Reducing training time by efficient localized kernel regression

Nicole Mücke

nicole.muecke@mathematik.uni-stuttgart.de

University of Stuttgart

Abstract

We study generalization properties of kernel regularized least squares regression based on a partitioning approach. We show that optimal rates of convergence are preserved if the number of local sets grows sufficiently slowly with the sample size. Moreover, the partitioning approach can be efficiently combined with local Nyström subsampling, improving computational cost twofold.

Learning Setting

- minimize expected risk

$$\mathcal{E}(f) = \int_{\mathcal{X}\times\mathbb{R}} (f(x)-y)^2 d\rho(x,y)$$

  over reproducing kernel Hilbert space $\mathcal{H}$ (RKHS) with bounded kernel K

- note: minimizer over all measurable $f:\mathcal{X}\longrightarrow\mathbb{R}$ is regression function

$$f_\rho = \mathbb{E}[Y|X] \in L^2(\mathcal{X},\rho_X)$$

The Partitioning Approach

- $\{\mathcal{X}_1,...,\mathcal{X}_m\}$ partition of $\mathcal{X}$
- on $\mathcal{X}_j$ define local reproducing kernel K_j with RKHS $\mathcal{H}_j$
- weighted global kernel: $K(x,x') = \sum_{j=1}^m p_j\, K_j(x,x')$
- global RKHS is direct sum: $\mathcal{H} := \bigoplus_{j=1}^m \hat{\mathcal{H}}_j$

Defining the Estimator

- training data $\mathcal{D} = \{(x_1,y_1),...,(x_n,y_n)\}$ are split according to partition $\mathcal{D}_1,...,\mathcal{D}_m$
- on each set $\mathcal{X}_j$ compute a local estimator using a local kernel by solving

$$\hat{f}_j^\lambda = \operatorname*{Arg\,Min}_{f\in\mathcal{H}_j} \frac{1}{|D_j|}\sum_{(x,y)\in D_j}(f(x)-y)^2 + \lambda\,||f||^2_{\mathcal{H}_j}$$

- global estimator $\hat{f}^\lambda := \sum_{j=1}^m \hat{f}_j^\lambda \in \mathcal{H}$

Assumptions

Under which conditions is $\hat{f}^\lambda$ minimax optimal ?
The local covariances are

$$T_j = \mathbb{E}[K_j(X,\cdot)\otimes K_j(X,\cdot)]\ ,$$

giving the global one $T = \bigoplus_{j=1}^m T_j$.

- **Smoothness:** $||T^{-r}f_\rho||_{\mathcal{H}} < \infty, \quad 0<r\leq 1/2$.
- **Goodness of Partition:** For $0<\gamma\leq 1$ assume

$$Trace\big[(T+\lambda)^{-1}T\big] \lesssim \lambda^{-\gamma}\ .$$

Localization allows optimality:

Let $|\mathcal{D}_j| = \lfloor \frac{n}{m} \rfloor$. Then, with the choices

$$\lambda_n \simeq \left(\frac{1}{n}\right)^{\frac{1}{2r+1+\gamma}}$$

$$m_n \lesssim n^\alpha\ , \quad \alpha \leq \frac{2r}{2r+1+\gamma}$$

the excess risk satisfies

$$\mathbb{E}\Big[\ \mathcal{E}(\hat{f}^\lambda_{\mathcal{D}}) - \mathcal{E}(f_\rho)\ \Big] \lesssim \left(\frac{1}{n}\right)^{\frac{2r+1}{2r+1+\gamma}}.$$

Nyström Subsampling

Plain Nyström: Sample uniformly at random $l \leq n$ points $\tilde{x}_1,...,\tilde{x}_l$ from training data and seek for an estimator in a reduced space

$$\mathcal{H}_l = \{\ f\ :\ f = \sum_{j=1}^l \alpha_j K(\tilde{x}_j,\cdot)\ ,\ \alpha\in\mathbb{R}^l\ \}$$

by solving

$$\min_{f\in\mathcal{H}_l} \frac{1}{n}\sum_{j=1}^n (f(x_j)-y_j)^2 + \lambda||f||^2_{\mathcal{H}_l}\ .$$

Aim: Apply Nyström locally!

Combining Localization and Subsampling

If the number l of subsampled points on each local set satisfies

$$l_n \sim n^\beta\ , \quad \beta \geq \frac{1+\gamma}{2r+1+\gamma}$$

and if the number of local sets satisfies

$$m_n \lesssim n^\alpha\ , \quad \alpha \leq \frac{2r}{2r+1+\gamma}\ ,$$

then the choice for the regularization parameter λ_n given above guarantees the error bound

$$\mathbb{E}\Big[\mathcal{E}(\hat{f}^{\lambda_n}_{\mathcal{D}}) - \mathcal{E}(f_\rho)\Big] \lesssim \left(\frac{1}{n}\right)^{\frac{2r+1}{2r+1+\gamma}}.$$

This bound is known to be **minimax optimal!** (Caponnetto and E. De Vito 2006, Blanchard and M. 2017)

Computational Cost

1. KRLS	$\mathcal{O}(n^3)$
2. localized KRLS	$\mathcal{O}\big((\frac{n}{m})^3\big),\ 1\leq m\leq n^\alpha$
3. Nyström	$\mathcal{O}(nl^2+l^3),\ n^\beta\leq l\leq n$
4. local Nys.	$\mathcal{O}(\frac{n}{m}l^2+l^3),\ n^\beta\leq l\leq \frac{n}{m}$
5. distributed KRLS	$\mathcal{O}\big((\frac{n}{m})^3\big),\ 1\leq m\leq n^\alpha$

Comparison of time complexity for different large scale approaches: 1. Kernel Regularized Least Squares (KRLS), 2. KRLS combined with Partitioning, 3. Subsampling, 4. Subsampling combined with Partitioning, 5. Distributed KRLS on m machines