Large scale greedy feature-selection for multi-target learning

Antti Airola, Tapio Pahikkala et al.

ECML 2015 BigTargets Workshop
Joint work with many authors

- University of Turku: Antti Airola, Pekka Naula, Tapio Pahikkala, Tapio Salakoski (Multi-target greedy RLS)
Overview

- Large scale feature selection for multi-target learning
- Task: select minimal set of common features allowing accurate predictions over target tasks
- Greedy RLS: greedy regularized least-squares
- Linear time (#inputs, #features, #outputs, #selected)
- Highlights from experiments
  - Broad-DREAM Gene Essentiality Prediction Challenge
  - Outperforms multi-task Lasso for small feature budgets
- Also scales to full Genome Wide Association Studies; thousands of samples, hundreds of thousands of features (recent PhD thesis: Sebastian Okser)
Why feature selection?

1. **Accuracy**: regularizing effect, avoiding overfitting leads to better generalization
2. **Interpretability**: obtain a small set of features understandable by human expert
3. **Budget constraints**: obtaining features costs time and money
Model sparsity

\[ W_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 \end{pmatrix} , \quad W_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 3 & -1 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 3 & 1 & 4 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \]

- features x targets coefficient matrices
- \( W_1 \) 8 features needed for prediction
- \( W_2 \) 2 features needed for prediction
Learning task

Least-squares formulation

\[ \text{arg min}_{W \in \mathbb{R}^{d \times t}} \| XW - Y \|_F^2 \]
subject to \( C(W) \)

Notation

- **X**: data matrix
- **Y**: output matrix
- **W**: model coefficients
- \( \| \cdot \|_F \): Frobenius norm
- \( C(\cdot) \): Constraint (regularizer)
Multi-task Lasso (Zhang, 2006)

\[
\begin{align*}
\text{arg min}_{\mathbf{W} \in \mathbb{R}^{d \times t}} & \quad \| \mathbf{XW} - \mathbf{Y} \|_F^2 \\
\text{subject to} & \quad \sum_{i=1}^{d} \max_{j} |W_{i,j}| \leq r
\end{align*}
\]

- \(L_{1,\infty}\) norm enforces sparsity in the number of features
- \(r > 0\) regularization parameter
Greedy RLS

Greedy RLS (proposed)

\[
\text{arg min}_{W \in \mathbb{R}^{d \times t}} \|XW - Y\|_F^2 \\
\text{subject to } \|W\|_F^2 < r \quad \text{and} \\
\{|i | \exists j, W_{i,j} \neq 0\| \leq k
\]

- \( r > 0 \) regularization parameter
- \( k > 0 \) constraint on the number of features
- heuristics needed to search over the power set of features
Greedy regularized least-squares (Greedy RLS)
Starting from empty feature set, at each point add the feature reducing leave-one-out cross-validation error most
Stop once $k$ features have been selected
Algorithm 1 Multi-target greedy RLS

1: \( S \leftarrow \emptyset \) \quad \triangleright \text{selected features common for all tasks}
2: \textbf{while} \( |S| < k \) \textbf{do} \quad \triangleright \text{select } k \text{ features}
3: \( e \leftarrow \infty \)
4: \( b \leftarrow 0 \)
5: \textbf{for} \( i \in \{1, \ldots, d\} \setminus S \) \textbf{do} \quad \triangleright \text{test all features}
6: \quad e_{\text{avg}} \leftarrow 0
7: \quad \textbf{for} \( j \in \{1, \ldots, t\} \) \textbf{do} \quad \triangleright \text{LOO for task } j
8: \quad \quad e_{i,j} \leftarrow \mathcal{L}(X_{:, S \cup \{i\}}, Y_{:, j}) \quad \triangleright \text{LOO for task } j
9: \quad \quad e_{\text{avg}} \leftarrow e_{\text{avg}} + e_{i,j}/t
10: \quad \textbf{if} \ e_{\text{avg}} < e \textbf{then} \quad \triangleright \text{feature with lowest LOO-error}
11: \quad \quad e \leftarrow e_{\text{avg}}
12: \quad \quad b \leftarrow i
13: \quad S \leftarrow S \cup \{b\} \quad \triangleright \text{feature with lowest LOO-error}
14: \quad W \leftarrow \mathcal{A}(X_{:, S}, Y) \quad \triangleright \text{train final models}
15: \textbf{return} \( W, S \)
• Greedy RLS could be implemented as a general wrapper code calling a black-box solver
• $\#\text{selected} \times \#\text{features} \times \#\text{targets} \times \#\text{CV-rounds}$ calls for naive implementation!
• Matrix algebraic optimization for feature addition, leave-one-out... (for all targets simultaneously)
• Linear time algorithm ($\#\text{inputs}, \#\text{features}, \#\text{outputs}, \#\text{selected}$)
Algorithm 2 Multi-target greedy RLS

\[
\begin{align*}
A & \leftarrow \lambda^{-1}Y \\
g & \leftarrow \lambda^{-1}1 \\
C & \leftarrow \lambda^{-1}X \\
S & \leftarrow \emptyset \\
\text{while } |S| < k \text{ do} \\
& \quad e \leftarrow \infty \\
& \quad b \leftarrow 0 \\
& \quad \text{for } i \in \{1, \ldots, d\} \setminus S \text{ do} \\
& \quad \quad u \leftarrow C_{:,i}(1 + (X_{:,i})^T C_{:,i})^{-1} \\
& \quad \quad e_i \leftarrow 0 \\
& \quad \quad \tilde{A} \leftarrow A - u((X_{:,i})^T A) \\
& \quad \quad \text{for } h \in \{1, \ldots, t\} \text{ do} \\
& \quad \quad \quad \text{for } j \in \{1, \ldots, n\} \text{ do} \\
& \quad \quad \quad \quad \tilde{g}_j \leftarrow g_j - u_j C_{j,i} \\
& \quad \quad \quad \quad e_i \leftarrow e_i + (\tilde{g}_j)^{-2}(\tilde{A}_{j,h})^2 \\
& \quad \quad \quad \text{if } e_i < e \text{ then} \\
& \quad \quad \quad \quad e \leftarrow e_i \\
& \quad \quad \quad \quad b \leftarrow i \\
& \quad \quad \quad u \leftarrow C_{:,b}(1 + (X_{:,b})^T C_{:,b})^{-1} \\
& \quad \quad \quad A \leftarrow A - u((X_{:,b})^T A) \\
& \quad \quad \quad \text{for } j \in \{1, \ldots, n\} \text{ do} \\
& \quad \quad \quad \quad g_j \leftarrow g_j - u_j C_{j,b} \\
& \quad \quad \quad C \leftarrow C - u((X_{:,b})^T C) \\
& \quad \quad \quad S \leftarrow S \cup \{b\} \\
W & \leftarrow (X_{:,S})^T A
\end{align*}
\]
Table: Mulan datasets (Tsoumakas et al. 2011).

<table>
<thead>
<tr>
<th>Data sets</th>
<th>domain</th>
<th>labels</th>
<th>features</th>
<th>instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene</td>
<td>image</td>
<td>6</td>
<td>294</td>
<td>2407</td>
</tr>
<tr>
<td>Yeast</td>
<td>biology</td>
<td>14</td>
<td>103</td>
<td>2417</td>
</tr>
<tr>
<td>Emotions</td>
<td>music</td>
<td>6</td>
<td>72</td>
<td>593</td>
</tr>
<tr>
<td>Mediamill*</td>
<td>text</td>
<td>9</td>
<td>120</td>
<td>41583</td>
</tr>
<tr>
<td>Delicious</td>
<td>text</td>
<td>983</td>
<td>500</td>
<td>16105</td>
</tr>
<tr>
<td>Tmc2007</td>
<td>text</td>
<td>22</td>
<td>49060</td>
<td>28596</td>
</tr>
</tbody>
</table>
Greedy RLS vs. Lasso

Scene data
- MT-Lasso
- ML-gRLS

Yeast data
- MT-Lasso
- ML-gRLS

Large scale greedy feature-selection for multi-target learning
Greedy RLS vs. Lasso

Emotions data
- MT-Lasso
- ML-gRLS

Mediamill data
- MT-Lasso
- ML-gRLS
Greedy RLS vs. Lasso

**Delicious data**
- MT-Lasso
- ML-gRLS

**Tmc2007 data**
- MT-Lasso
- ML-gRLS

Antti Airola, Tapio Pahikkala et al. Large scale greedy feature-selection for multi-target learning
Greedy RLS: linear time algorithm for (multi-target) feature selection

Selects joint features for the target tasks

Competitive, when number of features to be selected small

Applications on Genome-Wide Association Studies

RLScore open source implementation at https://github.com/aatapa/RLScore