

Multi-Task Learning(MTL)

May 15, 2024

1 Introduction

When the original data representation is high dimensional and the number of examples provided to solve a regression or classification problem is limited, any learning algorithm which does not use any sort of prior knowledge will perform poorly due to lack of data to reliably estimate the model parameters.

If several related tasks are available, we can combine these tasks to boost performance on each single task. The philosophy is what is learned for each task can help other tasks be learned better. Assume T tasks each task t has dataset $Z_t = \{(X_{t,i}, Y_{t,i})\}_{i=1}^{n_t}$, where $X_{t,i}, Y_{t,i} \sim P^t$ which has a common sample space $\mathcal{X} \times \mathcal{Y} \subset \mathbb{R}^d \times \mathbb{R}$. MTL can be divided into several categories:

- Feature based: aims to learn common features among different tasks
- Instance based: share identified useful data
- Parameter based: share some common model parameters

2 Feature Based MTL

The core idea here is if all T tasks are related, there may exist some hidden feature space \mathcal{H} that first find a common map $h : \mathcal{X} \rightarrow \mathcal{H}$ for all T tasks and $Y_{t,i}|h(X_{t,i})$ may be easy to calculate.

A simple setting is by Argyriou[1], assume we need to estimate

$$EY_t|X_t = f_t(X_t) = a_t^T U X_t, \quad U \in \mathcal{O}(\mathbb{R}^{d \times d})$$

which can be understood as a orthogonal projection $U \in \mathbb{R}^{d \times d}$ map X_t to a hidden space \mathcal{H} , and for different task t the target is some linear combination of the hidden feature with parameter $a_t \in \mathbb{R}^d$.

Let $A = (a_1^T, \dots, a_T^T) \in \mathbb{R}^{d \times T}$, a regularized error function with loss L can be

$$Err(A, U) = \sum_{t=1}^T \sum_{i=1}^{n_t} L(Y_{t,i}, a_t^T U X_{t,i}) + \gamma \|A\|_2^2.$$

References

- [1] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine learning*, 73:243–272, 2008.