Towards Accurate Model Selection in Deep Unsupervised Domain Adaptation

Kaichao You, Ximei Wang, Mingsheng Long, Michael I. Jordan

**Validation in UDA: the problem**
- Supervised Learning
  - train/validation/test data come from the same distribution
  \[
  (x_1, y_1) \sim p \quad (x_2, y_2) \sim p \quad (x_3, y_3) \sim p
  \]
  - 
- Unsupervised Domain Adaptation (UDA)
  - train/test data come from different distributions
  - test data is unlabeled until the test phase, so target labels are not available for validation
  \[
  (x_1, y_1) \sim p \quad (x_2, y_2) \sim q
  \]
  - Status quo of model selection in UDA
    - Source Risk: a highly biased estimator of the underlying target risk in UDA
    - Target Risk: requires target labels that contradicts with the assumption of UDA
    - IWCV unstable because of the unbounded variance
    - TrCV: requires target labels that contradicts with the assumption of UDA

**Insights**
- Domain adaptation reduces distribution discrepancy, thus lowering the variance upper-bound
- Use a control variate to explicitly reduce the variance
- Density ratio can be estimated discriminatively

**Embed Adapted Features into Model Selection**
- Recent feature adaptation methods reduce distribution discrepancy
  \[
  d_w(q(x)) \leq d_s(q(x))
  \]
  - TrCV: requires target labels that contradicts with the assumption of UDA

**Control Variate**
- \[ E[z] = \tau \]
- \[ E[x^*] = E[x] + \tau (t - \tau) \]
- \[ \Var[x^*] = \Var[x] + \tau^2 \Var[t - \tau] = \tau(\tau\Var[t] - \Var[t]) = \tau^2 \]
- \[ \min \Var[x^*] = \left( 1 - \tau^2 \right) \Var[x] \text{ when } \bar{\eta} = \frac{\Var[t]}{\Var[t] - \Var[t]} \]
- \[ w_\bar{\eta} \text{ can be used as a control variate} \]
  \[
  E_{x \sim p}[w_\bar{\eta}(x)] = E_{x \sim p}[\frac{\eta}{\Var[x]} \Var[x/(x|\bar{\eta})}] = 1
  \]

**Discriminative Density Ratio Estimation**
- Can be estimated by a discriminative model to distinguish source examples from target examples
  \[
  w(x) = \frac{q(x)}{p(x)} = \frac{J_\tau(d = 0)}{J_\tau(d = 1)} \quad \frac{J_\tau(d = 1)}{J_\tau(d = 0)} \quad \frac{J_\tau(d = 0)}{J_\tau(d = 1)}
  \]
  - \[ \lambda = 0 \]
  - \[ \lambda = 1 \]

**Algorithm in Detail**

**Algorithm 1 GetRisk**

**Input:** Candidate model \( g(x) = T(F(x)) \)
- Training set: \( D_s = \{(x_j^s, y_j^s)\}_{j=1}^m \)
- Validation set: \( D_v = \{(x_j^v, y_j^v)\}_{j=1}^n \)
- \( D_s \) is partitioned into \( D_{sv} \) and \( D_{tv} \)

**Output:** DEV Risk \( R_{DEV}(g) \) of model \( g \)
- Compute features and predictions using model \( g \):
  \[ F_{\hat{\eta}} = \{ f_{\hat{\eta}}(x) \}_{j=1}^m \]
  \[ F_{\hat{\tau}} = \{ f_{\hat{\tau}}(x) \}_{j=1}^m \]

- Train a two-layer logistic regression model to classify \( F_{\hat{\eta}} \) and \( F_{\hat{\tau}} \) (label \( F_{\hat{\eta}} \) as 1 and \( F_{\hat{\tau}} \) as 0)

- Compute weighted loss \[ L = \sum_{j=1}^m \left[ w_j f_{\hat{\eta}}(x_j) \hat{\tau}(\hat{\eta}^j, \hat{\tau}^j) \right] \]

- Estimate coefficient \( \eta \)

- Compute DEV Risk:
  \[ R_{DEV}(g) = \frac{\sum_{j=1}^m L(x_j, \hat{\eta}, \hat{\tau})}{\Var[W]} \]

**Experimental Results**

**Experiments on a toy problem under covariate shift**

**TrCV**
- Source Risk
- Target Risk
- IWCV
- Dev (Proposed)

**ICML**
- Covariate Shift Assumption
- Problems in IWCV:
  - Unbiased but the variance is unbounded
  \[ \Var[z_{uv}] \leq \Var[z_{uv}] \quad \Var[y_1, y_2] \quad \Var(y) \]
  - Density ratio is not readily accessible
  Fitting a gaussian distribution as in the original paper is not reasonable.

**Algorithm 2 Deep Embedded Validation (DEV)**

**Input:** A set of candidate models \( G_m = \{g_i(x)\}_{m=1}^M \)

**Output:** The best model \( (G_{\hat{m}}(x)) \)

- Get DEV Risks of all models \( R = \{R_{DEV}(g_i)\}_{i=1}^m \)

- Rank the best model \( i = \arg\min_{1 \leq \hat{m} \leq M} R_{\hat{m}} \)