Learning to Transfer with Triply Adversarial Nets

Gilles Louppe
New York University
g.louppe@nyu.edu

Abstract

In classification, transfer learning (or its variants known as co-variate shift or domain adaptation) arises whenever target instances are governed by a distribution that may be arbitrarily different from the distribution of the source instances used at training. This problem has traditionally been solved by re-weighting approaches or by learning robust representations over domains. In this work, we propose a new paradigm based on the assumption that the co-variate shift is only due to a different representation of the same underlying objects. Accordingly, we propose to learn how to transform source instances into target instances, possibly across input spaces of distinct dimensions, structures or supports. For this purpose, we extend the generative adversarial networks framework of [1] to a triply adversarial process: a transformer network G for generating target instances from source instances, a discriminative network D for separating transformed source instances from actual target instances, and a classifier network $C \circ G$ for classifying source instances in the projected space. This 3-player game results in a network G capable of transforming source into target instances, while preserving separation between classes as enabled by C in the adversarial setup. Preliminary experiments demonstrate the potential of this novel approach, with promising results when the construction of C can be bootstrapped in a semi-supervised way from a few labeled instances from the target space.

1 Introduction

[GL: to write]

2 Problem statement

Let assume a probability space $(\Omega, \mathcal{F}, P)$, where Ω is a sample space, $\mathcal{F}$ is a set of events and P is a probability measure. Let consider a (multivariate) random variable $X : \Omega \mapsto \mathbb{R}^p$ inducing the source distribution p_X, along with a finite set $\{x_i\}_{i=1}^N$ of its realizations $X(\omega_i)$, for $\omega_i \in \Omega$. In classification, let us further assume that realizations from the source distribution are extended with realizations $\{y_i\}_{i=1}^N$ of a label random variable $Y : \Omega \mapsto \mathcal{Y}$, thereby inducing the joint source distribution $p_{X,Y}$. Similarly, let assume a (multivariate) random variable $U : \Omega \mapsto \mathbb{R}^q$ inducing the target distribution p_U, along with a finite set $\{u_j\}_{j=1}^M$ of its realizations $U(\omega_j)$, for $\omega_j \in \Omega$. Assuming it exists, our goal is to find a transfer function $T : \mathbb{R}^p \times \mathcal{Y} \mapsto \mathbb{R}^q$ such that

$$T(X(\omega), Y(\omega)) = U(\omega) \text{ for all } \omega \in \Omega. \tag{1}$$

Since we do not have training tuples $((X(\omega), Y(\omega)), U(\omega))$ (for the same unknown ω) from which T could be learned using standard regression algorithms, we propose instead to solve the closely related problem of finding a transfer function $\hat{T}$ such that

$$P(\{\omega | \hat{T}(X(\omega), Y(\omega)) = u\}) = P(\{\omega' | U(\omega') = u\}) \text{ for all } u \in \mathbb{R}^q. \tag{2}$$

In words, we are looking for a transfer function $\hat{T}$ such that realizations of $\hat{T}(X, Y)$ are indistinguishable from realizations of U. A function T for which Eqn. 1 is true necessarily satisfies Eqn. 2. The converse is however in general not true, since the sets of events $\{\omega | \hat{T}(X(\omega), Y(\omega)) = u\}$ and $\{\omega' | U(\omega') = u\}$ do not need to be the same for the equality to hold. Accordingly, the contribution of this work is to propose a procedure for constructing transfer functions satisfying Eqn. 2, and for which Eqn. 1 is *plausibly* satisfied.

In these terms, our framework encompasses the problems of transfer learning, domain adaptation or co-variate shift, where the labeled training samples correspond to the labeled realizations of the source distribution and where the unlabeled test samples correspond to realizations of the target distribution. At its core, this framework assumes that the underlying objects ω share a same universe Ω and that the source and target distributions only differ in the way they represent these objects. While not verified in all cases, we believe this assumption to be met in many practical situations, e.g., when learning to transfer across natural images from distinct datasets but representing the same concepts or when learning to adapt to a change of a measurement apparatus (but not of the underlying objects to be observed).

[GL: What are the necessary conditions on $p_{X,Y}$ and p_U for the existence of T? and of $\hat{T}$? I think results can be proven using information theory.]

3 Method

Generative adversarial networks (GAN) were first proposed by [1] as a way to generate samples from random noise $z \sim p_Z$. In this work, the authors pit a generative model G against an adversary classifier D whose repelling objective is to recognize real from fake samples. Both models G and D are trained simultaneously, in such a way that G learns to produce samples that are difficult to classify by D, while D incrementally adapts to changes in G. At the equilibrium, G models a distribution whose fake samples are recognized by D only by chance. In other words, assuming enough capacity in D, the distribution $p_{G(Z)}$ of fake samples converges towards the distribution of real samples.

In this work, we extend the GAN framework by first noticing that if we replace the noise distribution p_Z by the joint source distribution $p_{X,Y}$ and consider the real data distribution as the target distribution p_U, then the generative model G is a transfer function $\hat{T} : \mathbb{R}^p \times \mathcal{Y} \mapsto \mathbb{R}^q$ satisfying Eqn. 2. That is, generative adversarial networks provide a direct way for learning to transfer.

Theorem 1. *At the equilibrium, G satisfies Eqn. 2. [GL: This seems a direct consequence of the convergence results of [1]. Needs to be double-checked.]*

As noted earlier, transfer functions $\hat{T}$ satisfying Eqn. 2 are not unique. It is therefore critical to guide the adversarial training process in order to obtain a plausible transfer function. Of special interest in the context of classification are functions $\hat{T}$ preserving the conditional class distribution in the target space, i.e. such that

$$p_{Y|X}(y|x) = p_{Y|\hat{T}(X,Y)}(y|\hat{T}(x, y)) \text{ for all } x, y \in \mathbb{R}^p \times \mathcal{Y}. \tag{3}$$

In words, we are looking for a transfer function $\hat{T}$ such that realizations of $\hat{T}(X, Y)$ are indistinguishable from realizations of U, while preserving the class distribution and separability in the target space. In order to satisfy this constraint, we extend the GAN framework by adding a third network in the pit: a classifier C whose objective is to classify labeled source samples in the projected space. Accordingly, we extend G's objective function so as to maximize C's accuracy, while still fooling D. Formally, assuming binary classification (i.e. $\mathcal{Y} = \{0, 1\}$), the three networks D, C and G are concurrently trained so as to minimize the respective loss functions:

$$L(D) = -\mathbb{E}_{u \sim p_U}[\log(1 - D(u))] - \mathbb{E}_{x,y \sim p_{X,Y}}[\log(D(G(x, y)))] \tag{4}$$

$$L(C) = -\mathbb{E}_{x \sim p_{X|Y=0}}[\log(1 - C(G(x, y = 0)))] - \mathbb{E}_{x \sim p_{X|Y=1}}[\log(C(G(x, y = 1)))] \tag{5}$$

$$L(G) = \mathbb{E}_{x,y \sim p_{X,Y}}[\log(D(G(x, y)))] + L(C) \tag{6}$$

This 3-player game results in a network G capable of transforming source into target instances, while preserving separation between classes as enabled by C in the adversarial setup. Accordingly, when the transformed source distribution equals the target distribution, C converges towards a classifier that can be used for classifying samples from the target space, thereby providing a solution to transfer learning for classification.

[GL: Add figure]

Despite preserving class proportions and separability in the target space, the proposed framework does not guarantee that the learned classifier C assigns the correct labels $Y(\omega)$ to realizations $U(\omega)$. For instance, when $P(Y=0)=P(Y=1)$, the loss function $L(C)$ is equally minimized when the correct class is always perfectly predicted, i.e. when $C(G(x,y=0))=0$ and $C(G(x,y=1))=1$ for all x, than when the incorrect class is always wrongly predicted, i.e. when $C(G(x,y=0))=1$ and $C(G(x,y=1))=0$ for all x. Fortunately, this problem can be mitigated in the common case of semi-supervised learning, i.e. when at least a few pairs of *seed* realizations $U(\omega), Y(\omega)$ are known in the target space. Indeed, these seed samples can be used as additional training data when learning C, thereby indirectly constraining the learning of G towards a transfer function for which source instances projected close to the seed samples should share the same label, hence yielding an even more plausible transfer function. Formally, the loss function $L(C)$ is replaced with

$$\begin{aligned}L(C) = \gamma(&-\mathbb{E}_{x\sim p_{X|Y=0}}[\log(1-C(G(x,y=0)))] - \mathbb{E}_{x\sim p_{X|Y=1}}[\log(C(G(x,y=1)))])+\\ &(1-\gamma)(-\mathbb{E}_{u\sim p_{U|Y=0}}[\log(1-C(u)] - \mathbb{E}_{x\sim p_{U|Y=1}}[\log(C(u))])\end{aligned} \tag{7}$$

where γ is a parameter trading off the weight of incorrect predictions for seed samples and where expectations $\mathbb{E}_{u\sim p_{U|Y}}$ are approximated empirically over the known realizations in the target space.

4 Experiments

[GL: To do. See notebooks for a working proof of concept.] [GL: Comment on network architecture depending on the kind of transfer.]

5 Related work

[GL: to write]

6 Conclusions

[GL: to write]

Acknowledgments

[GL: todo]

References

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.