Learning to pivot

Machine Learning in High Energy Physics

Maxim Borisyak

National Research University Higher School of Economics

July 5, 2019

Learning to pivot

Pivoted classifier[1] f is a classifier which output does not depend on nuisance parameters Z:

$$\forall s : \forall z, z' : P(f(X) = s \mid Z = z) = P(f(X) = s \mid Z = z')$$

Examples:

- legal reasons;
- differences between simulation and real data;
- unobservable nuisance parameters.

[1] One can also consider a regressor or any other method that is based on likelihood maximization. This presentation is based on `https://arxiv.org/abs/1611.01046`

Ideally, classifier should be regularized:

$$\mathcal{L}_{\text{pivot}}(f) = \mathcal{L}(f) + \text{dependency-measure}(f, Z) \to \min$$

How dependencies can be measured?

Adversarial training

One way to measure dependency is as predictability of nuisance given output of f:

$$\begin{aligned} \text{dependency-measure}(f, Z) &= -\min_{r} \mathcal{L}_{\text{adv}}(r, f(X), Z); \\ \mathcal{L}_{\text{adv}}(r, f(X), Z) &= \mathop{\mathbb{E}}_{X,Z} \log P_r(Z \mid f(X)). \end{aligned}$$

Learning to pivot with adversarial networks

The final loss function:

$$L_{\text{pivoted}} = -\frac{1}{N}\sum_{i=1}^{N} \log P_f(y_i \mid x_i) + \frac{1}{N}\sum_{i=1}^{N} \log P_r(z_i \mid f(x_i)) \to \min$$

The training procedure is similar to GAN:

- train adversary r;
- make one step for classifier f with fixed r.

Conditional pivoting

Sometimes it is desirable to make a classifier independent from nuisances within each class.

$$\forall y : \forall s : \forall z, z' : P(f(X) = s \mid Z = z, Y = y) = P(f(X) = s \mid Z = z', Y = y)$$

In this case adversary should be parameterized by target, i.e. receive y as an additional input.

Summary

Summary

- dependencies can be measured as quality of a trained model;
- it is possible to pivot model with an adversary;
- conditional pivoting is done by adding target as input feature to the adversary.

Literature

- Louppe G, Kagan M, Cranmer K. Learning to pivot with adversarial networks. InAdvances in neural information processing systems 2017 (pp. 981-990).