Infinite Task Learning with Vector-Valued RKHSs

Parameterized Learning Tasks

Some learning tasks involve loss functions that depend on a hyperparameter. Examples: two parameterized tasks Quantile Regression (QR) and Cost-Sensitive Classification (CSC) with the following notations.

- \( X \) input data space \( (\mathbb{R}^d) \), \( \Theta \) parameter space \( (\mathbb{R}) \), \( Y \) output space \( (\mathbb{R}) \).
- Hypothesis space \( \mathcal{H} \subset \mathcal{F}(X;Y) \).
- Parameterized cost \( v: \Theta \times X \times Y \rightarrow \mathbb{R} \).

**QR:** Given \( X,Y \in \mathcal{X} \times \mathcal{Y} \) random variables, estimate the \( \theta \)-quantile of the conditional distribution \( P_{XY} \):

\[
q(x|\theta) = \inf\left\{ t \in \mathcal{Y} : P_{Y|X=x}(Y \leq t) \geq \theta \right\}, \quad \forall (x,\theta) \in \mathcal{X} \times (0,1). \tag{1}
\]

Pinball loss:

\[
v(\theta,y,h(x)) = |\theta - 1_{\mathcal{Y}}(y-h(x))| \quad \forall (\theta,y,h(x)) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{H}. \tag{2}
\]

**CSC:** Support Vector Machine with asymmetric loss function

\[
v(\theta,y,h(x)) = |\theta - 1_{\mathcal{Y}}(y)| + |y-h(x)|. \tag{3}
\]

Usual approach: Empirical risk minimization in some well chosen \( \mathcal{H} \) for a given value of \( \theta \). For several values, turn to Multi-Task learning [1].

**Learning an infinite number of tasks**

We propose to jointly solve parameterized tasks for an infinite number of values of the hyperparameter \( \theta \) using function-valued regression:

- Hypothesis space \( \mathcal{H} \subset \mathcal{F}(X;\mathcal{F}(\Theta;Y)) \) i.e \( h(x) \in \mathcal{F}(\Theta;Y) \).
- Local loss \( V(y,h(x)) := \int_{\Theta} v(\theta,y,h(x)|\theta) d\mu(\theta) \).

Minimizing population risk:

\[
\arg\min_{h \in \mathcal{H}} R(h) := E_{XY}[V(Y,h(X))]. \tag{4}
\]

**Proposition.** \( q \) defined in (1) minimizes (4) for the pinball loss (2).

\( \Rightarrow \) Extension of Multi-Task Learning to an infinite number of tasks [2].

**Sampled Empirical Risk**

Approximate expectation over \( P_{XY} \) and \( f_\Theta \):

\[
\left\{ (x_i,y_i) \right\}_{i=1}^n \text{ i.i.d } \sim P_{XY}
\]

\[
\left\{ (\Theta_j) \right\}_{j=1}^m \sim \mu \text{ (Quasi-Monte Carlo)}
\]

Sampled empirical risk:

\[
\hat{R}_n(h) := \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m v(\theta_j, y_i, h(x_i)|\Theta_j)). \tag{5}
\]

Regularized problem:

\[
\arg\min_{h \in \mathcal{H}} \hat{R}_n(h) + \lambda \Omega(h). \tag{5}
\]

**Vector-Valued RKHSs**

Natural extension of RKHSs for modelizing outputs in any Hilbert space.

- \( k_X: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R} \) and \( k_\Theta: \Theta \times \Theta \rightarrow \mathbb{R} \) two scalar valued kernels.
- Operator-valued kernel \( K(x,z) = k_X(x,z)k_\Theta \) associated to \( \mathcal{H}_K \) a space of function-valued functions.
- \( \mathcal{H}_K = \mathcal{S}pace \{ K(.,x)| x \in \mathcal{X}, f \in \mathcal{H}_K \} = \mathcal{H}_K \otimes \mathcal{H}_K \) and \( \mathcal{H}_K \).
- Hilbert norm \( \|h\|_\mathcal{H}_K \).

**Optimization**

**Proposition (Representer).** If \( \forall \theta \in \Theta, v(\theta, \cdot, \cdot) \) is proper lower semicontinuous with respect to its second argument, (5) has a unique solution \( h^* \in \mathcal{H}_K \) such that \( v(\varphi, \Theta) \in \mathcal{X} \times \Theta \).

- Solution shaped by \( k_X \) and \( k_\Theta \) (gaussian, laplacian, ...)
- Infinite dimensional problem \( \Rightarrow \) size \( n,m \)
- In practice, solved via smoothing \( v \) + L-BFGS.

**Excess Risk Guarantees**

Framework of vv-RKHS allows for proper analysis [3], tradeoff \( n/m \)

\[
R(h^*) \leq \hat{R}_n(h^*) + \frac{1}{\sqrt{n}} + O\left( \frac{\log(m)}{m} \right).
\]

**Numerical Experiments**

**QR:** Continuous model \( \Rightarrow \) new non-crossing constraint:

\[
\hat{\Omega}_n(h) := \lambda_n \sum_{i=1}^n \sum_{j=1}^m |\partial h(\theta_j)|.
\]

Left plot: strong non-crossing penalty \( (\lambda_n = 2) \). Right plot: no non-crossing penalty \( (\lambda_n = 0) \). The plots show 100 quantiles of the continuum learned, linearly spaced between 0 (blue) and 1 (red).

\( \Rightarrow \) Matches state of the art on 20 UCI datasets

**CSC:** Improved performances:

\[ \text{Iris dataset. } \text{Top: infinite learning; bottom: independent learning for } \theta \in [-0.9,0.9]. \]

**Code available:** https://bitbucket.org/RomainBrault/itl/