This CVPR2015 extended abstract is the Open Access version, provided by the Computer Vision Foundation.

Curriculum Learning of Multiple Tasks

Anastasia Pentina, Viktoriia Sharmanska, Christoph H. Lampert
Institute of Science and Technology Austria (IST Austria).

Multi-task algorithms are an alternative to traditional single-task machine learning algorithms, which can be used when several related tasks are needed to be solved. By sharing information between the tasks such algorithms are able to obtain solutions of reasonable quality from just few labeled training examples per task. Though multi-task algorithms differ by the type of information they transfer, their success highly depends on the assumption that such transfer takes place only between related tasks.

In this work we concentrate on the case of linear predictors and the assumption that weight vectors corresponding to similar tasks are close to each other with respect to Euclidean distance. This idea was introduced by Evgeniou and Pontil in [2], where they proposed an SVM-based algorithm with biased regularization that forces all the weight vectors to lie close to some common prototype. Therefore this algorithms treats all the tasks as equally related. However in a realistic scenario it might not be optimal as there might be some outlier tasks or group of tasks which are not related to others. In order to overcome this difficulty the algorithm from [2] was generalized by Evgeniou *et al.* in [3] by introducing a graph regularization. Alternatively, Chen *et al.* [1] proposed to penalize deviations in weight vectors between highly correlated tasks. However these approaches require prior information regarding the task relationships. In contrast, we propose an algorithm that does not require all tasks to be related and does not need any additional knowledge about their similarities.

Formally, we assume that the learner observes n tasks, $t_1,\dots,t_n$, which share the same input space $\mathcal{X}\subset\mathbb{R}^d$ and output space $\mathcal{Y}=\{-1,1\}$. Every task t_i is represented by a set of m_i training examples $S_i=\{(x_1^i,y_1^i),\dots,(x_{m_i}^i,y_{m_i}^i)\}$ sampled i.i.d. according to some unknown data distribution D_i. The goal of the learner is to find n weight vectors $w_1,\dots,w_n$, such that the average expected error rate of the corresponding linear classifiers is minimized:

$$\mathrm{er}(w_1,\dots,w_n)=\frac{1}{n}\sum_{i=1}^{n}\mathop{\mathbf{E}}_{(x,y)\sim D_i}[\![y\neq\mathrm{sign}\langle w_i,x\rangle]\!]. \tag{1}$$

We propose to solve this problem by processing tasks sequentially instead of jointly and transferring information between subsequent tasks. Specifically, if the tasks are processed in order $\pi\in\mathfrak{S}$, where $\mathfrak{S}$ is the set of all permutations over n elements, a task $t_{\pi(i)}$ serves a source of additional information for the next task $t_{\pi(i+1)}$ for all $i=1,\dots,n-1$. In other words, we propose to decompose a multi-task problem into a sequence of domain adaptation problems and any known algorithm can be used to solve a task based on the previous one. This approach makes learning more tolerant to possible variations among the tasks because it does not need all of them to be related. However its success may depend on the order in which tasks are processed, since information transfer between subsequent tasks is assumed to be beneficial. In order to better understand the role of the task order we analyze it using statistical learning theory.

To perform our theoretical analysis, we assume that every task is solved using some fixed, deterministic algorithm $\mathcal{A}$. This algorithm returns a weight vector for a task $t_{\pi(i)}$ based on the corresponding training data $S_{\pi(i)}$ and the weight vector $w_{\pi(i-1)}$ obtained for the previous task. Under these assumptions we can formulate the following result:

Theorem 1. *For any deterministic learning algorithm $\mathcal{A}$ and any $\delta>0$, the following inequality holds with probability at least $1-\delta$ (over sampling the training sets $S_1,\dots,S_n$) uniformly for any order $\pi\in\mathcal{S}_n$:*

$$\frac{1}{2n}\sum_{i=1}^{n}\mathop{\mathbf{E}}_{(x,y)\sim D_i}[\![y\neq\mathrm{sign}\langle w_i,x\rangle]\!]\leq \frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{m_{\pi(i)}}\sum_{j=1}^{m_{\pi(i)}}\bar{\Phi}\left(\frac{y_j^{\pi(i)}\langle w_{\pi(i)},x_j^{\pi(i)}\rangle}{||x_j^{\pi(i)}||}\right)+\frac{||w_{\pi(i)}-w_{\pi(i-1)}||^2}{2\sqrt{\bar{m}}}\right]+\frac{1}{8\sqrt{\bar{m}}}-\frac{\log\delta}{n\sqrt{\bar{m}}}+\frac{\log n}{\sqrt{\bar{m}}}, \tag{2}$$

where $\bar{m}$ is the harmonic mean of the sample sizes $m_1,\dots,m_n$, $\bar{\Phi}(z)=\frac{1}{2}\left(1-\mathrm{erf}\left(\frac{z}{\sqrt{2}}\right)\right)$, $\mathrm{erf}(z)$ is the Gauss error function, $\pi(0)=0$, $w_0=\mathbf{0}$ and $w_{\pi(i)}=\mathcal{A}(w_{\pi(i-1)},S_{\pi(i)})$.

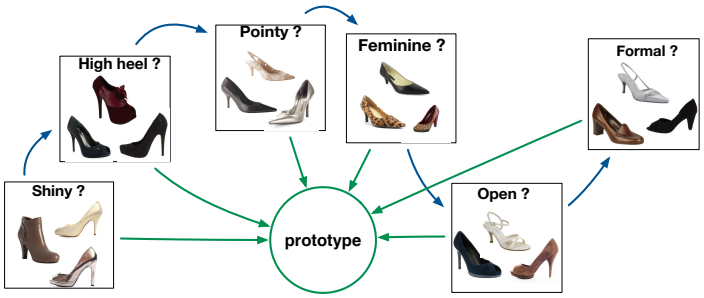

Figure 1: Schematic illustration of the proposed multi-task learning approach. If each task is related to some other task but not equally much to all others, learning tasks in a sequence (blue arrows) can be beneficial to classical multi-task learning based on sharing information from a single prototype (green arrows).

The left hand side of (2) is one half of the average expected error (1). This quantity the learner would like to minimize, but it is not directly computable because the data distributions D_i are unknown. In contrast the right hand side of (2) consists only of computable quantities and its low value ensures low expected error. Therefore it can be seen as a quality measure of an order π and by minimizing it with respect to π one can obtain a task order that is adjusted to the set of tasks $t_1,\dots,t_n$. Because (2) holds uniformly for all π, the guarantees given by Theorem 1 will also hold for the obtained order. We propose an algorithm for choosing a beneficial task order that is based on greedy optimization of (2).

Our experimental results on two real-world datasets show that learning tasks sequentially can be more effective than learning them jointly they also show that task order can significantly influence the performance of the algorithm and that our method is able to automatically choose a favorable task order based only on the training data.

[1] Xi Chen, Seyoung Kim, Qihang Lin, Jaime G. Carbonell, and Eric P. Xing. Graph-structured multi-task regression and an efficient optimization method for general fused lasso. arXiv:1005.3579 [stat.ML], 2010.

[2] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2004.

[3] Theodoros Evgeniou, Charles A. Micchelli, and Massimiliano Pontil. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research (JMLR)*, 6, 2005.

This is an extended abstract. The full paper is available at the Computer Vision Foundation webpage.