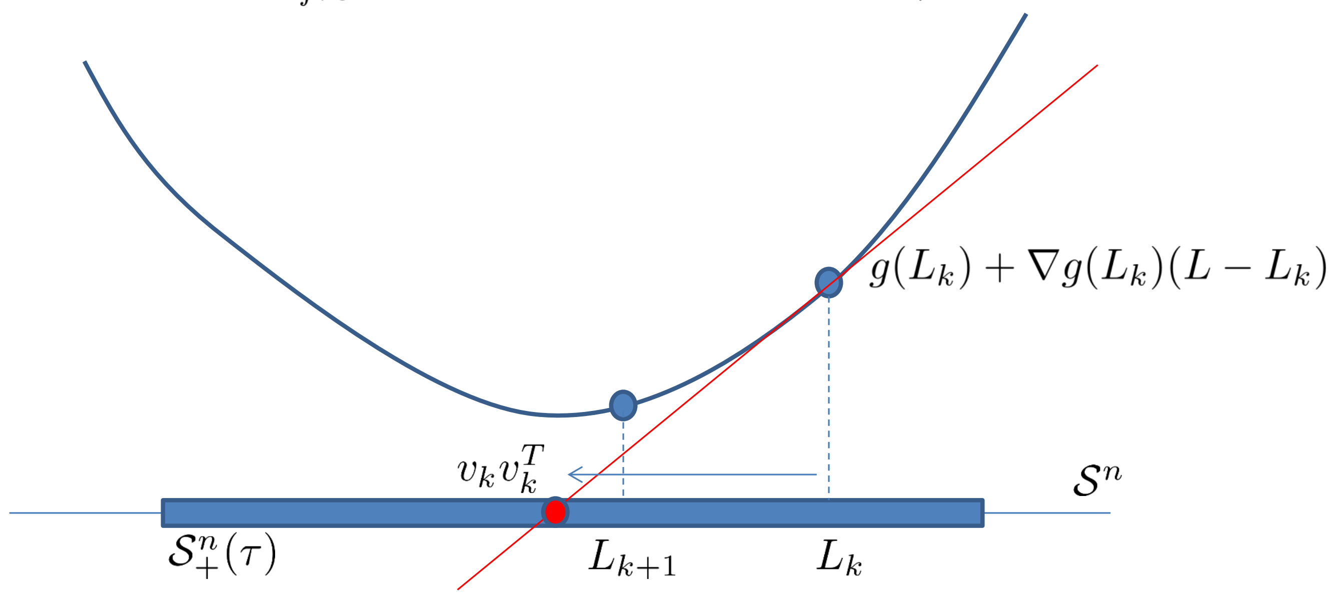
where $\phi = 1 + \frac{\gamma\tau}{l\lambda}$ with $\gamma = \max_i \|\mathbf{K}_i\|_2$. For dictionaries involving only Gaussian scalar kernels, the condition number is bounded as:

$$\phi \leq 1 + \frac{\tau}{\lambda}, \tag{14}$$

i.e., the convergence rate depends only on the relative strengths of regularization parameters λ, τ.

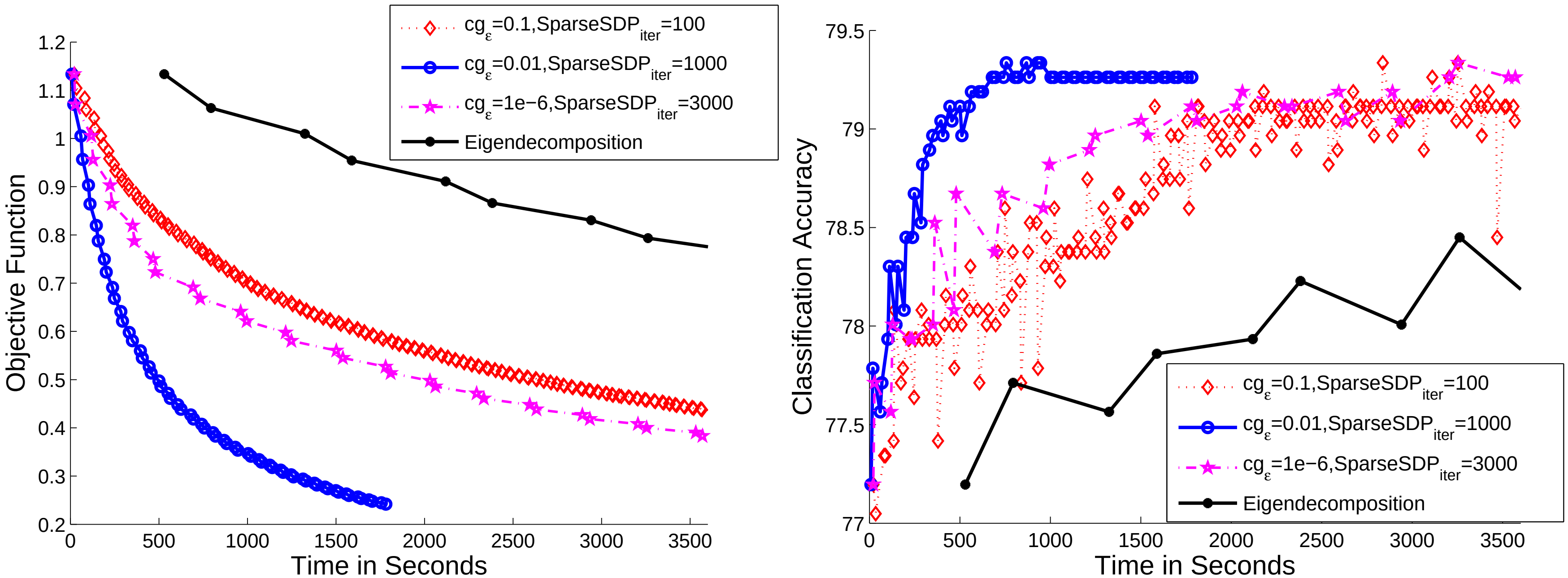
Sparse SDP solver for $\mathbf{L}$ [Hazan, 2008]

$g(L) = ||AL - Y||_{fro}^2 + \lambda \mathrm{trace}(B^T L)$ where $A = K_\eta C, B = C^T A$



- Adaptation: bounded trace, exact line search, analysis.
- Inexact eigenvector computation via truncated power method.
- **Proposition**: *Assume* l_1 *norm. For* $k \geq 16(\tau\gamma)^2/\epsilon$, $g(\mathbf{L}^{(k+1)}) - g(\mathbf{L}^\star) \leq \epsilon/2$ *where* $\gamma = \max_i \|\mathbf{K}_i\|_2$.

Cheap iterations using inexact numerical optimization



- Tradeoff: Many, cheap iterations versus few, expensive iterations.
- Caltech101: 3060 training, 1355 test images, $p = 1.7$, $\lambda = 0.001$
- Inexact solvers at the right make rapid progress towards highly competitive models.

Statistical Performance: VAR Financial Models

Table 1: VAR prediction of log-returns of 9 stocks.

	OLS	Lasso	MRCE	FES	IKL	OKL	IOKL
WMT	0.98	0.42	0.41	0.40	0.43	0.43	0.44
XOM	0.39	0.31	0.31	0.29	0.32	0.31	0.29
GM	1.68	0.71	0.71	0.62	0.62	0.59	**0.47**
Ford	2.15	0.77	0.77	0.69	0.56	0.48	**0.36**
GE	0.58	0.45	0.45	0.41	0.41	0.40	**0.37**
COP	0.98	0.79	0.79	0.79	0.81	0.80	**0.76**
Ctgrp	0.65	0.66	0.62	0.59	0.66	0.62	**0.58**
IBM	0.62	0.49	0.49	0.51	0.47	0.50	**0.42**
AIG	1.93	1.88	1.88	1.74	1.94	1.87	1.79
Average	1.11	0.72	0.71	0.67	0.69	0.67	**0.61**

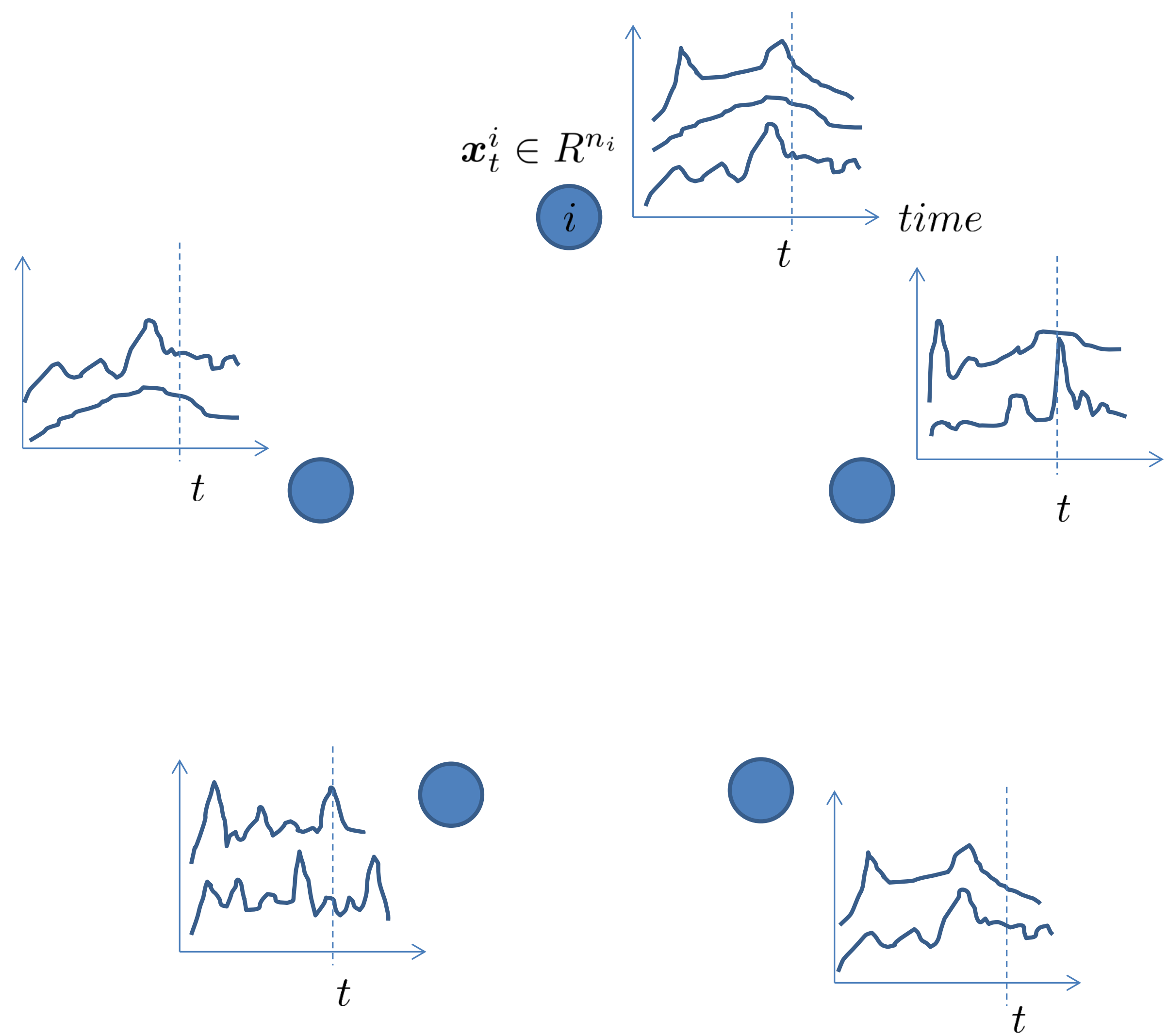
- Joint kernel learning better than scalar MKL and OKL alone.
- Dictionary of 117 Gaussian kernels (9 dimensions x 13 bandwidths)
- 13 kernels selected in IOKL.
- Comparisons: Independent OLS, Lasso, MRCE: Multivariate regression with error (inv) covariance estimation, FES: (Linear) Multivariate regression with Trace norm penalty on coefficients.

Figure 1: Output kernel matrix $\mathbf{L}$

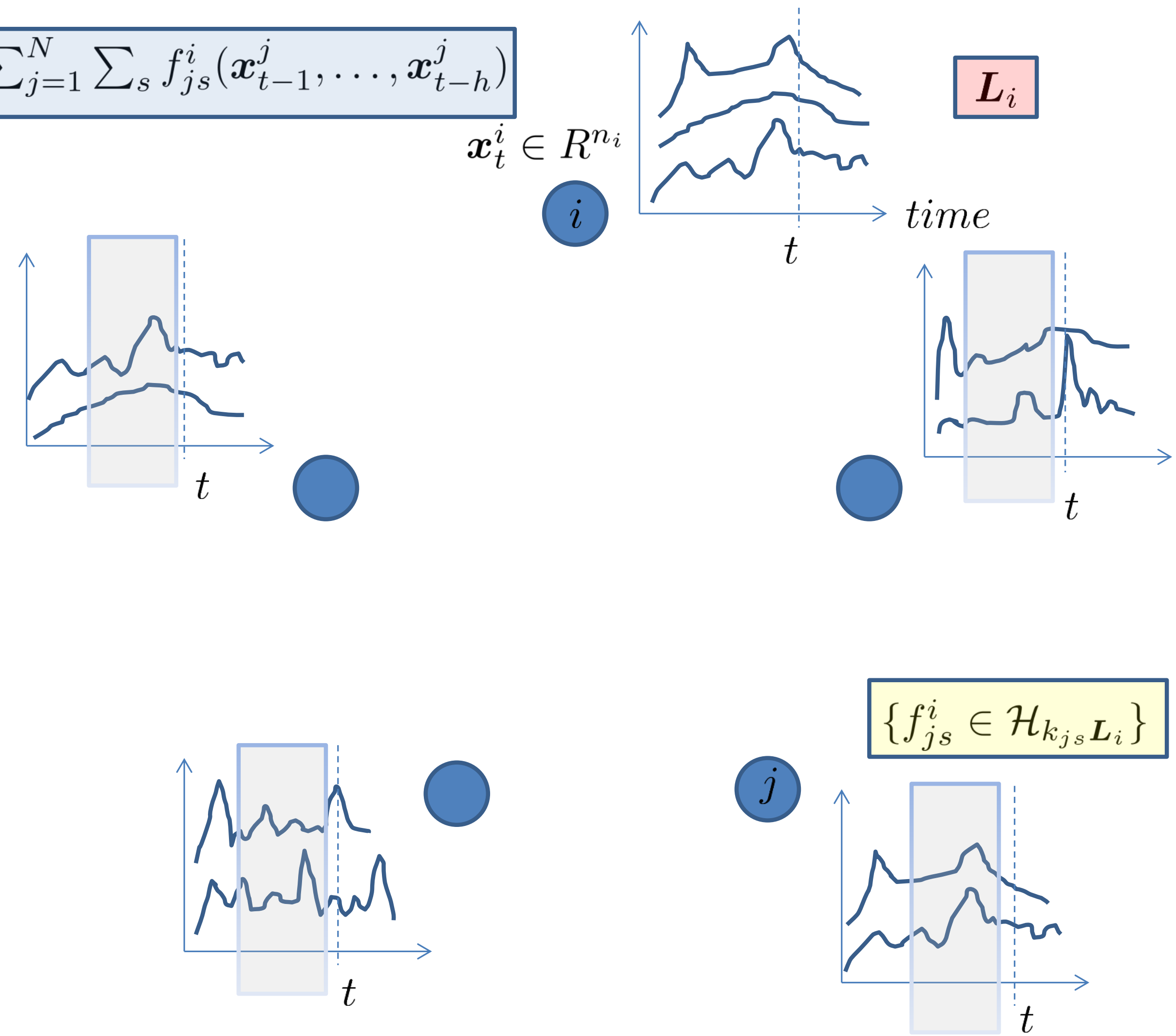
	Walmart	Exxon	GM	Ford	GE	ConocoPhillips	Citigroup	IBM	AIG
Walmart	0.26	0.11	0.60	0.76	0.26	0.17	0.25	0.22	0.27
Exxon	0.11	0.27	0.19	0.24	0.23	0.31	0.16	0.17	0.31
GM	0.60	0.19	2.22	2.67	0.82	0.35	0.79	0.68	0.76
Ford	0.76	0.24	2.67	3.72	0.99	0.52	0.75	0.63	0.96
GE	0.26	0.23	0.82	0.99	0.46	0.36	0.38	0.35	0.48
ocoPhillips	0.17	0.31	0.35	0.52	0.36	0.55	0.18	0.21	0.46
Citigroup	0.25	0.16	0.79	0.75	0.38	0.18	0.48	0.42	0.37
IBM	0.22	0.17	0.68	0.63	0.35	0.21	0.42	0.46	0.36
AIG	0.27	0.31	0.76	0.96	0.48	0.46	0.37	0.36	0.59

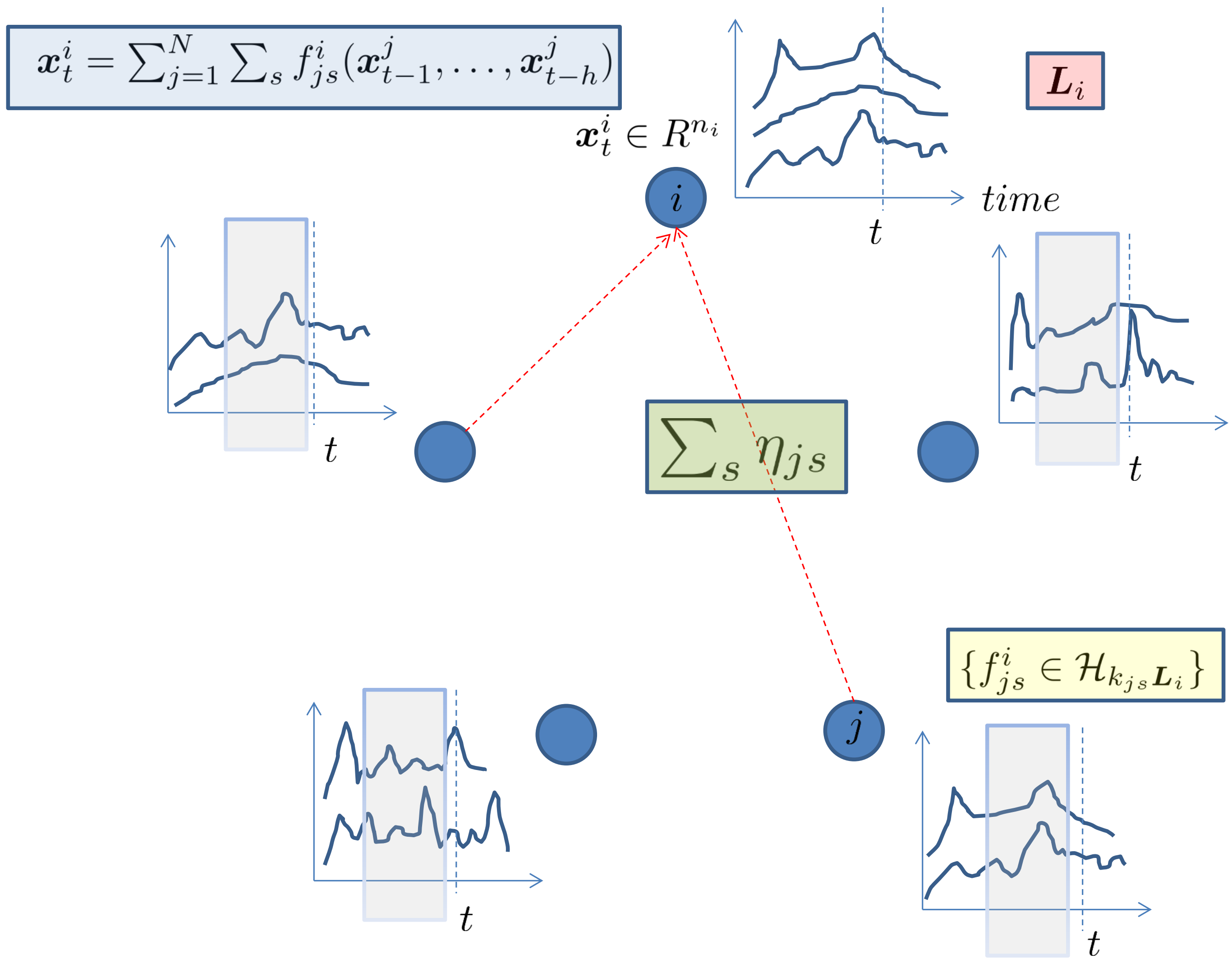
Application: Non-linear Granger Causality

- Given observations from an interconnected system of N distinct sources (nodes) of high-dimensional time series data, infer causal relationships between nodes.
- **Granger Causality** [Granger, 1980]: *If past evolution of a subset of nodes A_i is predictive of the future evolution of node i, more so than the past values of i alone, then A_i is said to causally influence i collectively.*
- Operationalizes causality by linking it to prediction. Caveat: causal insight is bounded by prediction accuracy.
- Sparsity - a natural prior, particularly in a nonlinear functional sense.


$\boldsymbol{x}_t^i \in R^{n_i}$
i
$time$
t
t
t
t
t

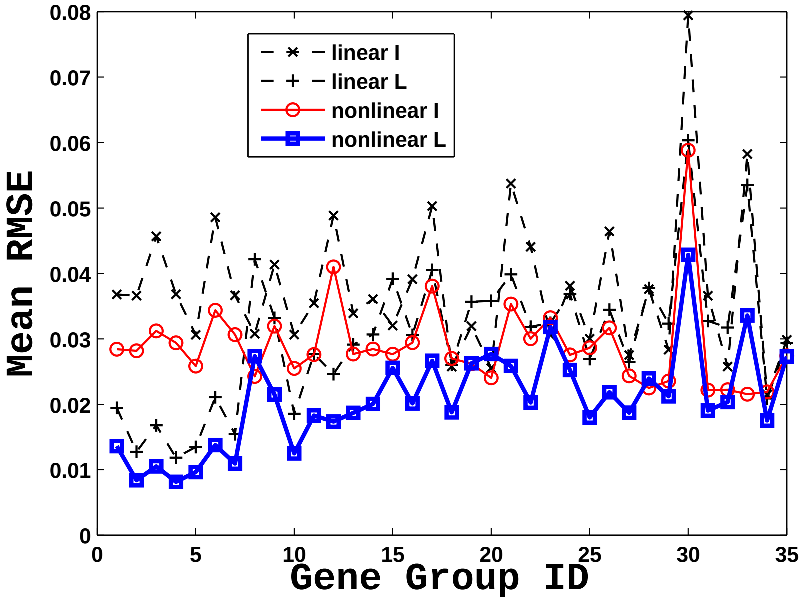
$$\boldsymbol{x}_t^i = \sum_{j=1}^{N} \sum_s f_{js}^i(\boldsymbol{x}_{t-1}^j, \ldots, \boldsymbol{x}_{t-h}^j)$$

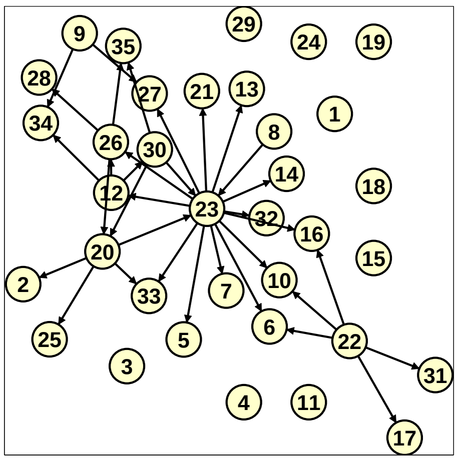


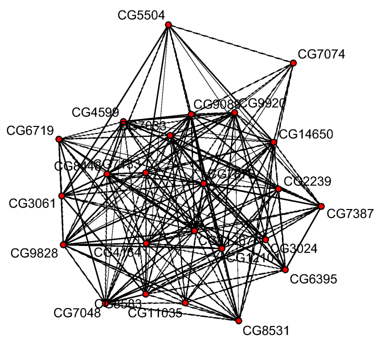

$\boldsymbol{x}_t^i = \sum_{j=1}^N \sum_s f_{js}^i(\boldsymbol{x}_{t-1}^j, \ldots, \boldsymbol{x}_{t-h}^j)$
$\boldsymbol{x}_t^i \in R^{n_i}$
$\boldsymbol{L}_i$
$time$
t
i
$\sum_s \eta_{js}$
$\{f_{js}^i \in \mathcal{H}_{k_{js}\boldsymbol{L}_i}\}$
j

Non-linear Granger Causality: Gene Network Inference

- Data: Gene expression levels for full life-cycle of Drosophilia. 2397 genes in 35 functional groups.
- Goal: Infer causal relationships between Gene groups and within-group.







- Full kernel learning gives best predictive (causal) performance.
- Causal Graph reveals centrality of a group not found by linear models.

Summary

- Goal: to make vector-valued RKHS methods more practical
  - Scalable Kernel learning techniques for separable matrix kernels
  - Selection and design of inexact solvers
  - Applications to high-dimensional causal inference problems
  - Generalized scalar MKL algorithms and theory
- Lots of open algorithm design problems:
  - Better solvers: pre-conditioned CG, first order SDPs
  - Extensions to non-separable matrix-valued kernels, .e.g., $\sum_j k_j \mathbf{L}_j$, $\overrightarrow{k}(\mathbf{x}, \boldsymbol{z})_{ij} = k(T_i\mathbf{x}, T_j\boldsymbol{z})$, Hessian of Gaussian kernel.
  - Scalability via randomized approximations.
  - Functional Regression and other non-$\mathbf{R}^n$ problems.
  - Connections to mean embeddings of conditional distributions.