Ratio-based Multiple Kernel Clustering

Grigorios Tzortzis and Aristidis Likas

Department of Computer Science & Engineering
University of Ioannina, Greece

Outline

Outline

Multiple Kernel Learning

Many popular machine learning algorithms utilize the kernel trick and perform learning in feature space

- SVM, kernel k-means, kernel PCA

Multiple Kernel Learning

Many popular machine learning algorithms utilize the kernel trick and perform learning in feature space

- SVM, kernel k-means, kernel PCA

Kernel Trick

- To exploit nonlinearities in the data, very often:
  - Instances are mapped from input space to a higher dimensional feature space $\mathcal{H}$ via a nonlinear transformation ϕ
  - Learning is executed in feature space instead of input space
- A kernel $\mathcal{K}$ is employed to get the inner products in feature space without explicitly defining transformation ϕ
  - The transformation is intractable for certain kernels
- Examples of kernels
  - RBF kernel: $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2\right)$
  - Polynomial kernel: $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \left(\mathbf{x}_i^\top \mathbf{x}_j + \gamma\right)^\delta$

Multiple Kernel Learning

The efficacy of kernel-based methods is dependent on the choice of an appropriate kernel for the underlying problem

- An unsuitable kernel can significantly degrade results
- The best kernel is not known in advance

Multiple kernel learning (MKL) tackles the kernel selection problem and aims at inferring a kernel that suits the data

- A parametric form for the kernel is assumed
- Appropriate values for the parameters are estimated during training

Multiple Kernel Learning

- The kernel, $\widetilde{\mathcal{K}}$, is parametrized as a combination of some predefined, called basis, kernels $\mathcal{K}^{(v)}$
  - Usually a linear mixture is employed $\widetilde{\mathcal{K}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{v=1}^{V} \theta_v \mathcal{K}^{(v)}(\mathbf{x}_i, \mathbf{x}_j)$
  - More rarely, a nonlinear combination is employed
- Basis kernels are obtained by:
  - Applying different types of kernels on the same instances
  - Using the different views of the instances for multi-view data
- Most existing MKL studies tackle supervised problems
  - Our focus is to perform MKL in the clustering domain, where existing literature is very limited

Supervised MKL

- MKL has been mainly studied under the SVM paradigm
- Locate the hyperplane with the largest margin and also learn an appropriate kernel

- Given a two-class labeled dataset $\mathcal{X} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, $\mathbf{x}_i \in \Re^d$ and $y_i \in \{\pm 1\}$

- Assume a kernel $\widetilde{\mathcal{K}}$ parametrized by a vector $\boldsymbol{\theta}$ of parameters, to which transformation $\widetilde{\phi}$ and feature space $\widetilde{\mathcal{H}}$ correspond
  - A linear combination of basis kernels $\widetilde{\mathcal{K}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{v=1}^{V} \theta_v \mathcal{K}^{(v)}(\mathbf{x}_i, \mathbf{x}_j)$
  - Alternatively, a nonlinear combination

Supervised MKL

Most MKL methods derive from the following margin-based optimization problem:

$$\min_{\boldsymbol{\theta},\mathbf{w},b,\boldsymbol{\xi}} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i,$$

$$s.t. \ \ \theta_v \geq 0, \ \|\boldsymbol{\theta}\|_p^p \leq 1, \ y_i\left(\mathbf{w}^\top\widetilde{\phi}(\mathbf{x}_i) + b\right) \geq 1 - \xi_i, \ \xi_i \geq 0$$

- $\mathbf{w}$, b are the coefficients of the SVM hyperplane
- $\|\mathbf{w}\|$ is the reciprocal of the margin
- $\boldsymbol{\xi}$ is the vector of slack variables capturing the misclassification error
- $C > 0$ is a regularization constant

The norm constraint on $\boldsymbol{\theta}$ is employed to avoid overfitting

- The 1-norm promotes a sparse kernel combination
- Higher p-norms often lead to better results

Unsupervised MKL

Maximum Margin Clustering (MMC)

- MMC[1] extends the large margin principle of SVM to clustering
- Goal: Find a labeling (clustering) $\mathbf{y}$ of dataset $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$, $\mathbf{x}_i \in \Re^d$ that results in the largest margin

$$\min_{\mathbf{y}} \min_{\mathbf{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i,$$

$$s.t. \quad -\ell \leq \sum_{i=1}^{N} y_i \leq \ell,\ \mathbf{y} \in \{\pm 1\}^N,\ y_i\left(\mathbf{w}^\top \phi(\mathbf{x}_i) + b\right) \geq 1 - \xi_i,\ \xi_i \geq 0$$

- Limited to two-cluster problems
- A cluster balance constraint is required to avoid trivial solutions

[1] L. Xu et al., *Maximum margin clustering*, NIPS, 2004

Unsupervised MKL

Maximum Margin Clustering (MMC)

- MMC[1] extends the large margin principle of SVM to clustering
- Goal: Find a labeling (clustering) $\mathbf{y}$ of dataset $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$, $\mathbf{x}_i \in \Re^d$ that results in the largest margin

$$\min_{\mathbf{y}} \min_{\mathbf{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i,$$

$$s.t. \quad -\ell \leq \sum_{i=1}^{N} y_i \leq \ell,\ \mathbf{y} \in \{\pm 1\}^N,\ y_i\left(\mathbf{w}^\top \phi(\mathbf{x}_i) + b\right) \geq 1 - \xi_i,\ \xi_i \geq 0$$

- Limited to two-cluster problems
- A cluster balance constraint is required to avoid trivial solutions

Most unsupervised MKL methods extend the MMC framework to learn a parametric kernel $\widetilde{\mathcal{K}}$ along with the clusters, similar to supervised MKL

[1] L. Xu et al., *Maximum margin clustering*, NIPS, 2004

Maximum Margin Clustering Example

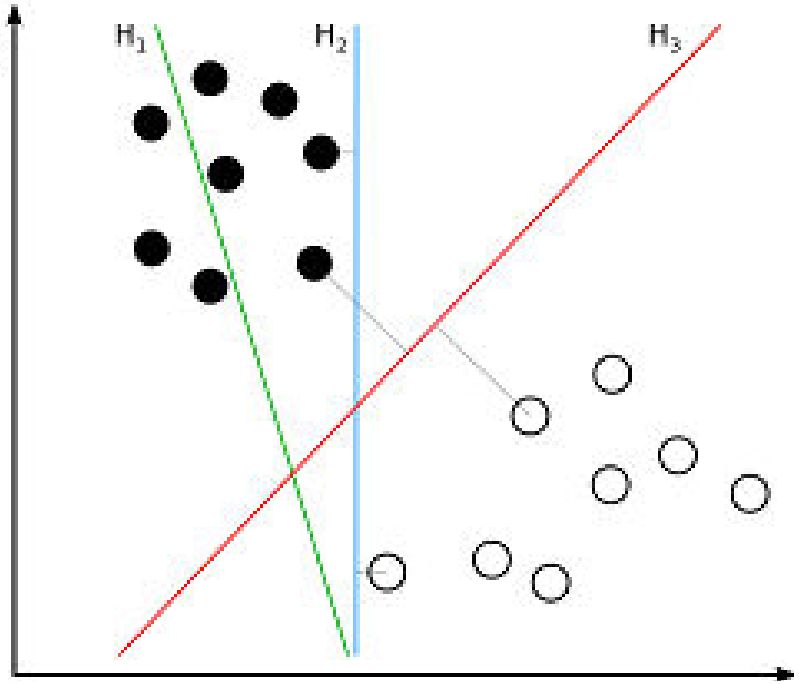


Figure: MMC prefers H3 as it exhibits the largest margin

Scaling Problem

Definition

The margin-based objective $(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i)$ can become arbitrarily small by simply scaling the kernel by a positive scalar $(\widetilde{\mathcal{K}} \rightarrow \alpha\widetilde{\mathcal{K}}, \alpha > 0)$[1]

Implication

The margin-based objective does not suffice to accurately assess the quality of the learned kernel

[1] K. Gai et al., *Learning kernels with radiuses of minimum enclosing balls*, NIPS, 2010

Contribution

We consider multiple kernel clustering and propose an approach that:

- Learns a suitable kernel along with the cluster assignments
- Considers both the separation and the compactness of the clusters
- Is invariant to scalings of the learned kernel
- Is invariant to the type of p-norm constraint on the kernel parameters

Motivation

- Most MKL approaches optimize the margin alone (a separation measure)
- The margin suffers from the scaling problem
- Most MKL methods focus on supervised learning

Outline

Ratio-based Multiple Kernel Clustering (RMKC) Formulation

- Given a dataset $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^{N}$, $\mathbf{x}_i \in \Re^d$
- Assume a kernel $\widetilde{\mathcal{K}}$ parametrized by a vector $\boldsymbol{\theta}$ of parameters, to which transformation $\widetilde{\phi}$ and feature space $\widetilde{\mathcal{H}}$ correspond

We utilize the separation and the compactness of the clusters to perform multiple kernel clustering and:

- Infer the cluster labels
- Learn appropriate values for the kernel $\widetilde{\mathcal{K}}$ parameters

RMKC Formulation

Objective

Minimize the ratio between the margin (MMC objective) and the intra-cluster variance in feature space $\widetilde{\mathcal{H}}$ (kernel k-means objective):

$$\min_{\boldsymbol{\theta},\mathbf{y}} \mathcal{J}(\boldsymbol{\theta},\mathbf{y}),\ s.t.\ \ \theta_v \geq 0,\ \|\boldsymbol{\theta}\|_p^p = 1,\ -\ell \leq \sum_{i=1}^{N} y_i \leq \ell,\ \mathbf{y} \in \{\pm 1\}^N$$

$$\mathcal{J}(\boldsymbol{\theta},\mathbf{y}) = \min_{\mathbf{w},b,\boldsymbol{\xi}} \ \frac{1}{2}\mathcal{E}(\boldsymbol{\theta},\mathbf{y})\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i,\ s.t.\ \ y_i\left(\mathbf{w}^\top\widetilde{\phi}(\mathbf{x}_i)+b\right) \geq 1-\xi_i,\ \xi_i \geq 0$$

- $\mathcal{E}(\boldsymbol{\theta},\mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{2}\delta_{ik}\|\widetilde{\phi}(\mathbf{x}_i) - \widetilde{\mathbf{m}}_k\|^2$ is the intra-cluster variance
  - $\delta_{ik} = 1$ if $y_i = 2k-3$ and $\delta_{ik} = 0$ otherwise, $\widetilde{\mathbf{m}}_k = \frac{\sum_{i=1}^{N}\delta_{ik}\widetilde{\phi}(\mathbf{x}_i)}{\sum_{i=1}^{N}\delta_{ik}}$
- A cluster balance constraint is required to avoid trivial solutions
- The norm constraint on $\boldsymbol{\theta}$ is employed to avoid overfitting

RMKC Objective Analysis

$$\min_{\boldsymbol{\theta},\mathbf{y}} \mathcal{J}(\boldsymbol{\theta},\mathbf{y}),\ s.t.\ \ \theta_v \geq 0,\ \|\boldsymbol{\theta}\|_p^p = 1,\ -\ell \leq \sum_{i=1}^{N} y_i \leq \ell,\ \mathbf{y} \in \{\pm 1\}^N$$

$$\mathcal{J}(\boldsymbol{\theta},\mathbf{y}) = \min_{\mathbf{w},b,\boldsymbol{\xi}} \ \frac{1}{2}\mathcal{E}(\boldsymbol{\theta},\mathbf{y})\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i,\ s.t.\ \ y_i\left(\mathbf{w}^\top \widetilde{\phi}(\mathbf{x}_i) + b\right) \geq 1 - \xi_i,\ \xi_i \geq 0$$

Search for a pair of $\{\boldsymbol{\theta},\mathbf{y}\}$ values that yields a small variance to margin ratio $(\mathcal{E}(\boldsymbol{\theta},\mathbf{y})\|\mathbf{w}\|^2)$

- Both the separation (margin) and the compactness (intra-cluster variance) are considered
  - Better partitionings can be possibly obtained compared to approaches that rely on either of the two
- Limited to two-cluster problems

RMKC Properties

Scale Invariance

- The RMKC objective is invariant to scalings of the kernel $\widetilde{\mathcal{K}}$
- It can accurately capture the quality of the learned kernel

RMKC Properties

Scale Invariance

- The RMKC objective is invariant to scalings of the kernel $\widetilde{\mathcal{K}}$
- It can accurately capture the quality of the learned kernel

Norm Invariance

- The RMKC formulation is invariant to the type of p-norm constraint on the kernel parameters $\boldsymbol{\theta}$
  - This only holds for the global optimum solution
  - This only holds when considering linear combinations of basis kernels
- The choice of p-norm becomes less crucial
  - Different p-norms may still produce different local optimum solutions, but the global optimum is the same
  - The norm constraint can be even dropped without affecting the global optimum solution

RMKC Training

Iteratively update the clusters and the kernel parameters to locate a local optimum that depends on the initialization of $\{\boldsymbol{\theta}, \mathbf{y}\}$

Evaluating the Objective Function

Use the dual to compute the $\mathcal{J}(\boldsymbol{\theta}, \mathbf{y})$ value for some fixed $\{\boldsymbol{\theta}, \mathbf{y}\}$:

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{N} \alpha_i - \frac{1}{2\mathcal{E}(\boldsymbol{\theta}, \mathbf{y})} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \widetilde{K}_{ij}, \; s.t. \; 0 \leq \alpha_i \leq C, \; \sum_{i=1}^{N} \alpha_i y_i = 0$$

- The optimal solution $\boldsymbol{\alpha}^*$ of the dual can be found by employing a standard SVM solver
- Strong duality holds:
  $\mathcal{J}(\boldsymbol{\theta}, \mathbf{y}) = \sum_{i=1}^{N} \alpha_i^* - \frac{1}{2\mathcal{E}(\boldsymbol{\theta}, \mathbf{y})} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i^* \alpha_j^* y_i y_j \widetilde{K}_{ij}$

RMKC Training - Updating the Kernel Parameters

- The cluster labels $\mathbf{y}$ are kept fixed and the parameters $\boldsymbol{\theta}$ are updated using gradient descent
- The gradient is calculated based on the dual:

$$\frac{\partial \mathcal{J}(\boldsymbol{\theta}, \mathbf{y})}{\partial \theta_v} = \frac{1}{2\mathcal{E}(\boldsymbol{\theta}, \mathbf{y})^2} \sum_{i,j=1}^{N} \alpha_i^* \alpha_j^* y_i y_j \widetilde{K}_{ij} \frac{\partial \mathcal{E}(\boldsymbol{\theta}, \mathbf{y})}{\partial \theta_v} - \frac{1}{2\mathcal{E}(\boldsymbol{\theta}, \mathbf{y})} \sum_{i,j=1}^{N} \alpha_i^* \alpha_j^* y_i y_j \frac{\partial \widetilde{K}_{ij}}{\partial \theta_v}$$

To update the kernel parameters:

- Solve the dual for the current $\{\boldsymbol{\theta}, \mathbf{y}\}$ values to get the gradient
- Take a step along the gradient
- Project the parameters back to their feasible set $(\theta_v \geq 0, \|\boldsymbol{\theta}\|_p^p = 1)$

RMKC Training - Updating the Clusters

- The kernel parameters $\boldsymbol{\theta}$ are kept fixed and the cluster labels $\mathbf{y}$ are updated using a practical search framework
- A sequence of L candidate cluster label vectors $\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(L)}$, ($L$ is user-defined) is constructed, where instances are moved from one cluster to the other
  - Starting from the current labels, $\mathbf{y}^{(0)}$, this sequence is incrementally built
  - Compared to the previous candidate label vector in the sequence, the next contains one more instance whose cluster label has been changed
  - Compared to $\mathbf{y}^{(0)}$, $\mathbf{y}^{(l)}$ contains l instances whose label has changed
- $\mathbf{y}$ is updated by selecting the label vector $\mathbf{y}^{(l^*)}$ attaining the smallest objective value ($l^* = \text{argmin}_{0 \leq l \leq L} \mathcal{J}(\boldsymbol{\theta}, \mathbf{y}^{(l)})$)
  - Eventually, l^* instances swap clusters

RMKC Training - Updating the Clusters

- Given $\mathbf{y}^{(l)}$, how $\mathbf{y}^{(l+1)}$ is constructed?
  - Find the hyperplane corresponding to $\mathbf{y}^{(l)}$ by solving the dual
  - Select as the $(l+1)$-th instance to change clusters the one we are the less confident about its labeling, according to the hyperplane

Kernel Parametrization

The RMKC formulation can handle both the popular linear combination of basis kernels

$$\widetilde{\mathcal{K}}(\mathbf{x}_i, \mathbf{x}_j) = \sum_{v=1}^{V} \theta_v \mathcal{K}^{(v)}(\mathbf{x}_i, \mathbf{x}_j)$$

and more general forms of parametric kernels, such as nonlinear mixtures of basis kernels

- Existing MKL approaches are usually limited to a specific form for the parametric kernel

Prerequisite

The gradient of the RMKC objective must exist for the selected parametrization of kernel $\widetilde{\mathcal{K}}$

Outline

Evaluation

- We compare RMKC with the linear combination of basis kernels to:
  1. Kernel k-means
  2. IterSVR[1], a margin-based MMC approach
  3. Multi-view kernel k-means (MVKKM)[2], a variance-based MKL approach
  4. Multi-view spectral clustering (MVSpec)[2], a variance-based MKL approach
- We consider both single view and multi-view datasets
  - For multi-view datasets a linear basis kernel is employed for each view
  - For single view datasets 10 RBF basis kernels with varying σ are constructed
- Kernel k-means and iterSVR do not perform kernel learning
  - We report the performance of the best basis kernel
  - For iterSVR we also report the average performance over all basis kernels

[1] K. Zhang et al., *Maximum margin clustering made practical*, ICML, 2007
[2] G. Tzortzis, A. Likas, *Kernel-based weighted multi-view clustering*, ICDM, 2012

Evaluation

- For all datasets we extract several two-class subsets and search for two clusters
- Performance is measured in terms of clustering accuracy
  - Higher accuracy values indicate a better match between cluster and class labels
  - RMKC, iterSVR and kernel k-means are restarted 30 times and their average accuracy is reported
- Parameter Configuration
  - RMKC $\rightarrow L = 30$, $\ell = 0.5N$, grid search for C $(\{10^{-2}, 10^{-1}, \ldots, 10^{2}\})$
  - IterSVR $\rightarrow \ell = 0.03N$ (for unbalanced datasets $\ell = 0.3N$), grid search for C $(\{10^{-2}, 10^{-1}, \ldots, 10^{2}\})$
  - MVKKM & MVSpec $\rightarrow$ grid search for the sparsity controlling parameter p $(\{1, 1.5, \ldots, 5\})$

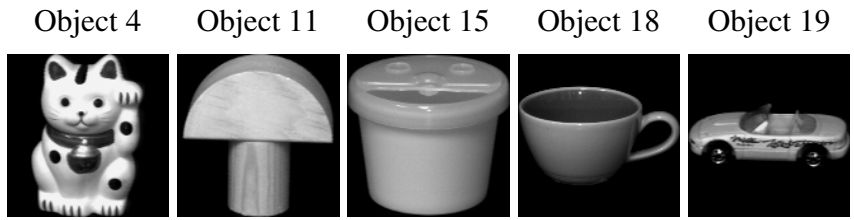
Datasets

Multiple Features/Optdigits – Collections of handwritten digits

- Five views / A single view
- Ten classes for both

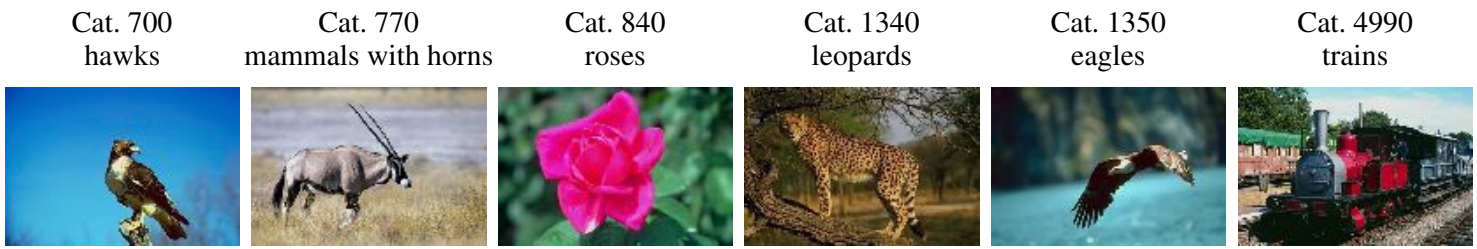
Coil-20 – Image collection

- A single view
- 20 classes



Corel – Image collection

- Seven views (color and texture related views)
- 34 classes



Norm Invariance in Practice

The RMKC formulation is norm invariant on its global optimum solution. What about the local optimum solutions obtained during training?

Table: RMKC results for different p-norm constraints

Dataset	No-norm	1-norm	2-norm
COIL-20	98.75 ± 2.60	98.61 ± 2.65	98.43 ± 2.73
Corel	94.55 ± 1.62	94.64 ± 1.58	94.69 ± 1.62
Multiple features	99.58 ± 0.22	99.53 ± 0.37	99.59 ± 0.23
Optdigits	97.77 ± 2.45	97.65 ± 2.71	97.75 ± 2.50

- The solutions obtained for the different p-norms are very similar
- Local optima are not significantly influenced by the choice of p-norm in practice

Comparative Results on the Image Collections

Table: Image clustering results

Dataset	RMKC (1-norm)	MVKKM	MVSpec	IterSVR (best)	IterSVR (average)
COIL-20					
4-11	$\mathbf{100.00 \pm 0.00}$	77.78	**100.00**	98.47 ± 8.37	98.34 ± 8.34
15-18	$\mathbf{100.00 \pm 0.00}$	90.28	95.83	99.72 ± 0.35	99.21 ± 0.21
15-19	$\mathbf{94.44 \pm 10.59}$	68.06	86.11	93.43 ± 14.30	91.86 ± 14.52
Corel					
700-4990	$\mathbf{97.62 \pm 0.65}$	95.00	95.00	96.43 ± 0.25	83.19 ± 1.85
770-840	$\mathbf{97.55 \pm 0.91}$	94.50	90.00	94.20 ± 3.04	87.85 ± 0.58
1340-1350	$\mathbf{95.50 \pm 0.00}$	95.00	95.00	92.50 ± 0.00	83.71 ± 0.00

- Kernel k-means attains the least accuracy in general (not shown here)
- RMKC outperforms the compared methods
- IterSVR is its closest competitor for COIL-20 and MVKKM for Corel
- IterSVR is competitive provided the optimal basis kernel is used, which is, however, not a priori known in practice

Comparative Results on the Handwritten Digits

Table: Handwritten digits clustering results

Dataset	RMKC (1-norm)	MVKKM	MVSpec	IterSVR (best)	IterSVR (average)
Mult. feat.					
1-7	99.62 ± 0.78	98.75	98.75	$\mathbf{99.75 \pm 0.00}$	96.85 ± 0.00
2-3	$\mathbf{99.70 \pm 0.23}$	99.25	99.00	99.50 ± 0.00	94.13 ± 7.16
6-8	$\mathbf{99.15 \pm 0.33}$	97.25	98.50	99.00 ± 0.00	94.94 ± 6.47
Optdigits					
1-7	99.56 ± 1.41	**100.00**	**100.00**	96.93 ± 9.83	94.26 ± 13.14
2-3	96.29 ± 5.44	90.56	88.89	$\mathbf{96.50 \pm 0.82}$	95.59 ± 2.70
6-8	$\mathbf{99.89 \pm 0.14}$	99.15	98.87	99.72 ± 0.00	99.45 ± 0.06

- Kernel k-means attains the least accuracy in general (not shown here)
- The best performance is shared between RMKC and iterSVR
- IterSVR is competitive provided the optimal basis kernel is used, which is, however, not a priori known in practice
  - IterSVR results degrade if an inappropriate basis kernel is chosen

Outline

Summary

- We proposed RMKC, an unsupervised MKL method that:
  - Assigns instances to clusters
  - Learns an appropriate kernel for the data
- Both the separation and the compactness of the clusters are considered during training
- RMKC is invariant to:
  - Scalings of the learned kernel
  - The type of p-norm constraint on the kernel parameters
- RMKC can handle various forms of parametric kernels
  - Linear combinations of basis kernels
  - Nonlinear combinations of basis kernels

Thank you!