Online Multi-Task Learning via Sparse Dictionary Optimization

Paul Ruvolo
Olin College of Engineering

Eric Eaton
University of Pennsylvania

Summary

We developed an efficient online method for learning multiple consecutive tasks based on the K-SVD algorithm for sparse dictionary optimization.

Capabilities of our ELLA-SVD algorithm:

- Learns multiple tasks consecutively
- Transfers knowledge to accelerate learning of new tasks
- Supports a variety of base learning algorithms
- Has lower computational cost than current lifelong learning algorithms
- Supports both task and feature similarity matrices

We demonstrate the effectiveness of ELLA-SVD in lifelong learning settings.

Introduction

Goal: Develop intelligent agents that

1. Quickly learn new tasks
2. Learn continually with experience
3. Exhibit versatility over multiple tasks

	Transfer Learning	Batch Multi-Task Learning
Optimizes performance over	Target task	All tasks
Learns tasks consecutively	Yes, efficiently	Very inefficiently
Computational cost	Low	High

Lifelong learning includes elements of both transfer and multi-task learning

This work investigates a formulation of online multi-task learning (MTL) based on sparse dictionary optimization.

This approach builds upon our earlier work on the Efficient Lifelong Learning Algorithm (ELLA) [Ruvolo & Eaton, ICML '13].

Background: Dictionary Learning for Sparse Coding via K-SVD

Goal: Given a data set $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbb{R}^d$, output a dictionary $\mathbf{L} \in \mathbb{R}^{d \times k}$ that sparse codes the data by solving:

$$\arg\min_{\mathbf{L}} \sum_{i=1}^{n} \min_{\mathbf{s}^{(i)}} \left\{ \|\mathbf{L}\mathbf{s}^{(i)} - \mathbf{x}_i\|_2^2 + \mu \|\mathbf{s}^{(i)}\|_0 \right\}$$

The K-SVD Algorithm

Iterate two steps until convergence to yield $\mathbf{L}$:

Step 1: update codes for each point

$$\mathbf{s}^{(i)} \leftarrow \arg\min_{\mathbf{s}} \left\{ \|\mathbf{L}\mathbf{s} - \mathbf{x}_i\|_2^2 + \mu \|\mathbf{s}\|_0 \right\}$$

Step 2: update each basis vector and the weights of the data points that utilize this basis vector

$$m \in \mathcal{A} \Leftrightarrow \mathbf{s}_j^{(m)} \neq 0$$

$$\mathbf{l}_j, \mathbf{s}_j^{(\mathcal{A})} \leftarrow \arg\min_{\mathbf{l}_j, \mathbf{s}_j^{(\mathcal{A})}} \sum_{i=1}^{n} \left(\|\mathbf{L}\mathbf{s}^{(i)} - \mathbf{x}_i\|_2^2 + \mu \|\mathbf{s}^{(i)}\|_0 \right)$$

Step 2 can be solved efficiently via SVD:

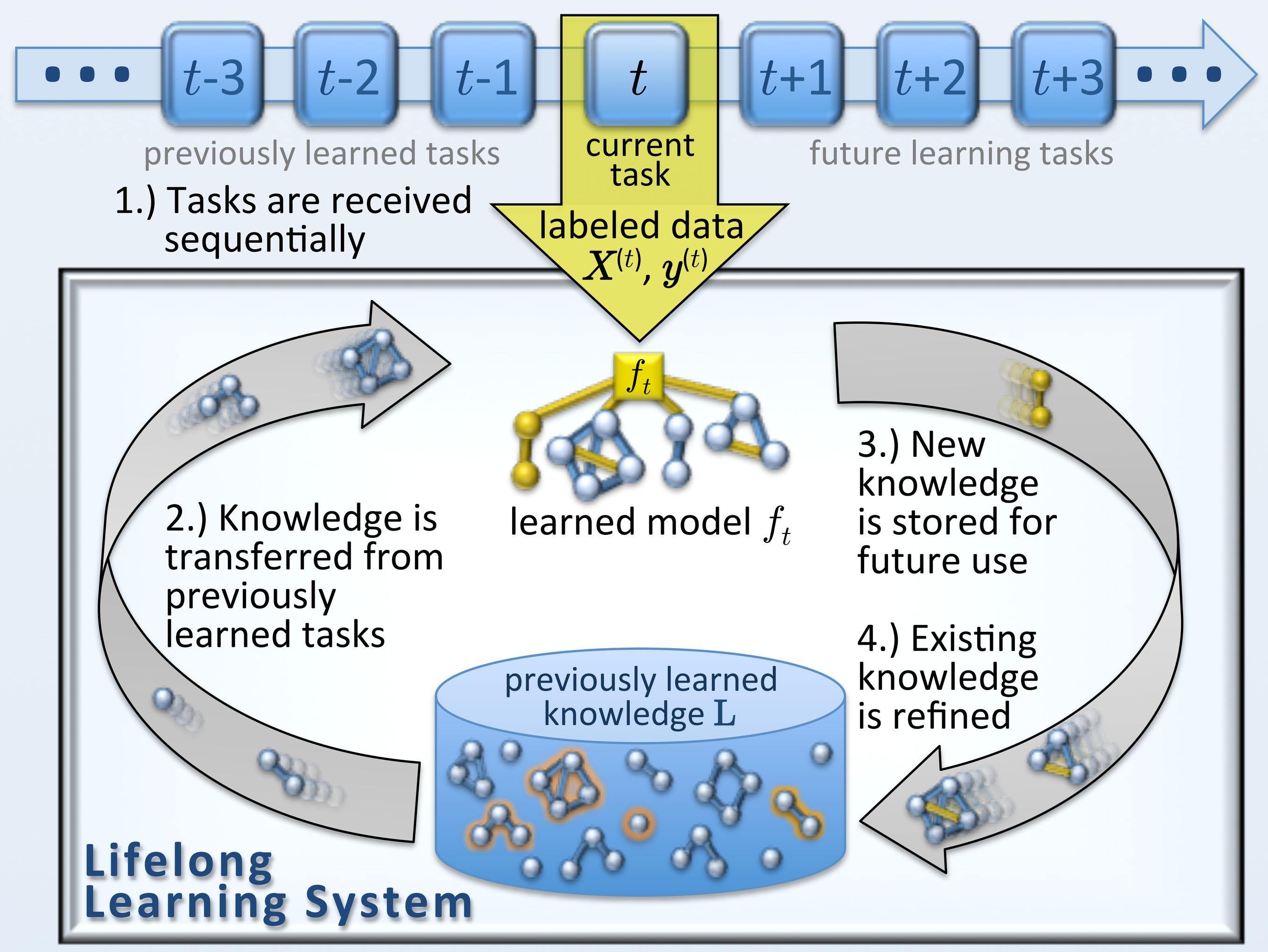
- Let the i[th] column of $\mathbf{E}$ be given by $\mathbf{e}_i = \mathbf{x}_i - \sum_{r \neq j} s_r^{(i)} \mathbf{l}_r$
- Then take

$$(\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}) = \mathrm{svd}(\mathbf{E}_{\mathcal{A}})$$

$$\mathbf{l}_j \leftarrow \mathbf{u}_1 \qquad \mathbf{s}_j^{(\mathcal{A})} \leftarrow \sigma_{1,1} \mathbf{v}_1$$

Surprisingly, we can efficiently find the global minimum!

Online Multi-Task Learning via K-SVD

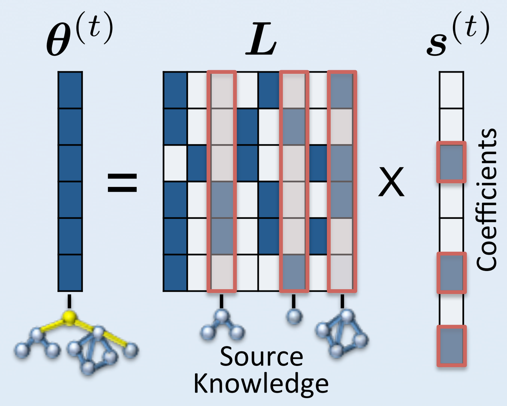


Assumes a parametric model for each task t

$$f^{(t)}(\mathbf{x}) = f(\mathbf{x}; \boldsymbol{\theta}^{(t)}) \qquad \boldsymbol{\theta}^{(t)} \in \mathbb{R}^d$$

The parameter vectors for each model are linear combinations of a shared latent basis $\mathbf{L}$

$$\boldsymbol{\theta}^{(t)} = \mathbf{L}\boldsymbol{s}^{(t)} \qquad \mathbf{L} \in \mathbb{R}^{d \times k},\ \boldsymbol{s}^{(t)} \in \mathbb{R}^k$$



The MTL objective function encourages transfer between models:

$$e_T(\mathbf{L}) = \frac{1}{T} \sum_{t=1}^{T} \min_{\boldsymbol{s}^{(t)}} \left\{ \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}\left(f\left(\boldsymbol{x}_i^{(t)}; \mathbf{L}\boldsymbol{s}^{(t)} \right), y_i^{(t)} \right) + \mu \|\boldsymbol{s}^{(t)}\|_1 \right\} + \lambda \|\mathbf{L}\|_{\mathsf{F}}^2$$

#tasks seen so far — model fit to data — sparsity — complexity

We can re-write this objective as a sparse coding problem [Ruvolo & Eaton, ICML '13]

$$g_T(\mathbf{L}) = \frac{1}{T} \sum_{t=1}^{T} \min_{\boldsymbol{s}^{(t)}} \left\{ \|\boldsymbol{\theta}^{(t)} - \mathbf{L}\boldsymbol{s}^{(t)}\|_{\mathbf{D}^{(t)}}^2 + \mu \|\boldsymbol{s}^{(t)}\|_1 \right\} + \lambda \|\mathbf{L}\|_{\mathsf{F}}^2$$

where: $\boldsymbol{\theta}^{(t)} = \arg\min_{\boldsymbol{\theta}} \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}\left(f(\boldsymbol{x}_i^{(t)}; \boldsymbol{\theta}), y_i^{(t)} \right)$

$\boldsymbol{D}^{(t)}$ is ½ the Hessian of the single-task loss evaluated at $\boldsymbol{\theta}^{(t)}$

$\|\mathbf{x}\|_{\mathbf{D}}^2 = \mathbf{x}^\top \mathbf{D} \mathbf{x}$

Using K-SVD for Multi-Task Learning

The sparse coding formulation of MTL is similar to the K-SVD objective.

Key Idea: Use SVD to efficiently solve the MTL objective

- Need to use the generalized SVD $(\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}) = \mathrm{gsvd}(\mathbf{E}_{\mathcal{A}}, \mathbf{M}, \mathbf{W})$ instead of the SVD to properly account for 2[nd] order information, where

$$\mathbf{M} = \frac{1}{|\mathcal{A}_j|} \sum_{t' \in \mathcal{A}_j} \mathbf{D}^{(t')} \qquad w_t = \frac{\mathbf{1}^\top \mathbf{D}^{(t)} \mathbf{1}}{\sum_{t' \in \mathcal{A}_j} \mathbf{1}^\top \mathbf{D}^{(t')} \mathbf{1}}$$

- $\mathbf{M}$ and $\mathbf{W}$ serve as feature and task relationship matrices

Modifications to Learn Tasks Online

- When training on task t, update only $s^{(t)}$ and the relevant basis vectors
- Perform each step of K-SVD only once per batch of training data

ELLA-SVD Algorithm

Given a new task t,

1. Train a single-task model $\boldsymbol{\theta}^{(t)}$ for task t
2. Reconstruct $\boldsymbol{\theta}^{(t)}$ in the current basis (LASSO):

$$\mathbf{s}^{(t)} \leftarrow \arg\min_{\mathbf{s}} \left\{ \|\mathbf{L}\mathbf{s} - \boldsymbol{\theta}^{(t)}\|_{\mathbf{D}^{(t)}}^2 + \mu \|\mathbf{s}\|_0 \right\}$$

3. Update the basis:

for $j = 1 \ldots k$ such that $s_j^{(t)} \neq 0$, solve via GSVD

$$\mathbf{l}_j, \mathbf{s}_j^{(\mathcal{A})} \leftarrow \arg\min_{\mathbf{l}_j, \mathbf{s}_j^{(\mathcal{A})}} \sum_{t=1}^{T} \left(w_t \|\mathbf{L}\mathbf{s}^{(t)} - \boldsymbol{\theta}^{(t)}\|_{\mathbf{M}}^2 + \mu \|\mathbf{s}^{(t)}\|_0 \right)$$

One pass per training set (no "loop until convergence")

Per-Task Computational Complexity

ELLA-SVD: O(base learner + $d^2k + k^2d + qd^3 + qr^2d$)

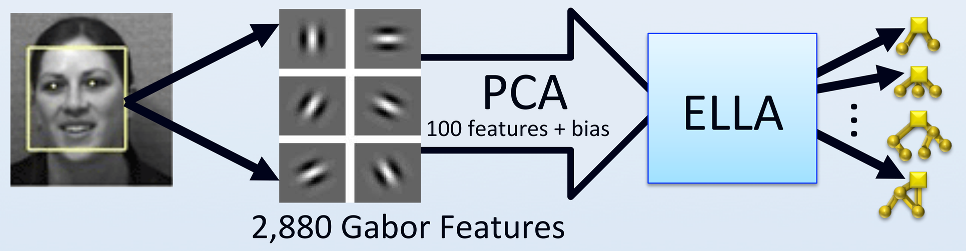
q = sparsity of $s^{(t)}$ $\quad r$ = # tasks utilizing same basis component

ELLA: O(base learner + d^3k^2) ← significantly less efficient than ELLA-SVD

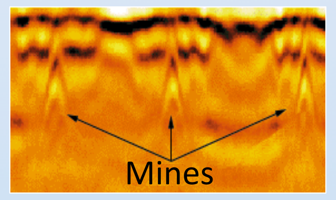
Applications

Facial Expression Recognition: identify presence of facial action units (#5 upper lid raiser, #10 upper lip raiser, #12 lip corner pull)

21 Classification Tasks:
- 7 subjects
- 450-999 images each

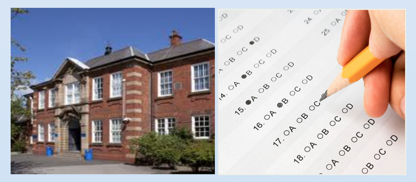


Land Mine Detection from radar



29 Classification Tasks:
- 29 regions
- 2 terrain types
- 14,820 instances total

Student Exam Score Prediction

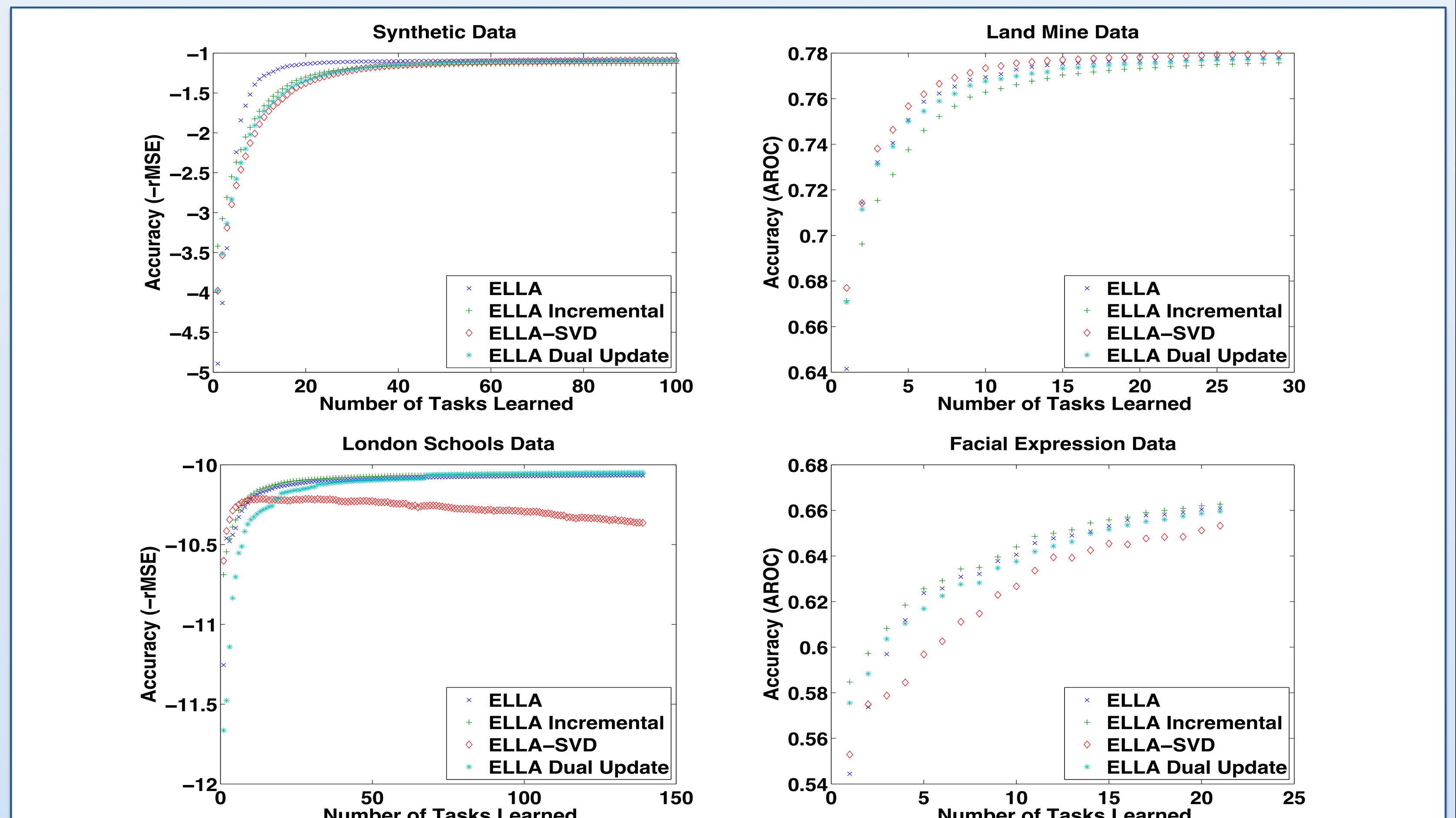


139 Regression Tasks:
- 139 schools
- 15,362 students total
- 4 school-specific features
- 3 student-specific features

Results

We compared ELLA-SVD to ELLA and two variants:

- ELLA Incremental – a more efficient but suboptimal version of ELLA
- ELLA Dual Update – a hybrid combination of ELLA-SVD & ELLA Incremental



Acknowledgement: This research was supported by ONR grant #N00014-11-1-0139

Sparse dictionary optimization provides a computationally efficient method for online multi-task learning