Abstract

The Kernel Group LASSO is an $\ell_1/\ell_2$ regularized (structured sparse) $\ell_2$ reconstruction problem, which performs well at multi-label classification and is defined in a Reproducing Kernel Hilbert Space. Unfortunately, computing the ground truth solution to this task is slow for real-time applications even with state-of-the-art optimization schemes like the Fast Shrinkage Thresholding Algorithm. We extend the Learned Iterative Shrinkage Thresholding Algorithm – a fast neural network introduced by Gregory and LeCun – to estimate the true result. We test our method in time series classification by training on the 6D Motion Gesture Database while utilizing the Global Alignment time series kernel.

Kernel Group LASSO

Denote by $X \neq \emptyset$ a set where the symmetric, positive semidefinite normalized kernel function $k: X \times X \rightarrow \mathbb{R}$ is defined:

$$k(x,y) = \frac{\langle \varphi(x), \varphi(y) \rangle}{\sqrt{\langle \varphi(x), \varphi(x) \rangle \langle \varphi(y), \varphi(y) \rangle}}, \quad \forall (x, y) \in X \times X$$

for reconstructing Kernel Hilbert Space $\mathcal{H}$ and feature mapping $\varphi: X \rightarrow \mathcal{H}$. Let $x \in X$ and $\alpha = (\alpha_1, \ldots, \alpha_d) \in X$ be a signal and a vector system (i.e., a dictionary) respectively. Then for group structure $G \subseteq \{1, \ldots, N\}$, $\lambda > 0$, consider the Kernel Group LASSO (KGLASSO) procedure [1]:

$$\alpha^* = \arg \min_{\alpha \in \mathbb{R}^d} \frac{1}{2} k(D, D) \alpha - k(D, x) \alpha + \lambda \sum_{G \in G} \| \mathcal{G} \alpha \|_2^2, \quad (2)$$

i.e., we aim to reconstruct $\varphi(x)$ with group sparse linear combination of $\varphi(D)$ within $\mathcal{H}$. This is an $\ell_1/\ell_2$ regularized quadratic programing problem that can be solved by the Fast Iterative Shrinkage Thresholding Algorithm (FISTA) [2] after precomputing $k(D, D)$ and $k(D, x)$.

Advantages:
- Kernels can discover more subtle similarities and make the system undercomplete.
- Group structure can reduce the problem size by choosing from fewer variables.

Limitations:
- FISTA still has significant time complexity as it requires several iterations.
- KGLASSO scales quadratically in dictionary size $N$.
- KGLASSO scales linearly in signal count.
- Kernel computations can be very slow and often lead to dense matrices, which are difficult to store and deal with.

Learned Iterative Shrinkage Thresholding Algorithm

Due to the FISTA limitation above, supervised approximation schemes to it have gained attention recently. The Learned Iterative Shrinkage Thresholding Algorithm (LISTA) was proposed in [3]: a neural network for estimating the sparse code of the original linear and unstructured LASSO [4], where:

- adaptive soft-thresholding activation function was introduced to yield true sparse outputs with respect to tunable thresholds $\theta \in \mathbb{R}^K$.
- competition between dictionary elements was introduced by making the network recurrent with fixed depth $K \in \mathbb{N}$.

$$\alpha_0 = 0, \quad \alpha_K = h_0(Wx + S\alpha^{K-1}), \quad k = 1, \ldots, K, \quad (4)$$

i.e., signal $x$ is mapped to code space right away with $W$ and then $S$ rules out some of the active elements.

- minimization of $\ell_2$ loss was carried out in parameters $W, S, \theta$ among a batch of training samples $(x^{(i)}, y^{(i)}), i = 1, \ldots, M$ with stochastic gradient descent and backpropagation through time:

$$L(W, S, \theta) = \sum_{i=1}^{M} \| h_0(\| \alpha_i \|_2) - \alpha_i \|^2_2.$$  

Advantages:
- Matrix multiplications and soft-thresholding are fast.
- The algorithm has an adjustable iteration count $K$.
- The method is sparsity adaptive, as $\theta$ is learnable.

Conclusion

Our findings:
- LISTA can generalize to the structured and kernelized KGLASSO case.
- Single layer is already capable.
- Mapping from kernels $k(D, x)$ is very accurate.
- Mapping from signals $x$ the performance is considerable.

Results

Micro-averaged multi-label classification results were as follows.

<table>
<thead>
<tr>
<th>$K$</th>
<th>$K+1$</th>
<th>$K+2$</th>
<th>$K+3$</th>
<th>$K+4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss function value</td>
<td>0.300</td>
<td>0.329</td>
<td>0.497</td>
<td>1.077</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.962</td>
<td>0.961</td>
<td>0.793</td>
<td>0.850</td>
</tr>
<tr>
<td>Precision</td>
<td>0.930</td>
<td>0.924</td>
<td>0.624</td>
<td>0.761</td>
</tr>
<tr>
<td>Recall</td>
<td>0.950</td>
<td>0.954</td>
<td>0.418</td>
<td>0.712</td>
</tr>
<tr>
<td>$F_1$ score</td>
<td>0.940</td>
<td>0.939</td>
<td>0.555</td>
<td>0.745</td>
</tr>
</tbody>
</table>

References


This Activity is financially supported by EIT Digital.