Advanced data-driven modelling

Johan Suykens

KU Leuven, ESAT-STADIUS
Kasteelpark Arenberg 10
B-3001 Leuven (Heverlee), Belgium
Email: johan.suykens@esat.kuleuven.be
http://www.esat.kuleuven.be/stadius/

Optec retreat, Spa 2015

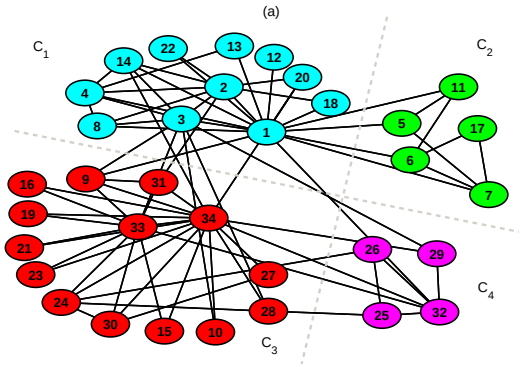
Outline

- Data world
- Challenges
- Sparsity in parametric and kernel-based models
- Learning with primal and dual model representations: a unifying picture

Introduction and motivation

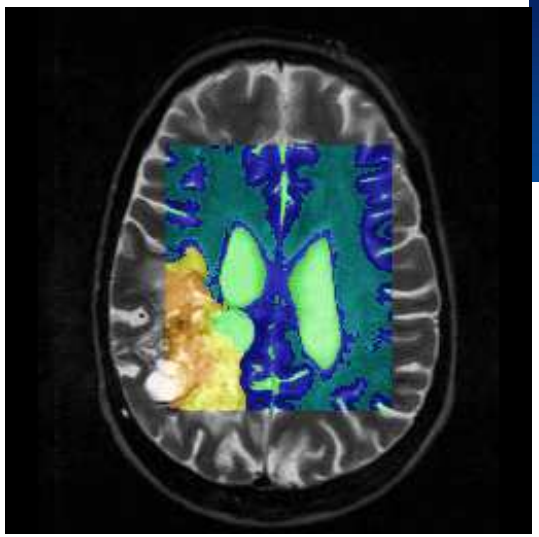
Data world

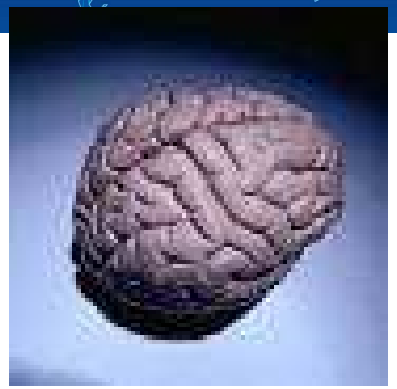



(a)
C1
C2
C3
C4
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34







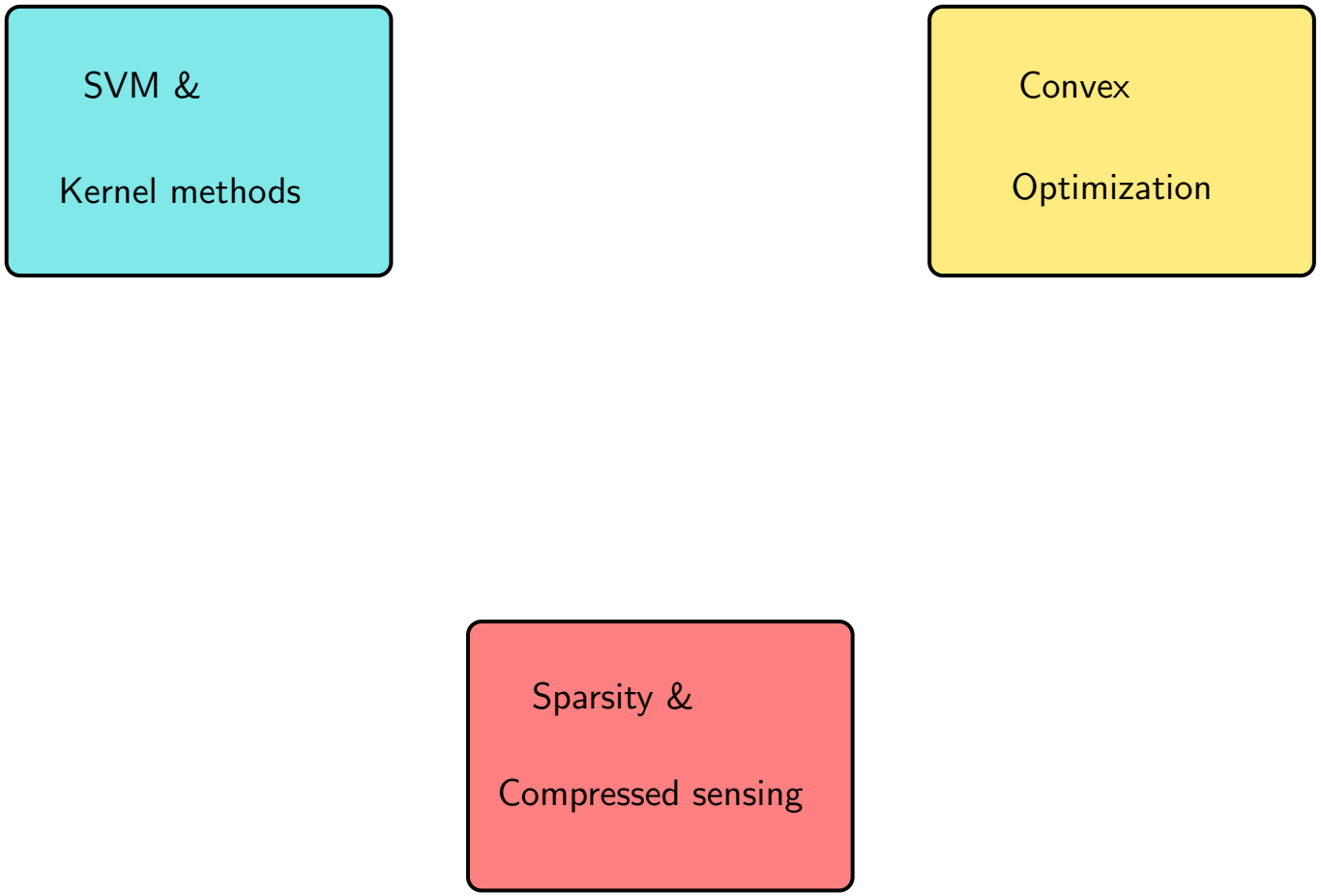



CampusInsight
MIND
SPELLER
AWARD-WINNING DEVICE CONVERTS BRAIN SIGNALS INTO WORDS
KATHOLIEKE UNIVERSITEIT
LEUVEN

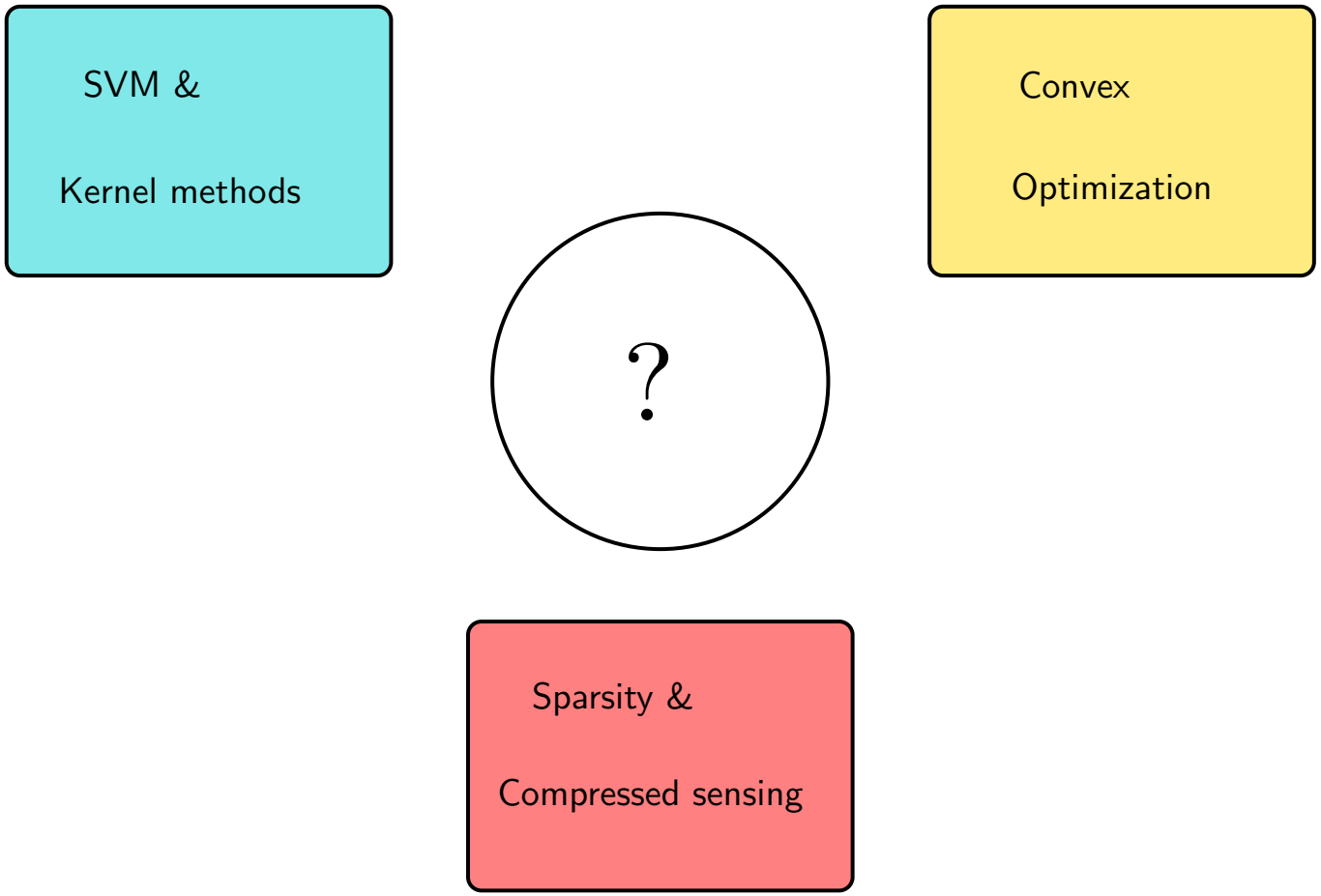
Challenges

- data-driven
- general methodology
- scalability
- need for new mathematical frameworks

Different paradigms



Different paradigms



Sparsity through regularization or loss function

Sparsity: through regularization or loss function

- through regularization: model $\hat{y} = w^T x + b$

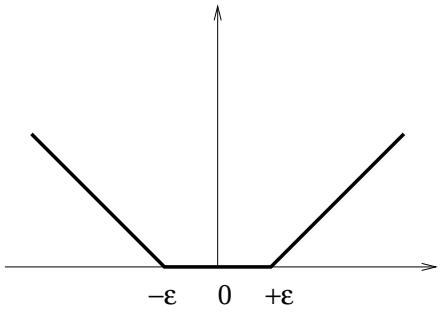
$$\min \sum_j |w_j| + \gamma \sum_i e_i^2$$

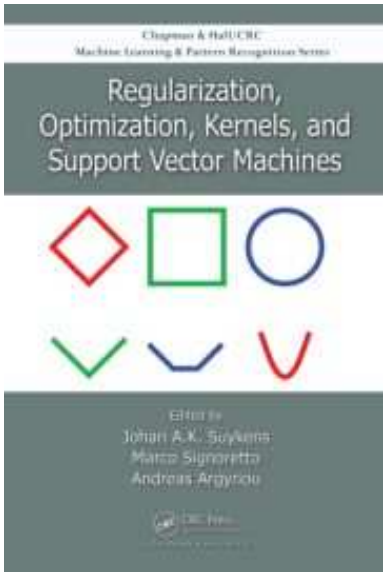
$\Rightarrow$ sparse w

- through loss function: model $\hat{y} = \sum_i \alpha_i K(x, x_i) + b$

$$\min \; w^T w + \gamma \sum_i L(e_i)$$

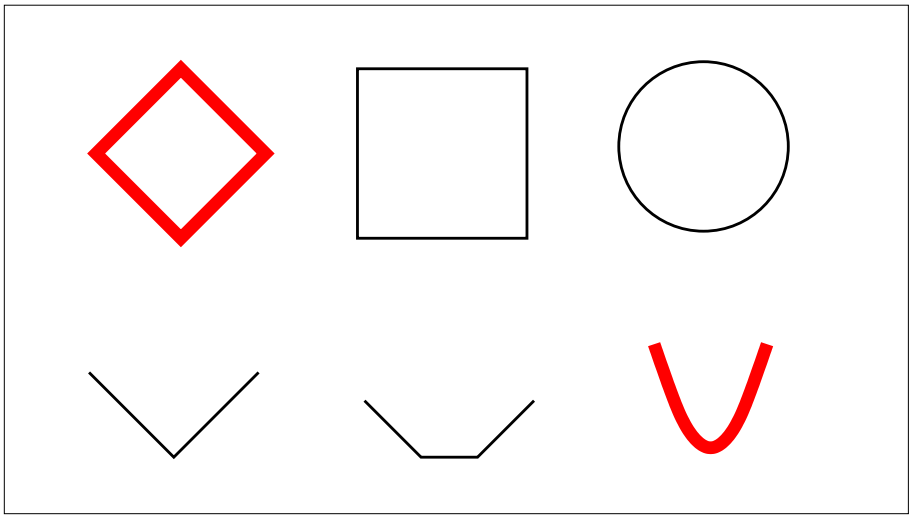
$\Rightarrow$ sparse α



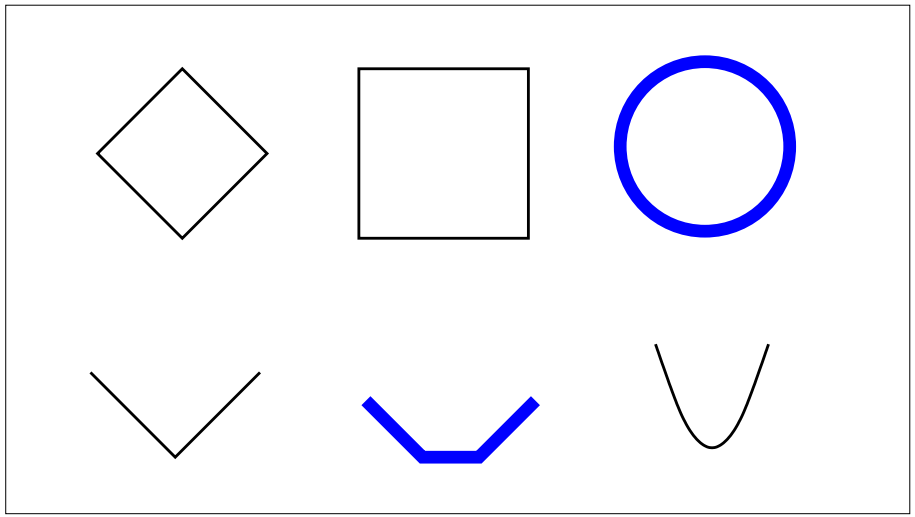


J.A.K. Suykens, M. Signoretto, A. Argyriou (Eds.)
Regularization, Optimization, Kernels, and Support Vector Machines,
Chapman & Hall/CRC, Machine Learning & Pattern Recognition Series,
Boca Raton US, Oct. 2014, 525 pp.

Regularization term and loss function

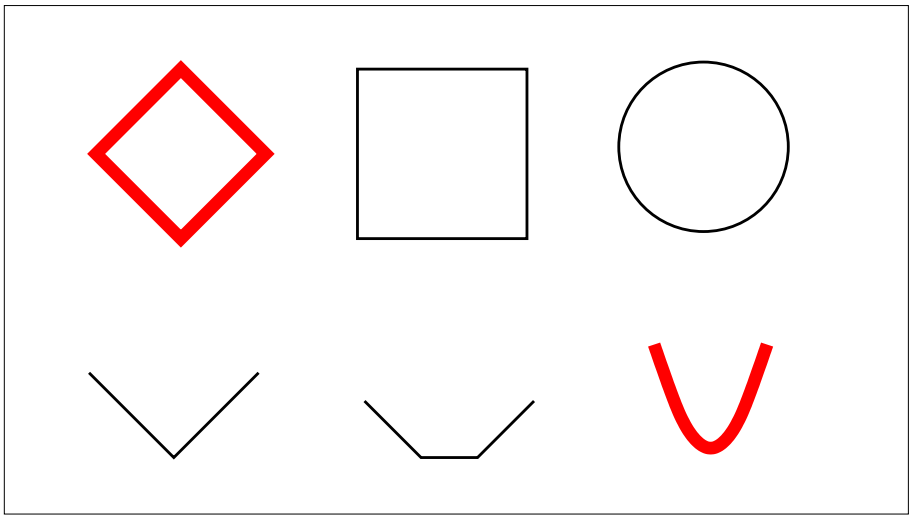


LASSO

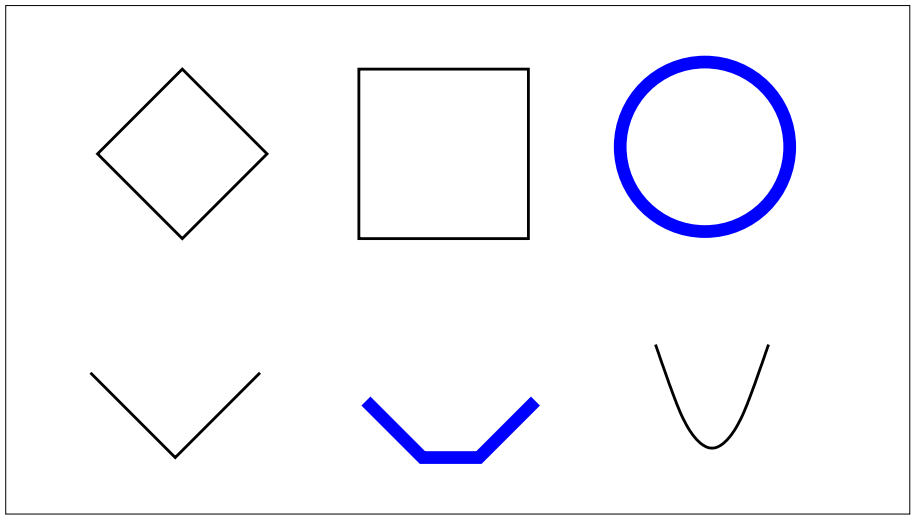


SVM

Regularization term and loss function



LASSO



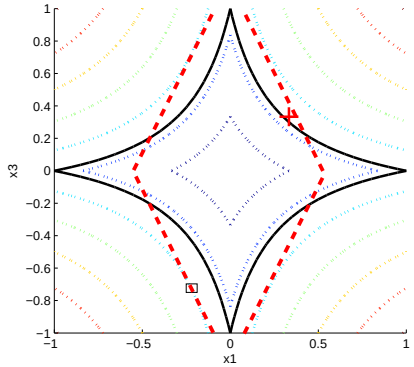
SVM

M. Jaggi, "An Equivalence between the Lasso and Support Vector Machines", Chapter 1 in *Regularization, Optimization, Kernels, and Support Vector Machines*, J.A.K. Suykens, M. Signoretto, A. Argyriou (Eds.), Chapman & Hall/CRC, Machine Learning & Pattern Recognition Series, Boca Raton US, 2014.

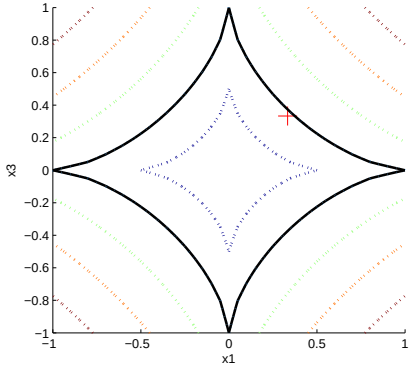
[Talk - C.M. Alaiz]

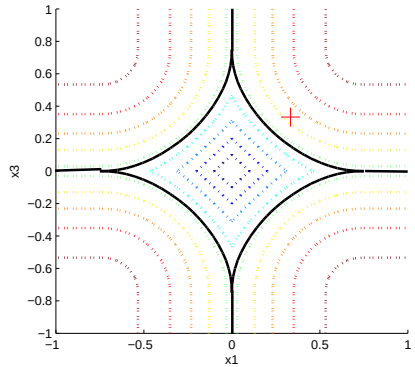
Sparser solutions

reweighted ℓ_1-norm

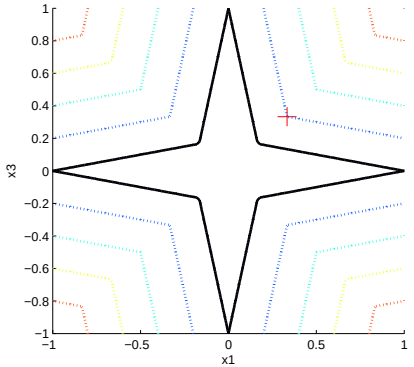


ℓ_q-norm with $q = 2/3$





smoothly clipped absolute deviation



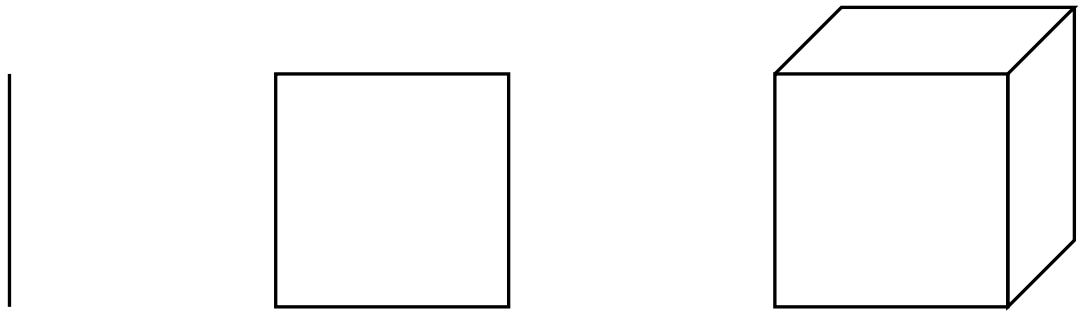
two-level ℓ_1

two-level ℓ_1 [Huang, Liu, Shi, Van Huffel, Suykens, SP 2015]

Large scale optimization

- Chunking and decomposition methods
- Sequential minimal optimization (SMO)
- Distributed optimization
- Coordinate descent method
- Frank-Wolfe method **[Talk - E. Frandi]**
- Stochastic learning **[Demo Salsa - R. Langone & V. Jumutc]**
- Ensemble methods

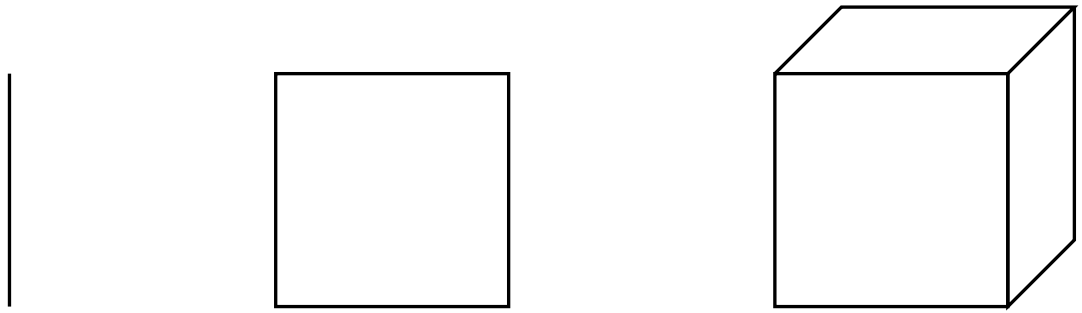
Sparsity: matrices and tensors



vector x matrix X tensor $\mathcal{X}$

data vector x		data matrix X		data tensor $\mathcal{X}$
vector model:	$\longrightarrow$	matrix model:	$\longrightarrow$	tensor model:
$\hat{y} = w^T x$		$\hat{y} = \langle W, X \rangle$		$\hat{y} = \langle \mathcal{W}, \mathcal{X} \rangle$

Sparsity: matrices and tensors



vector x matrix X tensor $\mathcal{X}$

data vector x		data matrix X		data tensor $\mathcal{X}$
vector model:	$\longrightarrow$	matrix model:	$\longrightarrow$	tensor model:
$\hat{y} = w^T x$		$\hat{y} = \langle W, X \rangle$		$\hat{y} = \langle \mathcal{W}, \mathcal{X} \rangle$
sparsity:		sparsity:		sparsity:
$\sum_j \lvert w_j \rvert$		$\lVert W \rVert_*$		$\lVert \mathcal{W} \rVert_*$

Learning with tensors [Signoretto, Tran Dinh, De Lathauwer, Suykens, ML 2014]

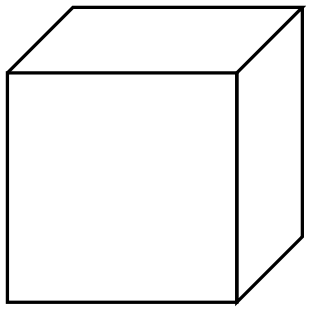
Robust tensor completion [Yang, Feng, Suykens, 2014] - **[Talk - Y. Yang]**

Kernels and Tensors

neuroscience: EEG data
$(\text{time samples} \times \text{frequency} \times \text{electrodes})$

computer vision: image (/video) compression/completion/$\cdots$
$(\text{pixel} \times \text{illumination} \times \text{expression} \times \cdots)$

web mining: analyze users behaviors
$(\text{users} \times \text{queries} \times \text{webpages})$



tensor $\mathcal{X}$

- Naive kernel: $K(\mathcal{X}, \mathcal{Y}) = \exp(-\frac{1}{2}\sigma^2 \|\text{vec}(\mathcal{X}) - \text{vec}(\mathcal{Y})\|_2^2)$
- Tensorial kernel exploiting structure: learning from few examples

[Signoretto et al., Neural Networks, 2011 & IEEE-TSP, 2012]

Function estimation in RKHS

- Find function f such that [Wahba, 1990; Evgeniou et al., 2000]

$$\min_{f \in \mathcal{H}_K} \frac{1}{N} \sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda \|f\|_K^2$$

  with $L(\cdot, \cdot)$ the loss function. $\|f\|_K$ is norm in RKHS $\mathcal{H}_K$ defined by K.

- Representer theorem: for convex loss function, solution of the form

$$f(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i)$$

  Reproducing property $f(x) = \langle f, K_x \rangle_K$ with $K_x(\cdot) = K(x, \cdot)$

- Sparse representation by ϵ-insensitive loss [Vapnik, 1998]

Black-box weather forecasting: multitask learning (1)



Weather data NOAA
mid 1998 - late 2010
350 stations located in US

Features:
Tmax, Tmin, precipitation,
wind speed, wind direction ,...

Preprocessing: kernel pca
(100 dominant components)

Black-box forecasting multiple weather stations simultaneously

[Signoretto, Frandi, Karevan, Suykens, IEEE-SCCI, 2014]

Black-box weather forecasting: multitask learning (2)

- Time-varying model: $y_t = f(x_t, t) + e_t$ with $x_t = [y_{t-1}, ..., y_{t-p}]^T$.

- Multiple regression functions $f^{(i)}$ for $i = 1, ..., I$
  Task-specific datasets $\mathcal{D}^{(i)} = \{(z_n, y_n)\}_{n=1}^{N_i}$ with $z = (x, t)$

- Auxiliary operator $F : \mathbb{R}^I \to \mathcal{H}_K$ to jointly estimate functions $f^{(i)}$ such that

$$f^{(i)} = \langle K_{(x,t)}, F\epsilon^{(i)} \rangle$$

  with $\epsilon^{(i)}$ the canonical basis, and $K_{(x,t)} : (x', t') \mapsto K((x, t), (x', t'))$.

- Learning problem:

$$\min_{F \in \mathcal{F}} \sum_{i \in [I]} \sum_{(z,y) \in \mathcal{D}^{(i)}} (y - \langle K_z, F\epsilon^{(i)} \rangle)^2 \quad \text{s.t.} \quad \|F\|_* \leq \tau$$

  with $\mathcal{F}$ the set of finite-rank operators from $\mathbb{R}^I$ to $\mathcal{H}_K$ and nuclear norm $\|F\|_* = \sum_{i \in [I]} \sigma_i(F)$.

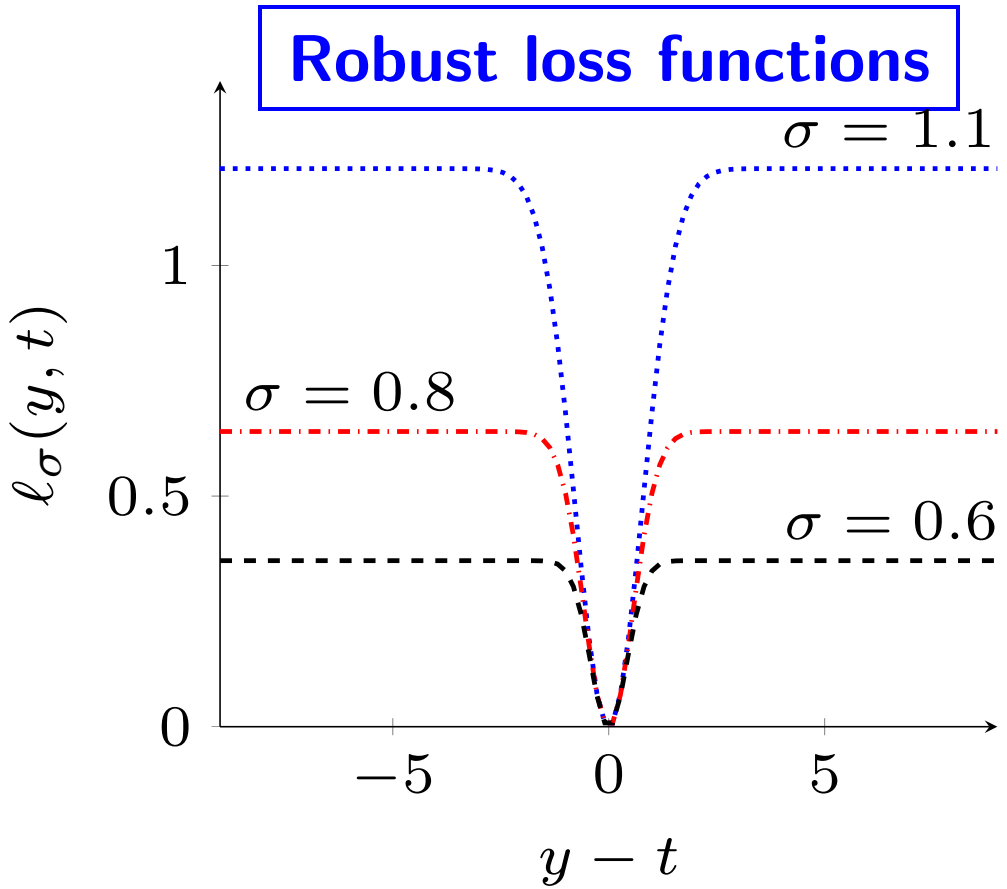
Kernels

Wide range of positive definite kernel functions possible:

- linear $K(x, z) = x^T z$
- polynomial $K(x, z) = (\eta + x^T z)^d$
- radial basis function $K(x, z) = \exp(-\|x - z\|_2^2/\sigma^2)$
- splines
- wavelets
- string kernel
- kernels from graphical models
- Fisher kernels
- graph kernels
- data fusion kernels
- tensorial kernels
- other

[Schölkopf & Smola, 2002; Shawe-Taylor & Cristianini, 2004; Jebara et al., 2004; other]

Robust loss functions



Correntropy loss function $\ell_\sigma(y,t) = \sigma^2(1 - e^{-(y-t)^2/\sigma^2})$

[Feng, Huang, Shi, Yang, Suykens, JMLR 2015; Principe, 2010]

Robust statistics [Huber, 1981; Hampel et al., 1986; Rousseeuw & Leroy, 1987]

(Re)-weighted LS for kernel models [Suykens et al., 2002; De Brabanter et al., 2009]

Learning with primal and dual model representations

Learning models from data: alternative views

- Consider model $\hat{y} = f(x; w)$, given input/output data $\{(x_i, y_i)\}_{i=1}^N$:

$$\min_w w^T w + \gamma \sum_{i=1}^N (y_i - f(x_i; w))^2$$

Learning models from data: alternative views

- Consider model $\hat{y} = f(x; w)$, given input/output data $\{(x_i, y_i)\}_{i=1}^N$:

$$\min_w \; w^T w + \gamma \sum_{i=1}^{N} (y_i - f(x_i; w))^2$$

- Rewrite the problem as

$$\begin{array}{ll} \min\limits_{w,e} & w^T w + \gamma \sum_{i=1}^N e_i^2 \\ \text{subject to} & e_i = y_i - f(x_i; w), i = 1, ..., N \end{array}$$

- Express the solution and the model in terms of Lagrange multipliers α_i

- For a model $f(x; w) = \sum_{j=1}^h w_j \varphi_j(x) = w^T \varphi(x)$ one obtains then $\hat{f}(x) = \sum_{i=1}^N \alpha_i K(x, x_i)$ with $K(x, x_i) = \varphi(x)^T \varphi(x_i)$.

Least Squares Support Vector Machines: "core models"

- Regression

$$\min_{w,b,e} w^T w + \gamma \sum_i e_i^2 \quad \text{s.t.} \quad y_i = w^T \varphi(x_i) + b + e_i, \quad \forall i$$

- Classification

$$\min_{w,b,e} w^T w + \gamma \sum_i e_i^2 \quad \text{s.t.} \quad y_i (w^T \varphi(x_i) + b) = 1 - e_i, \quad \forall i$$

- Kernel pca $(V = I)$, Kernel spectral clustering $(V = D^{-1})$

$$\min_{w,b,e} -w^T w + \gamma \sum_i v_i e_i^2 \quad \text{s.t.} \quad e_i = w^T \varphi(x_i) + b, \quad \forall i$$

- Kernel canonical correlation analysis/partial least squares

$$\min_{w,v,b,d,e,r} w^T w + v^T v + \nu \sum_i (e_i - r_i)^2 \text{ s.t. } \begin{cases} e_i &= w^T \varphi^{(1)}(x_i) + b \\ r_i &= v^T \varphi^{(2)}(y_i) + d \end{cases}$$

[Suykens & Vandewalle, 1999; Suykens et al., 2002; Alzate & Suykens, 2010]

Probability and quantum mechanics

- **Kernel pmf estimation**
  - *Primal:*

$$\min_{w,p_i} \tfrac{1}{2}\langle w,w\rangle \text{ subject to } \quad p_i = \langle w, \varphi(x_i)\rangle,\ i=1,...,N \text{ and } \sum_{i=1}^{N} p_i = 1$$

  - *Dual:* $p_i = \dfrac{\sum_{j=1}^{N} K(x_j,x_i)}{\sum_{i=1}^{N}\sum_{j=1}^{N} K(x_j,x_i)}$

- **Quantum measurement**: state vector $|\psi\rangle$, measurement operators M_i
  - *Primal:*

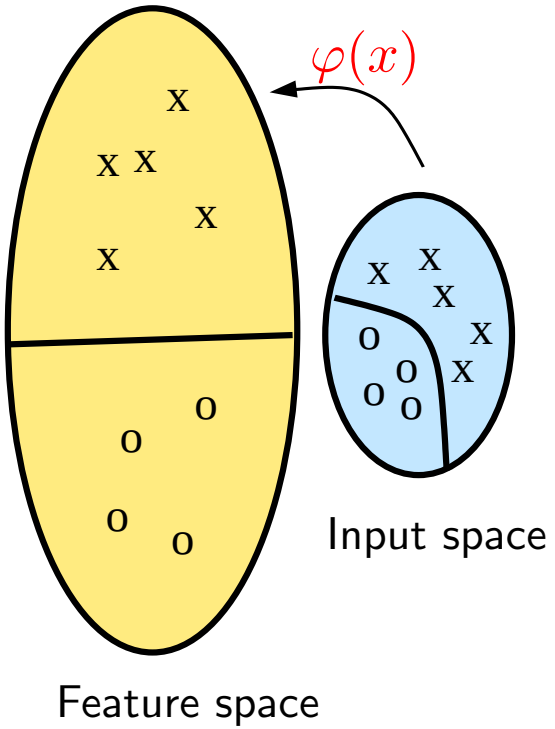
$$\min_{|w\rangle,p_i} \tfrac{1}{2}\langle w|w\rangle \text{ subject to } \quad p_i = \mathrm{Re}(\langle w|M_i\psi\rangle),\ i=1,...,N \text{ and } \sum_{i=1}^{N} p_i = 1$$

  - *Dual:* $p_i = \langle\psi|M_i|\psi\rangle$ (Born rule, orthogonal projective measurement)

[Suykens, Physical Review A, 2013]

SVMs: living in two worlds ...

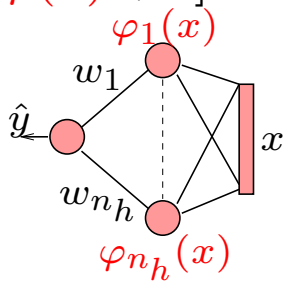




Primal space

Parametric

$$\hat{y} = \text{sign}[w^T \varphi(x) + b]$$

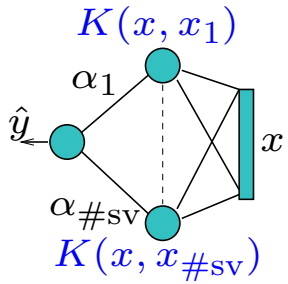


$$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j) \quad \textbf{(Mercer)}$$

Dual space

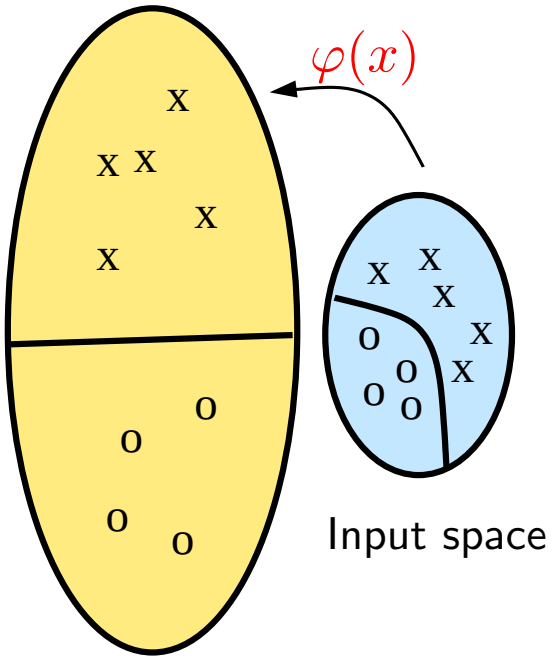
Nonparametric

$$\hat{y} = \text{sign}[\sum_{i=1}^{\#\text{sv}} \alpha_i y_i K(x, x_i) + b]$$



SVMs: living in two worlds ...

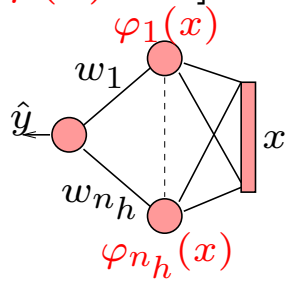




Primal space

Parametric

$$\hat{y} = \text{sign}[w^T \varphi(x) + b]$$

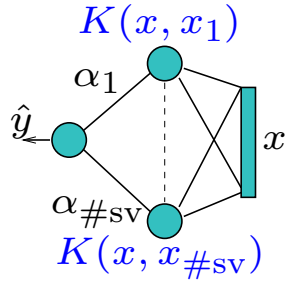


$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$ **("Kernel trick")**

Dual space

Nonparametric

$$\hat{y} = \text{sign}[\sum_{i=1}^{\#\text{sv}} \alpha_i y_i K(x, x_i) + b]$$



Parametric

Non−parametric

Linear model: solving in primal or dual?

inputs $x \in \mathbb{R}^d$, output $y \in \mathbb{R}$
training set $\{(x_i, y_i)\}_{i=1}^N$

$$(P): \quad \hat{y} = w^T x + b, \qquad w \in \mathbb{R}^d$$

Model

Linear model: solving in primal or dual?

inputs $x \in \mathbb{R}^d$, output $y \in \mathbb{R}$
training set $\{(x_i, y_i)\}_{i=1}^N$

$$\text{Model} \nearrow \quad (P): \quad \hat{y} = w^T x + b, \qquad w \in \mathbb{R}^d$$

$$\text{Model} \searrow \quad (D): \quad \hat{y} = \sum_i \alpha_i \, x_i^T x + b, \qquad \alpha \in \mathbb{R}^N$$

Linear model: solving in primal or dual?

few inputs, many data points: $d \ll N$

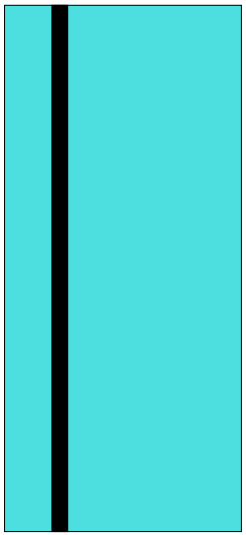


primal: $w \in \mathbb{R}^d$

dual: $\alpha \in \mathbb{R}^N$ (large kernel matrix: $N \times N$)

Linear model: solving in primal or dual?

many inputs, few data points: $d \gg N$



primal: $w \in \mathbb{R}^d$

dual: $\alpha \in \mathbb{R}^N$ (small kernel matrix: $N \times N$)

Feature map and kernel

From linear to nonlinear model:

$$\text{Model} \begin{cases} \nearrow & (P): \quad \hat{y} = w^T \varphi(x) + b \\ \searrow & (D): \quad \hat{y} = \sum_i \alpha_i K(x_i, x) + b \end{cases}$$

Mercer theorem:

$$K(x, z) = \varphi(x)^T \varphi(z)$$

Feature map $\varphi(x) = [\varphi_1(x); \varphi_2(x); ...; \varphi_h(x)]$
Kernel function $K(x, z)$ (e.g. linear, polynomial, RBF, ...)

- Use of feature map and positive definite kernel [Cortes & Vapnik, 1995]
- Extension to infinite dimensional case:
  - LS-SVM formulation [Signoretto, De Lathauwer, Suykens, 2011]
  - HHK Transform, coherent states, wavelets [Fanuel & Suykens, 2015]

Sparsity by fixed-size kernel method

Fixed-size method

1. **selection of a subset** from the data (random, quadratic Renyi entropy, incomplete Cholesky factorization, other)

2. kernel matrix on the subset

3. eigenvalue decomposition of kernel matrix

4. **approximation of the feature map** based on the eigenvectors (Nyström approximation)

5. estimation of the model in the primal using the approximate feature map (applicable to large data sets)

[Suykens et al., 2002] *(ls-svm book)*

Fixed-size method: performance in classification

	pid	spa	mgt	adu	ftc
N	768	4601	19020	45222	581012
N_{cv}	512	3068	13000	33000	531012
N_{test}	256	1533	6020	12222	50000
d	8	57	11	14	54
FS-LSSVM (# SV)	150	200	1000	500	500
C-SVM (# SV)	290	800	7000	11085	185000
ν-SVM (# SV)	331	1525	7252	12205	165205
RBF FS-LSSVM	76.7(3.43)	92.5(0.67)	86.6(0.51)	85.21(0.21)	81.8(0.52)
Lin FS-LSSVM	77.6(0.78)	90.9(0.75)	77.8(0.23)	83.9(0.17)	75.61(0.35)
RBF C-SVM	75.1(3.31)	92.6(0.76)	85.6(1.46)	84.81(0.20)	81.5(no cv)
Lin C-SVM	76.1(1.76)	91.9(0.82)	77.3(0.53)	83.5(0.28)	75.24(no cv)
RBF ν-SVM	75.8(3.34)	88.7(0.73)	84.2(1.42)	83.9(0.23)	81.6(no cv)
Maj. Rule	64.8(1.46)	60.6(0.58)	65.8(0.28)	83.4(0.1)	51.23(0.20)

- Fixed-size (FS) LSSVM: good performance and sparsity wrt C-SVM and ν-SVM
- Challenging to achieve high performance by very sparse models

[De Brabanter et al., CSDA 2010]

Two stages of sparsity

primal	
dual	subset selection Nyström approximation

Two stages of sparsity

	stage 1
primal	FS model estimation
	↑
dual	subset selection Nyström approximation

Two stages of sparsity

	stage 1	stage 2
primal	FS model estimation $\longrightarrow$	reweighted ℓ_1
	$\uparrow$	
dual	subset selection Nyström approximation	

Synergy between parametric & kernel-based models

[Mall & Suykens, IEEE-TNNLS 2015], reweighted ℓ_1 [Candes et al., 2008]

Two stages of sparsity

	stage 1	stage 2
primal	FS model estimation $\longrightarrow$	reweighted ℓ_1
dual	$\uparrow$ subset selection Nyström approximation	

Synergy between parametric & kernel-based models

[Mall & Suykens, IEEE-TNNLS 2015], reweighted ℓ_1 [Candes et al., 2008]

Other possible approaches with improved sparsity: SCAD [Fan & Li, 2001]; coefficient-based ℓ_q $(0 < q \leq 1)$ [Shi et al., 2013]; two-level ℓ_1 [Huang et al., 2014]

Kernel PCA and kernel spectral clustering

Kernel PCA

- Primal problem: [Suykens et al., 2002]

$$\min_{w,b,e} \ \frac{1}{2}w^T w - \frac{1}{2}\gamma \sum_{i=1}^{N} e_i^2 \ \text{ s.t. } \ e_i = w^T \varphi(x_i) + b, \ i = 1, ..., N.$$

- Dual problem corresponds to kernel PCA [Scholkopf et al., 1998]

$$\Omega_c \alpha = \lambda \alpha \ \text{ with } \ \lambda = 1/\gamma$$

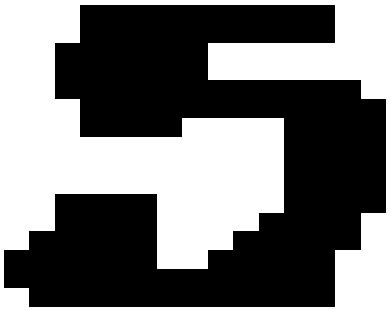
  with $\Omega_{c,ij} = (\varphi(x_i) - \hat{\mu}_\varphi)^T (\varphi(x_j) - \hat{\mu}_\varphi)$ the *centered kernel matrix*.

- Interpretation:
  1. pool of candidate components (objective function equals zero)
  2. select relevant components

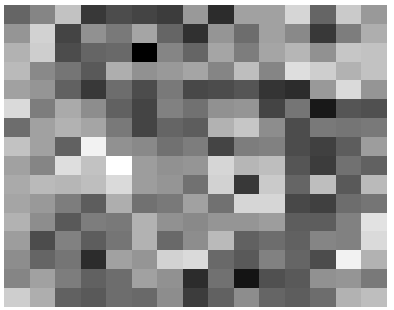
- Robust and sparse versions [Alzate & Suykens, 2008]: by taking other loss functions

Robustness: Kernel Component Analysis

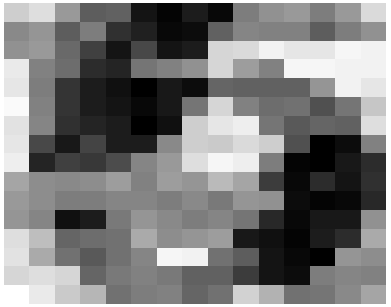
original image



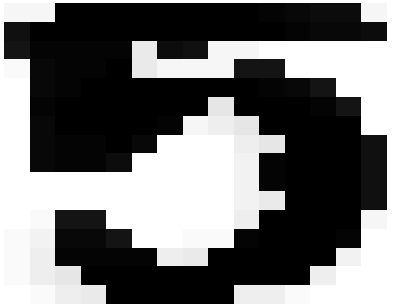
corrupted image



KPCA reconstruction



KCA reconstruction



Weighted LS-SVM [Alzate & Suykens, IEEE-TNN 2008]: robustness and sparsity

Kernel Spectral Clustering (KSC): case of two clusters

- **Primal problem:** training on given data $\{x_i\}_{i=1}^N$

$$\begin{array}{ll} \min\limits_{w,b,e} & \frac{1}{2}w^Tw - \gamma\frac{1}{2}e^TVe \\ \text{subject to} & e_i = w^T\varphi(x_i) + b, \quad i = 1, ..., N \end{array}$$

  with weighting matrix V and $\varphi(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^h$ the feature map.

- **Dual:**

$$VM_V\Omega\alpha = \lambda\alpha$$

  with $\lambda = 1/\gamma$, $M_V = I_N - \frac{1}{1_N^TV1_N}1_N1_N^TV$ weighted centering matrix, $\Omega = [\Omega_{ij}]$ kernel matrix with $\Omega_{ij} = \varphi(x_i)^T\varphi(x_j) = K(x_i, x_j)$

- Taking $V = D^{-1}$ with degree matrix $D = \text{diag}\{d_i\}$, $d_i = \sum_{j=1}^N \Omega_{ij}$ relates to random walks algorithm [Chung, 1997; Shi & Malik, 2000; Ng 2002]

[Alzate & Suykens, IEEE-PAMI, 2010]

Lagrangian and conditions for optimality

- Lagrangian:

$$\mathcal{L}(w,b,e;\alpha) = \frac{1}{2}w^T w - \gamma \frac{1}{2}\sum_{i=1}^{N} v_i e_i^2 + \sum_{i=1}^{N} \alpha_i (e_i - w^T\varphi(x_i) - b)$$

- Conditions for optimality:

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial w} = 0 \quad \Rightarrow \quad w = \sum_i \alpha_i \varphi(x_i) \\ \dfrac{\partial \mathcal{L}}{\partial b} = 0 \quad \Rightarrow \quad \sum_i \alpha_i = 0 \\ \dfrac{\partial \mathcal{L}}{\partial e_i} = 0 \quad \Rightarrow \quad \alpha_i = \gamma v_i e_i, \;\; i = 1,...,N \\ \dfrac{\partial \mathcal{L}}{\partial \alpha_i} = 0 \quad \Rightarrow \quad e_i = w^T\varphi(x_i) + b, \;\; i = 1,...,N \end{cases}$$

- Eliminate w, b, e, write solution in Lagrange multipliers α_i.

Primal and dual model representations

k clusters
$k-1$ sets of constraints (index $l = 1, ..., k-1$)

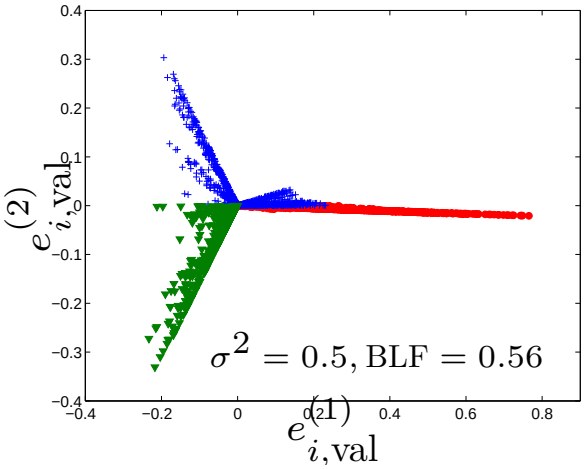
$$\mathcal{M} \nearrow \quad (P): \quad \text{sign}[\hat{e}_*^{(l)}] = \text{sign}[{w^{(l)}}^T \varphi(x_*) + b_l]$$

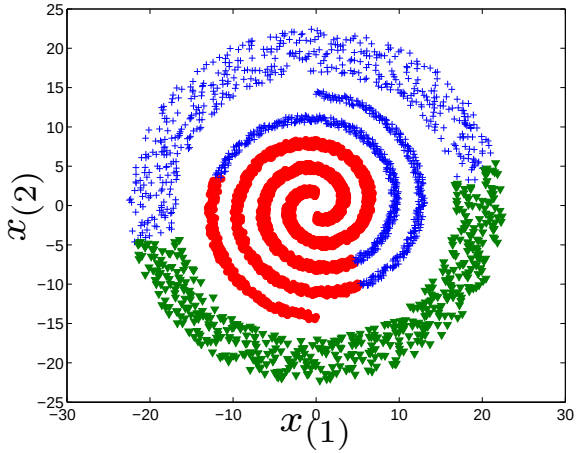
$$\mathcal{M} \searrow \quad (D): \quad \text{sign}[\hat{e}_*^{(l)}] = \text{sign}[\sum_j \alpha_j^{(l)} K(x_*, x_j) + b_l]$$

Advantages of kernel-based setting

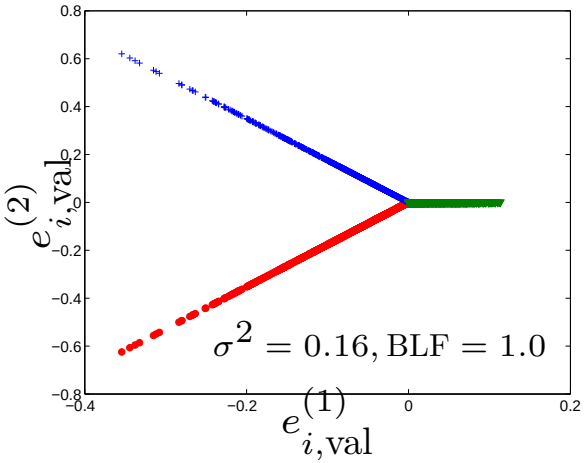
- **model-based** approach
- **out-of-sample extensions**, applying model to new data
- consider **training, validation and test data**
  (training problem corresponds to eigenvalue decomposition problem)
- model selection procedures
- **sparse representations and large scale methods**

Model selection: toy example

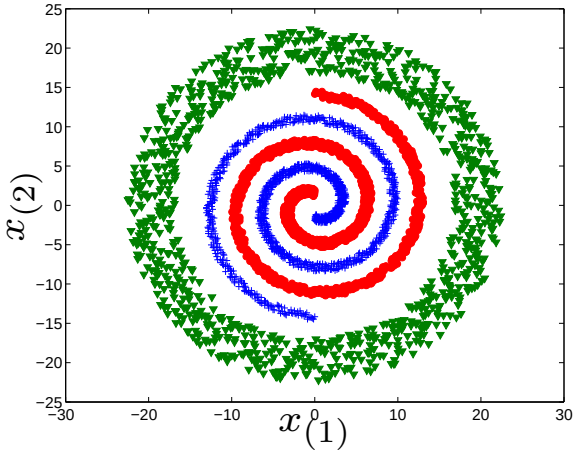




BAD



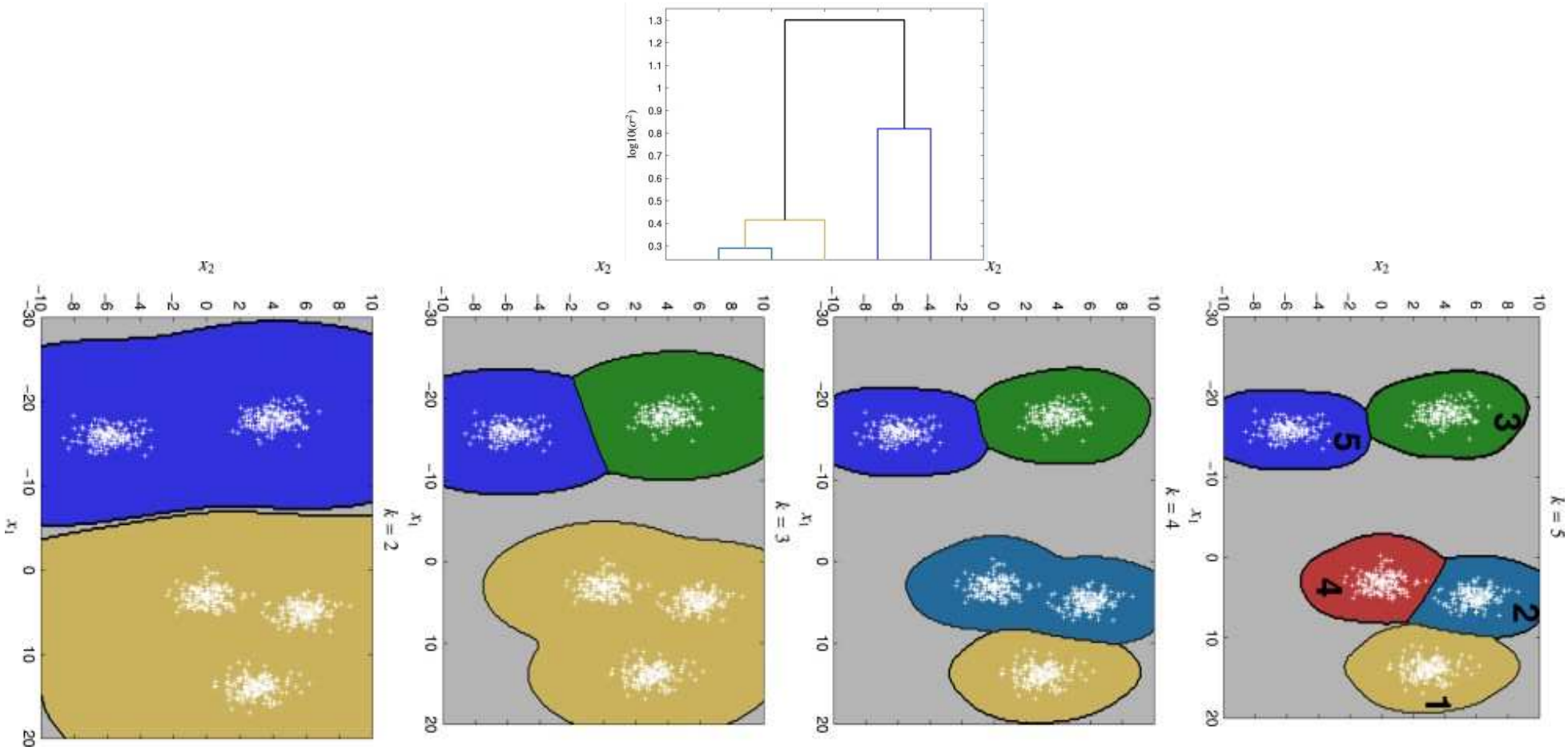
validation set



train + validation + test data

GOOD

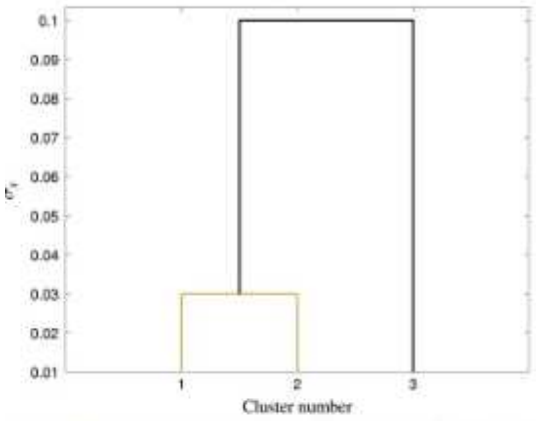
Hierarchical KSC



[Alzate & Suykens, 2012]

Hierarchical KSC









[Alzate & Suykens, 2012]

KSC for big data networks (1)

- **YouTube Network:** YouTube social network where users form friendship with each other and can create groups where others can join.
- **RoadCA Network:** a road network of California. Intersections and endpoints are represented by nodes. The roads connecting these intersections are represented as edges.
- **Livejournal Network**: free online social network where the users are bloggers declaring friendship.

Dataset	Nodes	Edges
YouTube	1,134,890	2,987,624
roadCA	1,965,206	5,533,214
Livejournal	3,997,962	34,681,189

[Mall, Langone, Suykens, Entropy, special issue Big data, 2013]

KSC for big data networks (2)

BAF-KSC:

- Select representative training subgraph using FURS
  (FURS = Fast and Unique Representative Subset selection [Mall et al., 2013]: selects nodes with high degree centrality belonging to different dense regions, using a deactivation and activation procedure)

- Perform model selection using BAF
  (BAF = Balanced Angular Fit: makes use of a cosine similarity measure related to the e-projection values on validation nodes)

- Train the KSC model by solving a small eigenvalue problem of size $\min(0.15N, 5000)^2$

- Apply out-of-sample extension to find cluster memberships of the remaining nodes

[Mall, Langone, Suykens, Entropy, special issue Big data, 2013]

KSC for big data networks (3)

BAF-KSC [Mall, Langone, Suykens, 2013] **Louvain** [Blondel et al., 2008]

Infomap [Lancichinetti, Fortunato, 2009] **CNM** [Clauset, Newman, Moore, 2004]

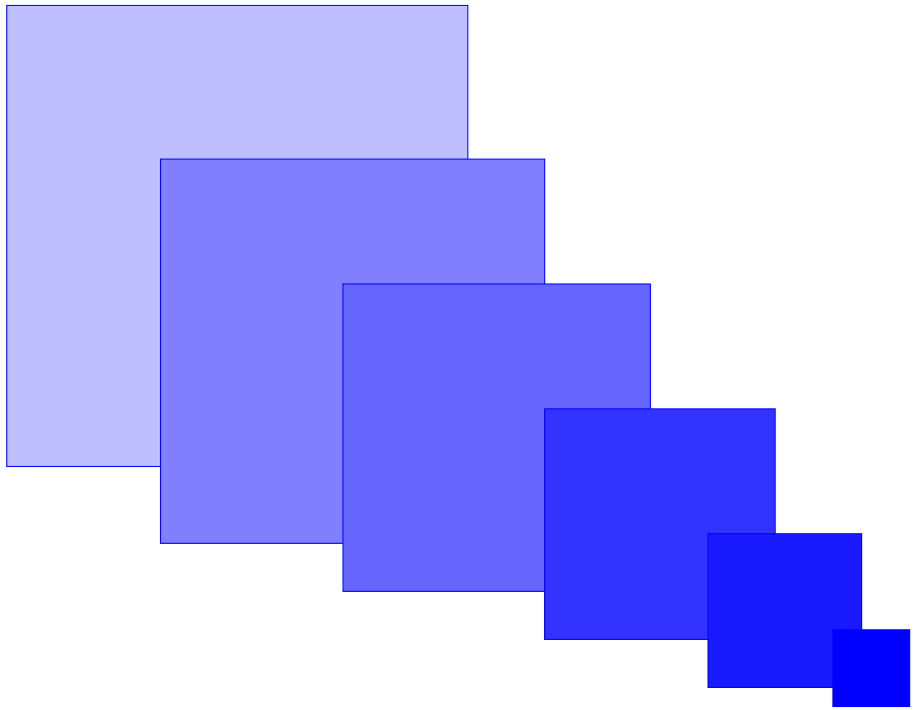
Dataset	BAF-KSC			Louvain			Infomap			CNM		
	Cl	Q	Con	Cl	Q	Con	Cl	Q	Con	Cl	Q	Con
Openflight	5	0.533	**0.002**	109	**0.61**	0.02	18	0.58	0.005	84	0.60	0.016
PGPnet	8	0.58	**0.002**	105	**0.88**	0.045	84	0.87	0.03	193	0.85	0.041
Metabolic	10	0.22	0.028	10	**0.43**	0.03	41	0.41	0.05	11	0.42	0.021
HepTh	6	0.45	**0.0004**	172	**0.65**	0.004	171	0.3	0.004	6	0.423	0.0004
HepPh	5	0.56	0.0004	82	**0.72**	0.007	69	0.62	0.06	6	0.48	0.0007
Enron	10	0.4	0.002	1272	**0.62**	0.05	1099	0.37	0.27	6	0.25	0.0045
Epinion	10	**0.22**	0.0003	33	0.006	0.0003	17	0.18	0.0002	10	0.14	**0.0**
Condmat	6	0.28	**0.0002**	1030	**0.79**	0.03	1086	0.79	0.025	8	0.38	0.0003

Flight network (Openflights), network based on trust (PGPnet), biological network (Metabolic), citation networks (HepTh, HepPh), communication network (Enron), review based network (Epinion), collaboration network (Condmat) [snap.stanford.edu]

Cl = Clusters, Q = modularity, Con = Conductance

BAF-KSC usually finds a smaller number of clusters and achieves lower conductance

Multilevel Hierarchical KSC for complex networks (1)



Generating a series of affinity matrices over different levels:
communities at level h become nodes for next level $h+1$

Multilevel Hierarchical KSC for complex networks (2)

- Start at ground level 0 and compute cosine similarities on validation nodes $S_{val}^{(0)}(i,j) = 1 - e_i^T e_j / (\|e_i\| \|e_j\|)$
- Select projections e_j satisfying $S_{val}^{(0)}(i,j) < t^{(0)}$ with $t^{(0)}$ a threshold value. Keep these nodes as the first cluster at level 0 and remove the nodes from matrix $S_{val}^{(0)}$ to obtain a reduced matrix.
- **Iterative procedure over different levels h:**
  Communities at level h become nodes for the next level $h+1$, by creating a set of matrices at different levels of hierarchy:

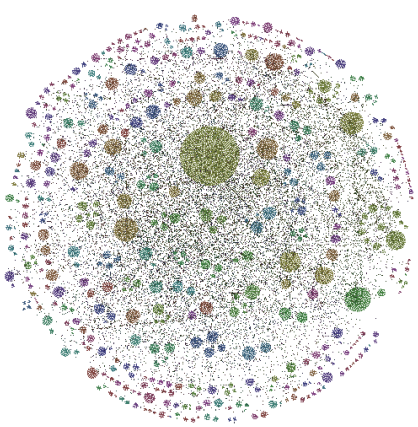
$$S_{val}^{(h)}(i,j) = \frac{1}{|C_i^{(h-1)}| \, |C_j^{(h-1)}|} \sum_{m \in C_i^{(h-1)}} \sum_{l \in C_j^{(h-1)}} S_{val}^{(h-1)}(m,l)$$

  and working with suitable threshold values $t^{(h)}$ for each level.

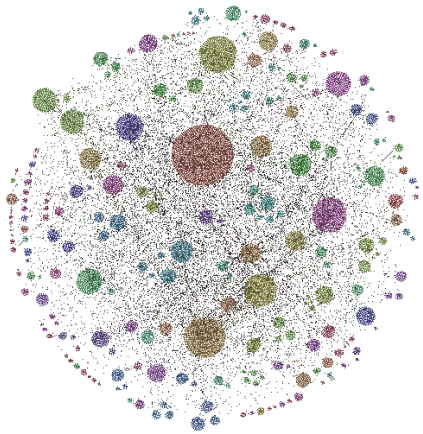
[Mall, Langone, Suykens, PLOS ONE, 2014]

Multilevel Hierarchical KSC for complex networks (3)

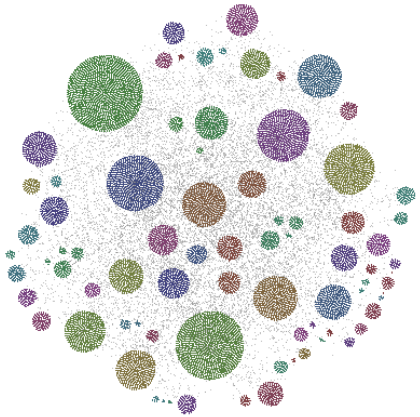
MH-KSC on PGP network:



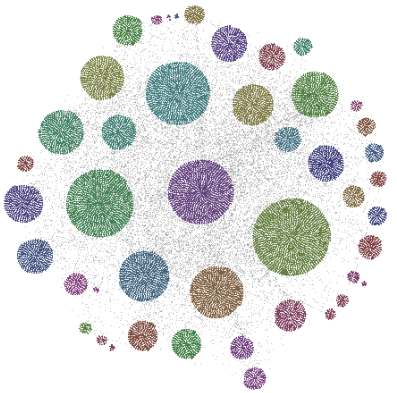
fine



intermediate



intermediate

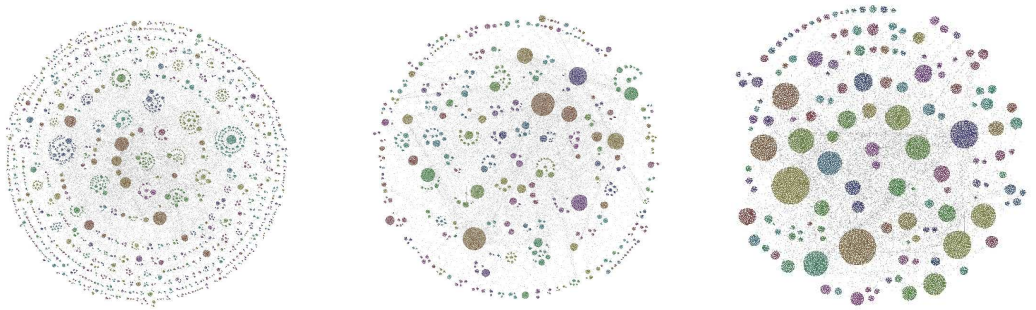


coarse

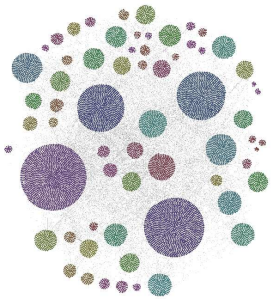
Multilevel Hierarchical KSC finds high quality clusters at coarse as well as fine and intermediate levels of hierarchy.

[Mall, Langone, Suykens, PLOS ONE, 2014]

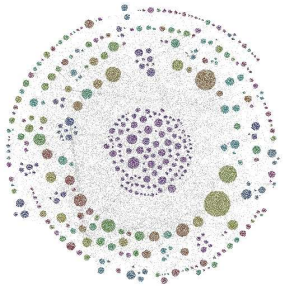
Multilevel Hierarchical KSC for complex networks (4)

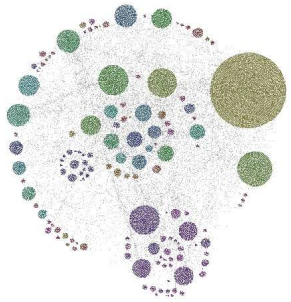


Louvain



Infomap



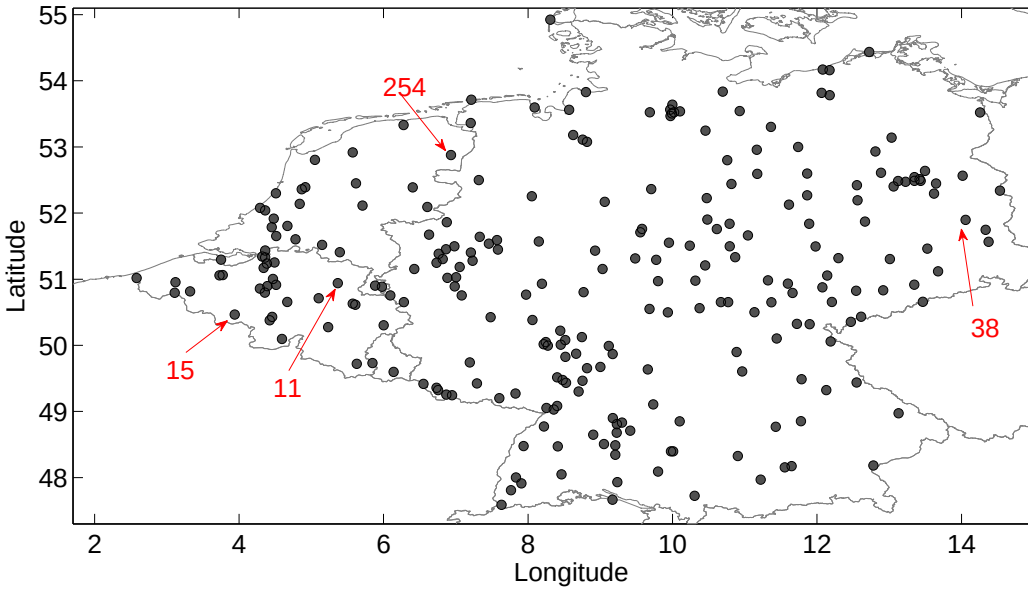


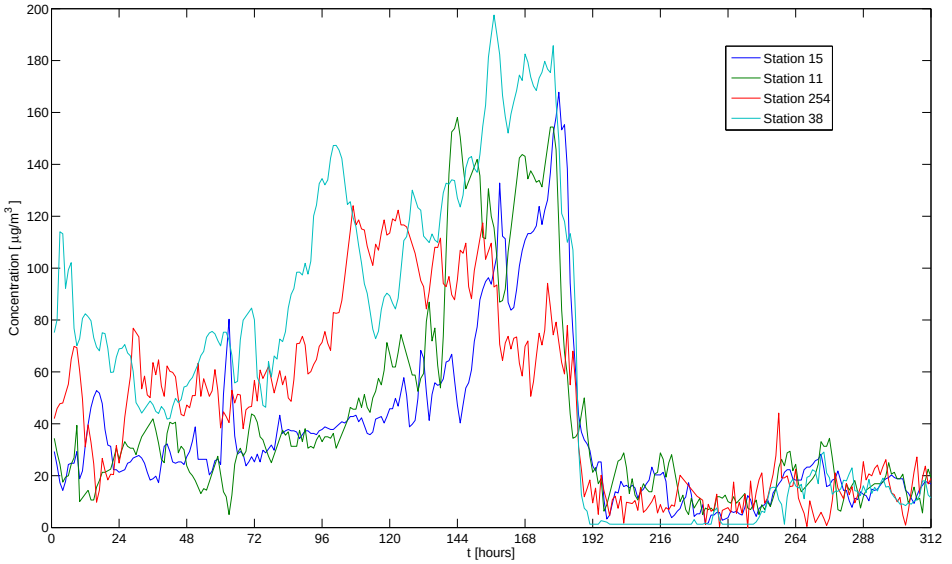
OSLOM

Louvain, Infomap, and OSLOM seem biased toward a particular scale in comparison with MH-KSC, based upon ARI, VI, Q metrics

Example: KSC of PM10 concentrations (1)

PM10 time-series: PM10 data (Particulate Matter) registered during a heavy pollution episode (Jan 20 2010 - Feb 1 2010) in Europe.

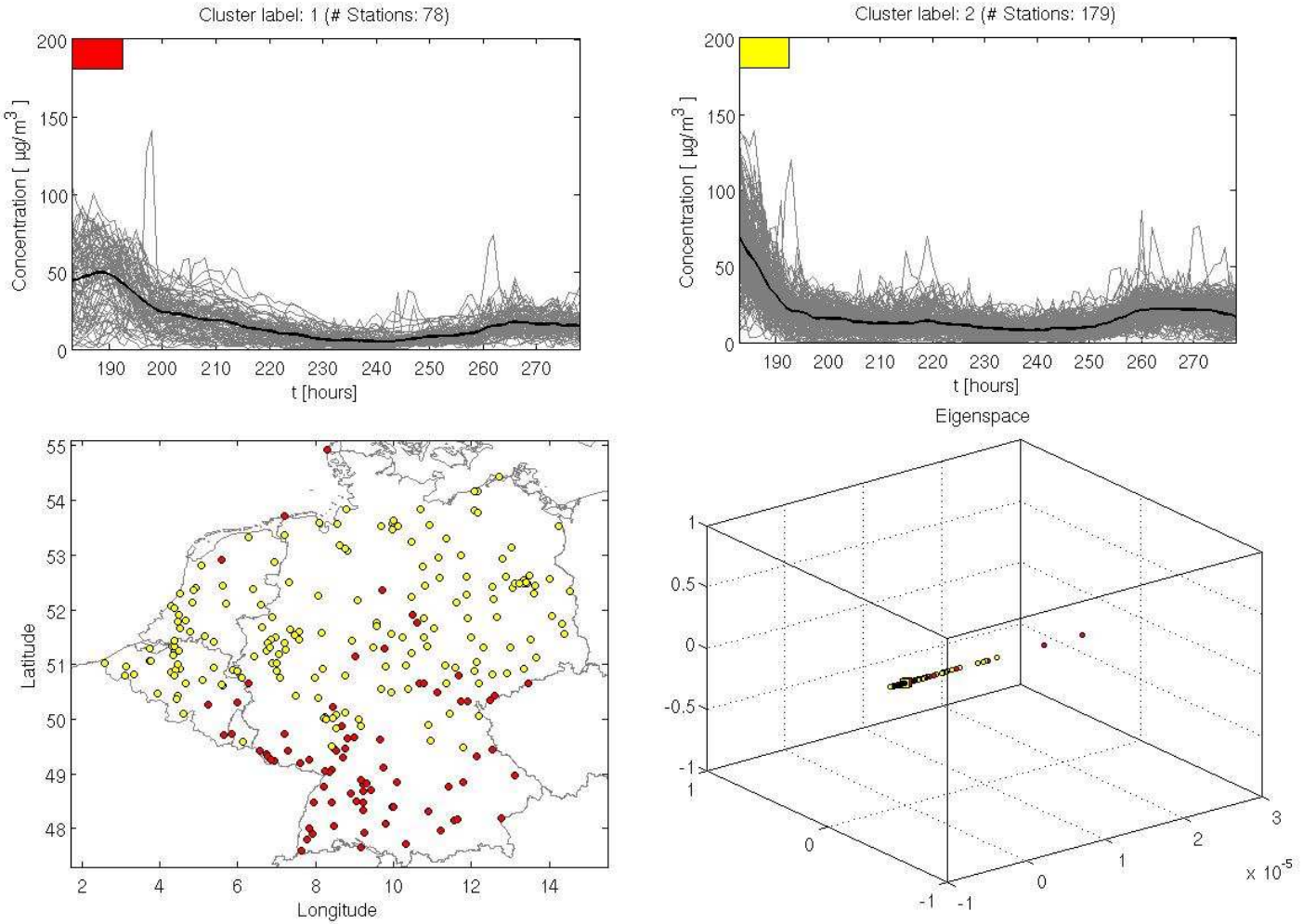




[Langone, Agudelo, De Moor, Suykens, Neurocomputing, 2014]

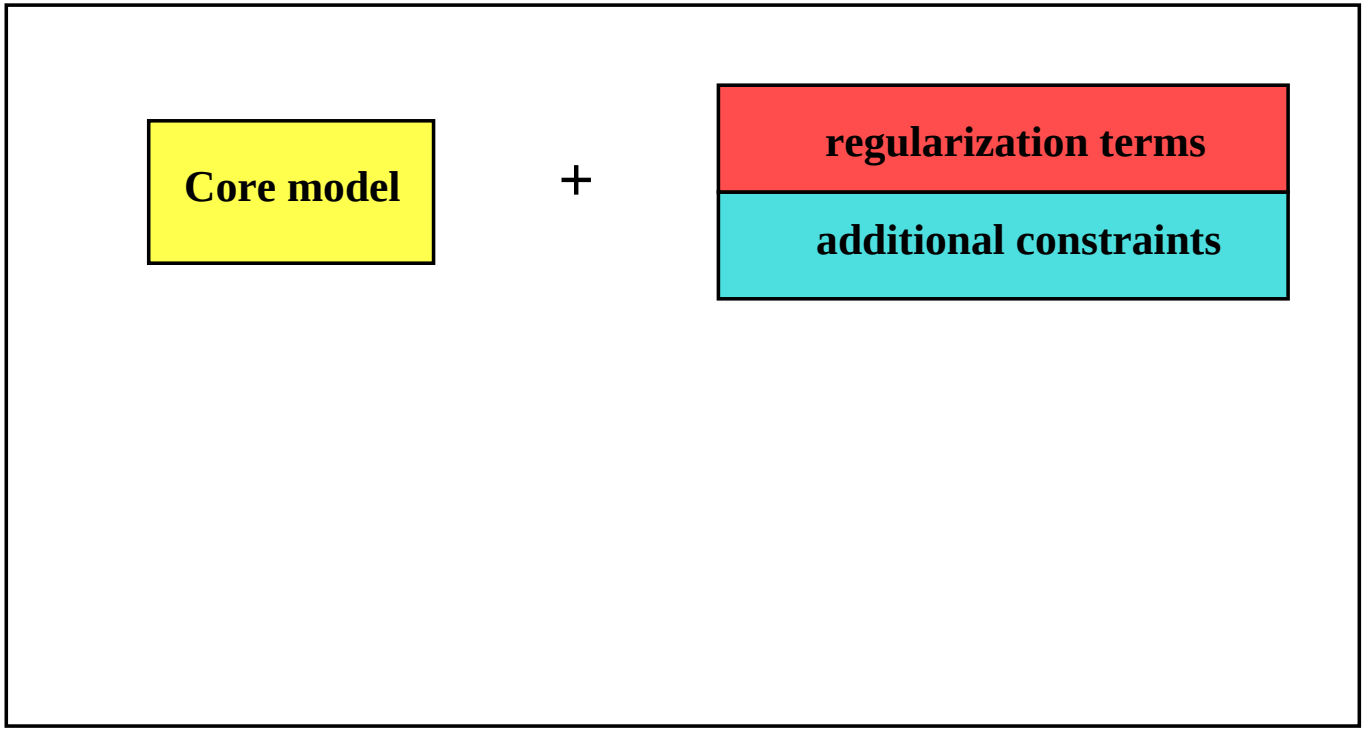
[Talk - R. Langone]

Example: KSC of PM10 concentrations (2)

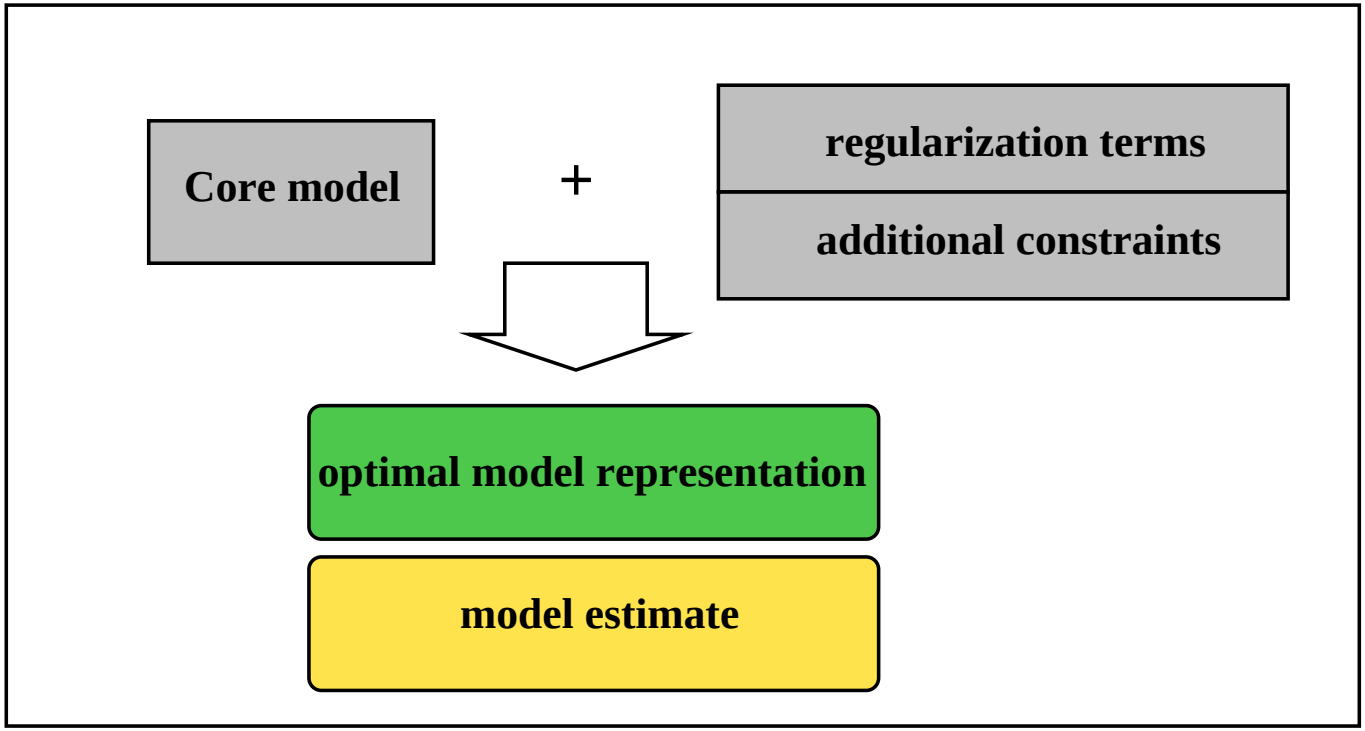


video - [Langone, Agudelo, De Moor, Suykens, Neurocomputing, 2014]

Core models + constraints

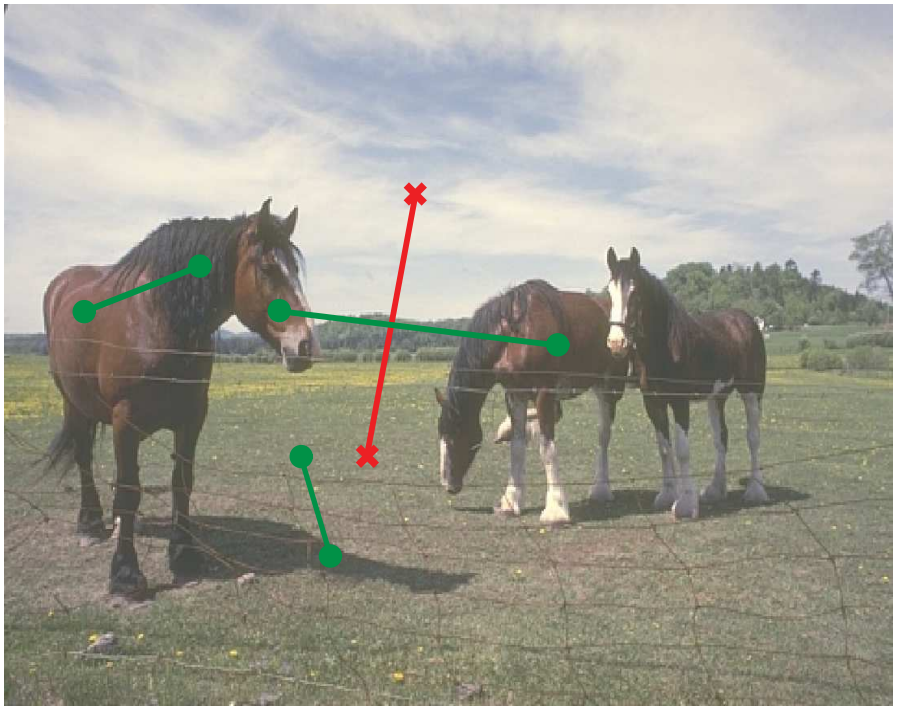


Core models + constraints



Adding prior knowledge: example

original image

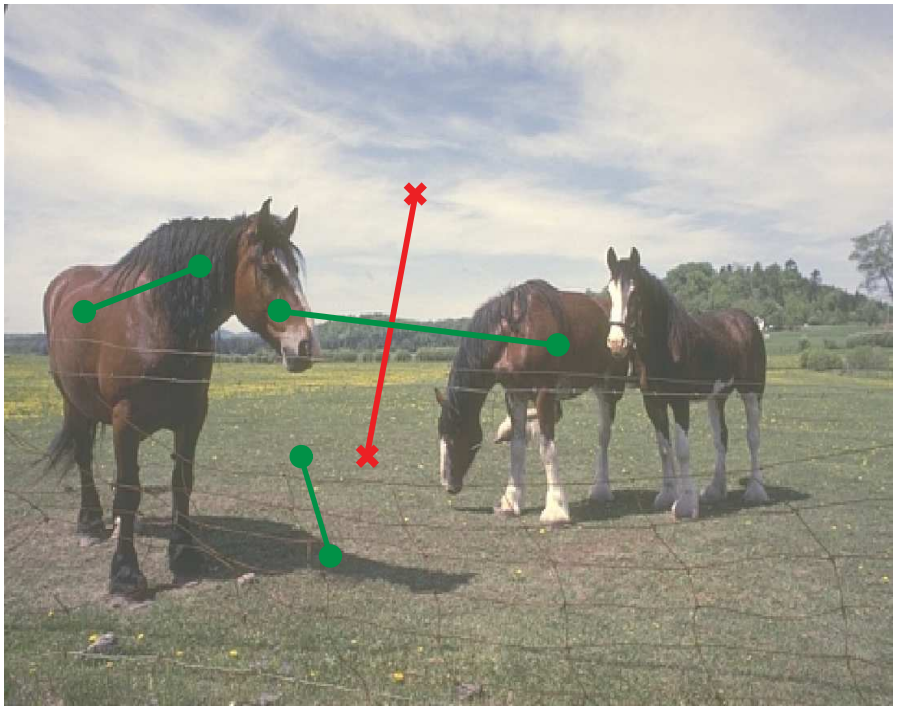


without constraints

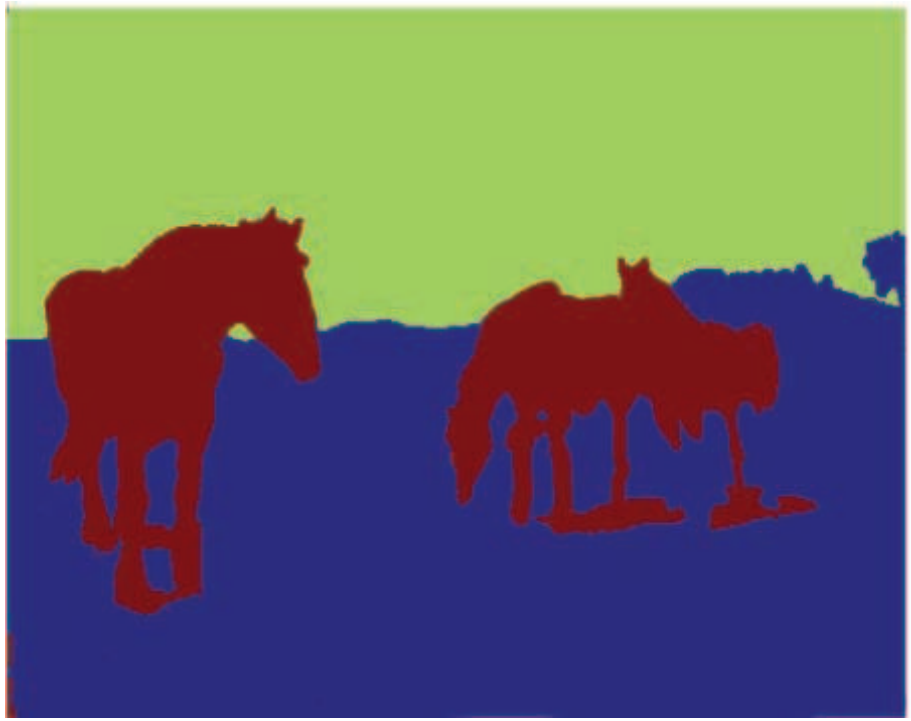


Adding prior knowledge: example

original image



with constraints



Semi-supervised learning using KSC (1)

- N unlabeled data, but additional labels on $M - N$ data
  $\mathcal{X} = \{x_1, ..., x_N, x_{N+1}, ..., x_M\}$

- Kernel spectral clustering as core model (binary case [Alzate & Suykens, WCCI 2012], multi-way/multi-class [Mehrkanoon et al., TNNLS 2015])

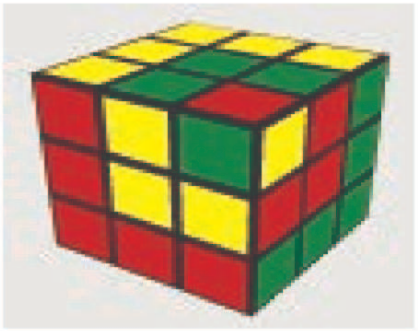
$$\begin{aligned} \min_{w,e,b} \quad & \frac{1}{2} w^T w - \gamma \frac{1}{2} e^T D^{-1} e + \rho \frac{1}{2} \sum_{m=N+1}^{M} (e_m - y_m)^2 \\ \text{subject to} \quad & e_i = w^T \varphi(x_i) + b, \; i = 1, ..., M \end{aligned}$$

  Dual solution is characterized by a linear system. Suitable for clustering as well as classification.

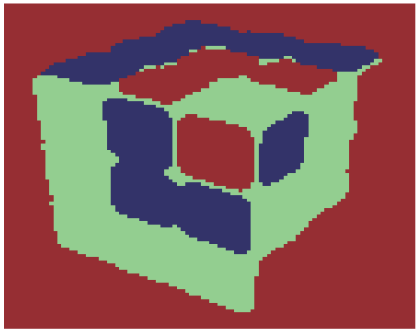
- Other approaches in semi-supervised learning and manifold learning, e.g. [Belkin et al., 2006]

Semi-supervised learning using KSC (2)

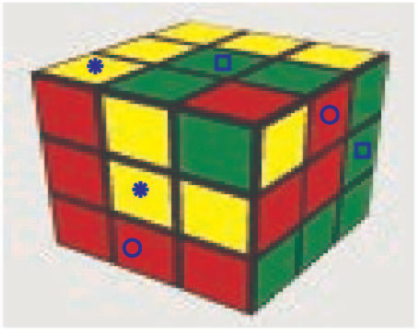
original image



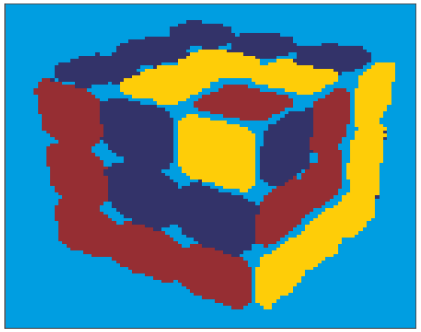
KSC



given a few labels



semi-supervised KSC



[Mehrkanoon, Alzate, Mall, Langone, Suykens, IEEE-TNNLS 2015], videos

SVD from LS-SVM

&

nonlinear extensions

Singular Value Decomposition (SVD)

- **Singular Value Decomposition (SVD)** of $A \in \mathbb{R}^{N \times M}$

$$A = U \Sigma V^T$$

with $U^T U = I_N$, $V^T V = I_M$, $\Sigma = \mathrm{diag}(\sigma_1, ..., \sigma_p) \in \mathbb{R}^{N \times M}$.

- **Early history overview** by [Stewart, 1993]:
  - early contributions by Beltrami (1873), Jordan (1874), Eckart & Young (1936), Lanczos (1958), and several others.
  - related work on integral equations: Schmidt (1907)

SVD - classical variational principle

- extrema of the bilinear form

$$f(u,v) = u^T A v \;\; \text{subject to} \;\; \|u\|^2 = \|v\|^2 = 1.$$

- solutions follow from the eigenvalue decomposition

$$\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} u \\ v \end{bmatrix}$$

where $\lambda \in \{\pm\sigma_1, \pm\sigma_2, ..., \pm\sigma_p, 0\}$ with multiplicity $M - N$ for the zero eigenvalue (assuming $M > N$ and non-zero σ_i values).

The Lanczos decomposition theorem

- **Theorem [Decomposition Theorem, Lanczos (1958)]**. An arbitrary non-zero matrix A can be written as

$$A = \tilde{U}\tilde{\Lambda}\tilde{V}^T$$

with $\tilde{U} \in \mathbb{R}^{N\times r}$ $(\tilde{U}^T\tilde{U} = I)$, $\tilde{\Lambda} \in \mathbb{R}^{r\times r}$ a positive diagonal matrix and $\tilde{V} \in \mathbb{R}^{M\times r}$ $(\tilde{V}^T\tilde{V} = I)$ where the matrices $\tilde{U}, \tilde{V}, \tilde{\Lambda}$ are defined by the shifted eigenvalue problem

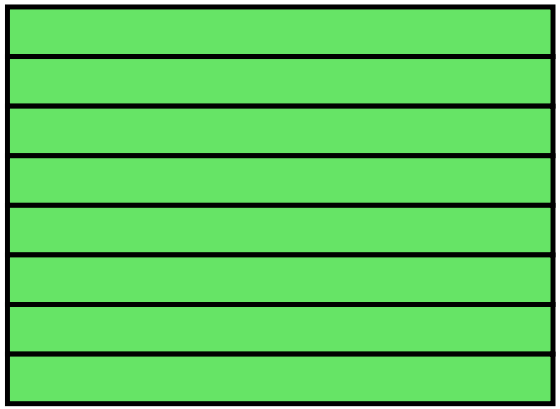
$$\begin{aligned} A\tilde{V} &= \tilde{U}\tilde{\Lambda} \\ A^T\tilde{U} &= \tilde{V}\tilde{\Lambda} \end{aligned}$$

with the additional condition that all diagonal elements of $\tilde{\Lambda}$ are non-zero positive numbers.

- Lanczos: *"It is remarkable that the principal axes associated with the zero eigenvalue do not participate at all in the formation of matrix A"*

SVD within the LS-SVM setting (1)

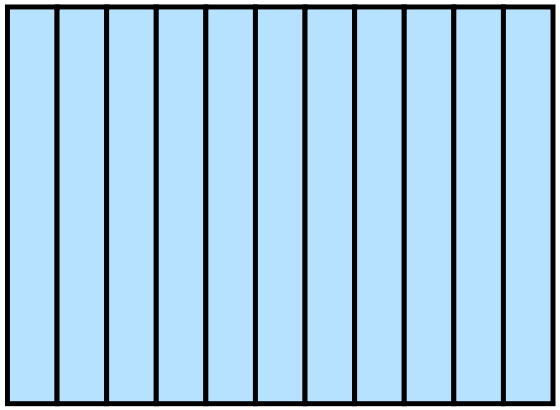
matrix A



$\{x_i\}$

SVD within the LS-SVM setting (1)

matrix A



$\{z_j\}$

SVD within the LS-SVM setting (2)

- **Obtain two sets of data points** (rows and columns):

$$x_i = A^T \epsilon_i, \quad z_j = A \varepsilon_j$$

for $i = 1, ..., N$, $j = 1, ..., M$ where $\epsilon_i, \varepsilon_j$ are standard basis vectors of dimension N and M.

- **Compatible feature maps:** $\varphi : \mathbb{R}^M \to \mathbb{R}^N$, $\psi : \mathbb{R}^N \to \mathbb{R}^N$ where

$$\begin{array}{rcl} \varphi(x_i) & = & C^T x_i = C^T A^T \epsilon_i \\ \psi(z_j) & = & z_j = A \varepsilon_j \end{array}$$

with $C \in \mathbb{R}^{M \times N}$ a **compatibility matrix**.

[Suykens, ACHA, 2015, in press]

SVD within the LS-SVM setting (3)

- **Primal problem** (new variational principle):

$$\min_{w,v,e,r} \; -w^T v + \tfrac{1}{2}\gamma \sum_{i=1}^{N} e_i^2 + \tfrac{1}{2}\gamma \sum_{j=1}^{M} r_j^2 \;\text{ subject to } \quad \begin{array}{l} e_i = w^T\varphi(x_i),\; i=1,...,N \\ r_j = v^T\psi(z_j),\; j=1,...,M \end{array}$$

  with $e_i, r_j \in \mathbb{R}$ and $w, v \in \mathbb{R}^N$.

- Lagrangian $\mathcal{L} = J - \sum_i \alpha_i \left(e_i - w^T\varphi(x_i)\right) - \sum_j \beta_j \left(r_j - v^T\psi(z_j)\right)$

- Optimality conditions $\frac{\partial \mathcal{L}}{\partial w} = 0, \frac{\partial \mathcal{L}}{\partial v} = 0, \frac{\partial \mathcal{L}}{\partial e_i} = 0, \frac{\partial \mathcal{L}}{\partial r_j} = 0, \frac{\partial \mathcal{L}}{\partial \alpha_i} = 0, \frac{\partial \mathcal{L}}{\partial \beta_j} = 0$

- Eliminate w, v, e, r; collect solutions corresponding to non-zero $\lambda = \frac{1}{\gamma}$:

$$\begin{array}{rcl} \left[\varphi(x_i)^T\psi(z_j)\right][\beta] & = & [\alpha]\tilde{\Lambda} \\ \left[\psi(z_j)^T\varphi(x_i)\right][\alpha] & = & [\beta]\tilde{\Lambda} \end{array}$$

SVD within the LS-SVM setting (4)

- **Theorem [Suykens, 2015].** If compatibility condition

$$ACA = A$$

holds, then the optimality conditions to the primal problem result in the shifted eigenvalue problem

$$\begin{aligned} A[\beta] &= [\alpha]\tilde{\Lambda} \\ A^T[\alpha] &= [\beta]\tilde{\Lambda} \end{aligned}$$

where

- all diagonal elements of $\tilde{\Lambda} = \mathrm{diag}(\lambda_1, ..., \lambda_r)$ are non-zero positive numbers and $\lambda_l = 1/\gamma_l$ for $l = 1, ..., r$ with r the rank of matrix A
- the matrices $[\alpha] \in \mathbb{R}^{N \times r}, [\beta] \in \mathbb{R}^{M \times r}$ consist of r Lagrange multiplier vectors $\alpha = (\alpha_1, ..., \alpha_N)^T$ and $\beta = (\beta_1, ..., \beta_M)^T$ as its columns.

[Suykens, ACHA, 2015, in press]

Choice of compatibility matrix C

- By using the vec operation:

$$(A^T \otimes A)\mathrm{vec}(C) = \mathrm{vec}(A)$$

- For a full rank matrix A:
  - case $N > M$: take $C = A^\dagger = (A^T A)^{-1} A^T$ satisfying $ACA = A$;
  - case $N < M$: take $C^T = (A^T)^\dagger = (AA^T)^{-1} A$;
  - case $N = M$: take $C = A^{-1}$,

- Note: instead of only one compatibility matrix C, one can also take two compatibility matrices C_1, C_2 (related to both φ and ψ)

Primal and dual model representations

- Primal and dual model representations

$$\mathcal{M} \begin{cases} \nearrow & (P): \quad e_i = w^T \varphi(x_i) \\ & \qquad\quad r_j = v^T \psi(z_j) \\ \searrow & (D): \quad e_i = \sum_j \beta_j \psi(z_j)^T \varphi(x_i) \\ & \qquad\quad r_j = \sum_i \alpha_i \varphi(x_i)^T \psi(z_j) \end{cases}$$

- One can consider an out-of-sample extension, corresponding to adding a row or a column to matrix A.

Nonlinear extensions to the SVD

- Construct a new matrix B based on the two data sources $\{x_i\}_{i=1}^N$, $\{z_j\}_{j=1}^M$ obtained from A:

$$B_{ij} = f(x_i, z_j)$$

with map $f : \mathbb{R}^M \times \mathbb{R}^N \to \mathbb{R}$.

- Take **nonlinear mapping based on compatible feature maps**

$$B_{ij} = K(\varphi(x_i), \psi(z_j))$$

with a kernel function $K : \mathbb{R}^q \times \mathbb{R}^q \to \mathbb{R}$ (note: the kernel is not restricted to be a Mercer kernel)

[Suykens, ACHA, 2015, in press]

Example

Matrix $B = [B_{ij}] = [K(\varphi(x_i), \psi(z_j))]$
with $\varphi(x_i) = C^T x_i$, $\psi(z_j) = z_j$, $C^T = (A^T)^\dagger$

Examples of different kernels K

- **linear:**
$$K(\varphi(x_i), \psi(z_j)) = \varphi(x_i)^T \psi(z_j) = x_i^T C z_j$$
corresponding to the SVD.

- **combination of linear and polynomial:** c, d positive
$$K(\varphi(x_i), \psi(z_j)) = \varphi(x_i)^T \psi(z_j) - c \left(\varphi(x_i)^T \psi(z_j)\right)^d$$

- **exponential:** η positive or negative
$$K(\varphi(x_i), \psi(z_j)) = \exp\left(\eta\, \varphi(x_i)^T \psi(z_j)\right)$$

Example - linear versus nonlinear SVD





SVD





lin+pol





exp, $\eta = 1$





exp, $\eta = -1$

original | 20 comp. | 100 comp.

Conclusions

- **Synergies** parametric and kernel based-modelling
- **Primal and dual** representations
- Sparse kernel models using **fixed-size method**
- **SVD from LS-SVM**, nonlinear extensions to SVD
  Future perspectives: robust SVD, new tensor decompositions
- **Beyond Mercer kernels**

Software: see ERC AdG A-DATADRIVE-B website
www.esat.kuleuven.be/stadius/ADB/software.php

Acknowledgements (1)

- Co-workers at ESAT-STADIUS:

  M. Agudelo, C. Alaiz, C. Alzate, A. Argyriou, R. Castro, J. De Brabanter, K. De Brabanter, L. De Lathauwer, B. De Moor, M. Espinoza, M. Fanuel, Y. Feng, E. Frandi, B. Gauthier, D. Geebelen, H. Hang, X. Huang, L. Houthuys, V. Jumutc, Z. Karevan, R. Langone, Y. Liu, R. Mall, S. Mehrkanoon, M. Novak, J. Puertas, L. Shi, M. Signoretto, V. Van Belle, J. Vandewalle, S. Van Huffel, C. Varon, X. Xi, Y. Yang, and others

- Many people for joint work, discussions, invitations, organizations

- Support from ERC AdG A-DATADRIVE-B, KU Leuven, GOA-MaNet, OPTEC, IUAP DYSCO, FWO projects, IWT, iMinds, BIL, COST

Acknowledgements (2)













Thank you