Tight Kernel Query Complexity of Kernel Ridge Regression and Kernel $k$-means Clustering

Manuel Fernández V, David P. Woodruff, Taisuke Yasuda
Overview

● Preliminaries
● Kernel ridge regression
● Kernel $k$-means clustering
● Query-efficient algorithm for mixtures of Gaussians
Kernel Method

- Many machine learning tasks can be expressed as a function of the inner product matrix $G$ of the data points (rather than the design matrix)
- Implicitly apply the exact same algorithm to the data set under a feature map through the use of a *kernel function*
- The analogue of the inner product matrix $G$ is called the *kernel matrix* $K$
Kernel Query Complexity

- In this work, we study *kernel query complexity*: the number of entries of the kernel matrix $\mathbf{K}$ read
Kernel Ridge Regression (KRR)

- Kernel method applied to ridge regression
  \[
  \alpha_{\text{opt}} = \underset{\alpha \in \mathbb{R}^n}{\text{argmin}} \|K\alpha - z\|_2^2 + \lambda \alpha^\top K\alpha
  \]
  \[
  = (K + \lambda I_n)^{-1}z
  \]

- Approximation guarantee
  \[
  \|\hat{\alpha} - \alpha_{\text{opt}}\|_2 \leq \varepsilon \|\alpha_{\text{opt}}\|_2
  \]
Query-Efficient Algorithms

- State of the art approximation algorithms have \textit{sublinear} and \textit{data-dependent} runtime and query complexity (Musco and Musco NeurIPS 2017, El Alaoui and Mahoney NeurIPS 2015)
- Sample $\tilde{O}(d_{\text{eff}}^\lambda / \varepsilon)$ rows proportionally to ridge leverage scores where
  \[
  d_{\text{eff}}^\lambda(K) := \text{tr} \left( K(K + \lambda I_n)^{-1} \right) = \sum_{i=1}^{r} \frac{\sigma_i^2}{\sigma_i^2 + \lambda}
  \]
- Query complexity $\tilde{O}(nd_{\text{eff}}^\lambda / \varepsilon)$
Contribution 1: Tight Lower Bounds for KRR

Theorem (informal)

Any randomized algorithm computing a \((1 + \varepsilon)\)-approximate KRR solution with probability at least 2/3 makes at least \(\Omega(n d_{\text{eff}}^{\lambda} / \varepsilon)\) kernel queries.

- Effective against randomized and adaptive (data-dependent) algorithms
- Tight up to logarithmic factors
Contribution 1: Tight Lower Bounds for KRR

Proof (sketch)

- By Yao’s minimax principle, suffices to prove for deterministic algorithms on a hard input distribution.
- Our hard input distribution: all ones vector for the target vector $z$, regularization $\lambda = n/k$. 
Contribition 1: Tight Lower Bounds for KRR

- Data distribution $\mu_{KRR}$ for the kernel matrix:

\[
\begin{align*}
\frac{2ne}{k} \\
\frac{n\epsilon}{k}
\end{align*}
\]
Contribution 1: Tight Lower Bounds for KRR

- Inner product matrix of standard basis vectors, $\frac{2n\varepsilon}{k}$ copies of $e_j$ for the first $\frac{k}{4\varepsilon}$ coordinates, and $\frac{n\varepsilon}{k}$ copies of the next $\frac{k}{2\varepsilon}$
- Half of the data points belong to “large clusters”, the other half belong to “small clusters”
- In order to label a row as “large cluster” or “small cluster”, any algorithm must read $\Omega\left(\frac{k}{\varepsilon}\right)$ entries of the row
- In order to label a constant fraction of rows, need to read $\Omega\left(\frac{nk}{\varepsilon}\right)$ entries of the kernel matrix
Contribution 1: Tight Lower Bounds for KRR

Lemma

Any randomized algorithm for labeling a constant fraction of rows of a kernel matrix drawn from $\mu_{KRR}$ must read $\Omega(nk/\varepsilon)$ kernel entries.

- Proven using standard techniques
Contribution 1: Tight Lower Bounds for KRR Reduction

Main Idea: one can just read off the labels of all the rows from the optimal KRR solution, and one can do this for a constant fraction of the rows from an approximate KRR solution.
Contribution 1: Tight Lower Bounds for KRR

- Let $K = U \Sigma U^T$ be the SVD of the kernel matrix.
- The columns are the eigenvectors of $K$ and the cluster size $n_j$ is the corresponding eigenvalue, and these are orthogonal.
- The target vector is the sum of these columns.

\[
z = \sum_{j \in [3J/4]} \sqrt{n_j} U e_j
\]
Contribution 1: Tight Lower Bounds for KRR
Contribution 1: Tight Lower Bounds for KRR

Optimal KRR solution

\[
\alpha_{opt} = (K + \lambda I_n)^{-1} z \\
= \sum_{j \in [3J/4]} \frac{1}{n_j + \lambda} \left( \sqrt{n_j} U e_j \right)
\]
Contribution 1: Tight Lower Bounds for KRR

Optimal KRR solution

\[ e_i^\top \alpha_{\text{opt}} = \begin{cases} (2n\varepsilon/k + n/k)^{-1} = \frac{k/n}{1+2\varepsilon} & \text{if row } i \text{ has block size } 2n\varepsilon/k \\ (n\varepsilon/k + n/k)^{-1} = \frac{k/n}{1+\varepsilon} & \text{if row } i \text{ has block size } n\varepsilon/k \end{cases} \]

Thus, the entries are separated by a multiplicative \((1 \pm \Omega(\varepsilon))\) factor.
Contribution 1: Tight Lower Bounds for KRR

Approximate KRR solution

- By averaging the approximation guarantee over the coordinates, we can still distinguish the cluster sizes for a constant fraction of the coordinates

\[ \|\hat{\alpha} - \alpha_{opt}\|_2 \leq \varepsilon \|\alpha_{opt}\|_2 \]
Contribution 1: Tight Lower Bounds for KRR

\[ d_{\text{eff}}^\lambda = \sum_{j \in [3J/4]} \frac{n_j}{n_j + \lambda} = \Theta \left( \sum_{j \in [3J/4]} \frac{n\varepsilon/k}{n\varepsilon/k + n/k} \right) = \Theta(k) \]
Contribution 1: Tight Lower Bounds for KRR

Remarks

- Settles a variant of an open question of El Alaoui and Mahoney: is the effective statistical dimension a lower bound on the query complexity? (they consider an approximation guarantee on the statistical risk instead of the argmin)
- Techniques extend to any indicator kernel function, including all kernels that are a function of the inner product or Euclidean distance
- Lower bound is easily modified to an instance where the top $d_{\text{eff}}^\lambda$ singular values scales as the regularization $\lambda$
Kernel $k$-means Clustering (KKMC)

- Kernel method applied to $k$-means clustering
- Objective: a partition of the data set into $k$ clusters that minimizes the sum of squared distances to the nearest centroid
- For a feature map $\varphi : \mathcal{X} \rightarrow \mathcal{F}$, objective function is

$$
cost(C) := \sum_{j=1}^{k} \sum_{x \in C_j} \left\| \varphi(x) - \mu_j \right\|_F^2
$$

$$
\mu_j := \frac{1}{|C_j|} \sum_{x \in C_j} \varphi(x)
$$
Contribution 2: Tight Lower Bounds for KKMC

Theorem (informal)

Any randomized algorithm computing a \((1 + \varepsilon)\)-approximate KKMC solution with probability at least 2/3 makes at least \(\Omega(\frac{nk}{\varepsilon})\) kernel queries.

- Effective against randomized and adaptive (data-dependent) algorithms
- Tight up to logarithmic factors
Contribution 2: Tight Lower Bounds for KKMC

- Similar techniques, hard distribution is sums of standard basis vectors

\[ k \text{ blocks} \]

\[ 1/\varepsilon \text{ coordinates} \]
Kernel $k$-means Clustering of Mixtures of Gaussians

- For input distributions encountered in practice, previous lower bound may be pessimistic
- We show that for a mixture of $k$ isotropic Gaussians, we can solve KKMC in only $\tilde{O}(n/\epsilon)$ kernel queries
Contribution 3: Query-Efficient Algorithm for Mixtures of Gaussians

Theorem (informal)

Given a mixture of $k$ Gaussians with mean separation $\tilde{O}(\sigma)$, there exists a randomized algorithm which returns a $(1 + \varepsilon)$-approximate $k$-means clustering solution reading $\tilde{O}(n/\varepsilon)$ kernel queries with probability at least 2/3.
Contribution 3: Query-Efficient Algorithm for Mixtures of Gaussians

Proof (sketch)

- Learn the means of the Gaussians in $\text{poly}(k, 1/\varepsilon, d)$ samples (Regev and Vijayaraghavan, FOCS 2017)
- Use the learned means to identify the true means of $O(\log n/\varepsilon)$ Gaussians
- Subtract off Gaussians from the same mean from each other to obtain zero-mean Gaussians
- Use the zero-mean Gaussians to sketch the data set in $O(n \log n/\varepsilon)$ samples
- Cluster the sketched data set