Supervised Learning with Similarity Functions

Purushottam Kar¹ and Prateek Jain²

¹Indian Institute of Technology, Kanpur, UP, INDIA
²Microsoft Research India, Bangalore, KA, INDIA

Introduction

Goal: Supervised learning with indefinite kernels
Why use indefinite kernels?
- Several domains possess natural notions of similarity
  - Bioinformatics: B.L.A.S.T. scores for protein sequences
  - OCR: tangent distance similarity measures
  - Image retrieval: earth mover’s distance
Satisfiability for Mercer’s theorem a hard-to-verify property
Not clear why non psd-ness should limit usability of a kernel

Existing work

Most works address only the problem of classification
Broadly three main approaches
- Use indefinite kernels directly [1]: results in non-convex formulations
- Find a proxy PSD kernel [2]: expensive + loss of domain knowledge
- Use kernel-task alignment [3]: efficient + generalization guarantees
Several results for classification using the third approach [3, 4, 5]

Our contributions

Propose a notion of kernel “goodness” for general supervised learning
- Previous notions obtained as a special case
- Develop landmarking-based algorithms to perform supervised learning
  - Consider three tasks: real regression, ordinal regression, ranking
  - Provide generalization bounds
  - Apply sparse learning techniques to reduce landmark complexity
  - Fast testing times + generalization guarantees
  - Experimental evaluation of landmarking based techniques

What is a good similarity function

“Margin” view: positives closer to positives than negatives by a margin
- Cannot be extended for other supervised learning problems
  - We take a “target value” view
  - Target value at a point recoverable from neighbors of the point
  - Implicitly ensures a smoothness prior

Definition 1. Good similarity function

A similarity function \( K: X \times X \to \mathbb{R} \) is \((\epsilon, \gamma, B)\)-good for a learning task \( y: X \to \mathbb{Y} \) for some bounded weighting function \( w: X \to [-B, B] \), for at least a \((1 - \epsilon)\) fraction of the domain, we have \( y(x) = \mathbb{E}_{x' \sim Y} w(x') y(x' x) \).

Need to modify a bit to incorporate surrogate loss functions
Can be adapted to various learning tasks using appropriate loss functions
Reduces to earlier notion [3] for binary classification

Evaluating the model

- The proposed notion of goodness is evaluated on two grounds
- Utility: “good” similarity functions should yield effective predictors

Definition 2. Utility criterion

A similarity function \( K \) is \((\epsilon, \gamma)\)-useful w.r.t. a loss function \( l(\cdot, \cdot) \) if for any \( \epsilon > 0 \), using polynomially many labeled and unlabeled samples, one can w.h.p. generate a hypothesis \( \hat{f}(x, K) \) s.t. \( \mathbb{E}_{x \sim D} \left[ l(\hat{f}(x), y(x)) \right] \leq \epsilon_0 + \epsilon_1 \).

Admissibility: PSD kernels with large margin should remain “good”

Definition 3. Good PSD Kernel

A kernel \( K \) with RKHS \( H_K \) and feature map \( \phi: \mathcal{X} \to H_K \) is \((\epsilon, \gamma, \ell)\)-good w.r.t. loss function \( \ell \) if for some \( W^* \in H_K \), we have \( \mathbb{E}_{x \sim \mathcal{D}} \left[ \ell(W^* \phi(x), y(x)) \right] < \epsilon_0 \).

Learning with similarity functions

Algorithm 4. (Landmarking based learning algorithm)

- Given: An \((\epsilon, B, \gamma)\)-good kernel \( K \) and training points \( \mathcal{T} = \{(x_i, y_i)\}_{i=1}^n \)
- Sample \( d \) unlabeled landmarks from domain \( \mathcal{L} = \{x'_1, \ldots, x'_d\} \)
- Let \( \psi_i: \mathcal{X} \to H_K \) be \( K(x, x'_i) \in \mathcal{R}^d \)
- Obtain \( \hat{w} = \arg \min_{\hat{w}} \sum_{i=1}^d \ell(\hat{w}, \psi_i(x'_i), y_i) \)
- Output: \( \hat{f} = x \mapsto \hat{w} \psi_i(x) \)

Landmarks can be subsampled from training points themselves
- Provide generalization guarantees for such “double-dipping”
- Sparse Regression: often only a small fraction of landmarks are useful
  - Landmark pruning essential for fast predictors
  - Propose modified model that takes into account only “useful” landmarks
  - Use sparse learning techniques [6] to learn a predictor
  - Utility guarantee ensures sparsity as well as generalization error bounds

References


Overview of theoretical guarantees

<table>
<thead>
<tr>
<th>Task</th>
<th>Utility</th>
<th>Samples required</th>
<th>Admissibility for ((\epsilon, \gamma))-good kernel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classification</td>
<td>(l(\cdot, \cdot))</td>
<td>(O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon \gamma} + \frac{1}{\epsilon_1}\right))</td>
<td>(O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon \gamma} + \frac{1}{\epsilon_1}\right))</td>
</tr>
<tr>
<td>Regression</td>
<td>(l(\cdot, \cdot))</td>
<td>(O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon \gamma} + \frac{1}{\epsilon_1}\right))</td>
<td>(O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon \gamma} + \frac{1}{\epsilon_1}\right))</td>
</tr>
<tr>
<td>Ordinal</td>
<td>(l(\cdot, \cdot))</td>
<td>(O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon \gamma} + \frac{1}{\epsilon_1}\right))</td>
<td>(O\left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon \gamma} + \frac{1}{\epsilon_1}\right))</td>
</tr>
</tbody>
</table>

Experimental results

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Sigmoid kernels</th>
<th>Manhattan kernels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Malaria</td>
<td>2.1e-002</td>
<td>6.2e-003</td>
</tr>
<tr>
<td>DMissing</td>
<td>5.9e-002</td>
<td>1.6e-002</td>
</tr>
<tr>
<td>CPData</td>
<td>4.1e-002</td>
<td>1.4e-003</td>
</tr>
<tr>
<td>Felsonaly</td>
<td>1.8e-001</td>
<td>1.4e-002</td>
</tr>
</tbody>
</table>

Table: MSE for real regression: Kernel regression vs. Sparse learning

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Sigmoid kernels</th>
<th>Manhattan kernels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wine-Red</td>
<td>6.8e-001</td>
<td>4.2e-001</td>
</tr>
<tr>
<td>Wine-White</td>
<td>6.2e-001</td>
<td>8.9e-001</td>
</tr>
<tr>
<td>Bank-32</td>
<td>2.7e-000</td>
<td>1.6e-000</td>
</tr>
<tr>
<td>Horse-16</td>
<td>2.7e-000</td>
<td>1.5e-000</td>
</tr>
</tbody>
</table>

Table: MSE for ordinal regression: Kernel regression vs. Landmarking

Figure: MSE for landmarking (RegLand), sparse landmarking (RegLand-Sp) and kernel regression (KR)

Figure: Absolute error for landmarking (ORLand) and kernel regression (KR)