Distributed Adaptive Sampling for Kernel Matrix Approximation

Daniele Calandriello, Alessandro Lazaric, Michal Valko

informatics mathematics Inria

Motivation

- Kernel methods are *versatile* and *accurate*
- Strong generalization guarantees but *poor scalability*

$\mathcal{O}(n^3)$ *time* $\mathcal{O}(n^2)$ *space* (***n*** number of samples)

- *Current limitation:* Many approximate schemes are either ***not scalable*** or ***not accurate***

⇒ **We propose a ~~parallel~~ distributed incremental approximation scheme for kernel methods with complexity and error guarantees adaptive to the dataset and kernel structure**

⇒ **Runs in a single pass over the dataset, and can update its solution if new data arrives**

⇒ **On a single machine, computes a solution in subquadratic $\mathcal{O}(n)$ time, avoids constructing the whole K_n**

⇒ **On multiple machines, computes a solution in logarithmic $\mathcal{O}(\log(n))$ time, without increase in total work**

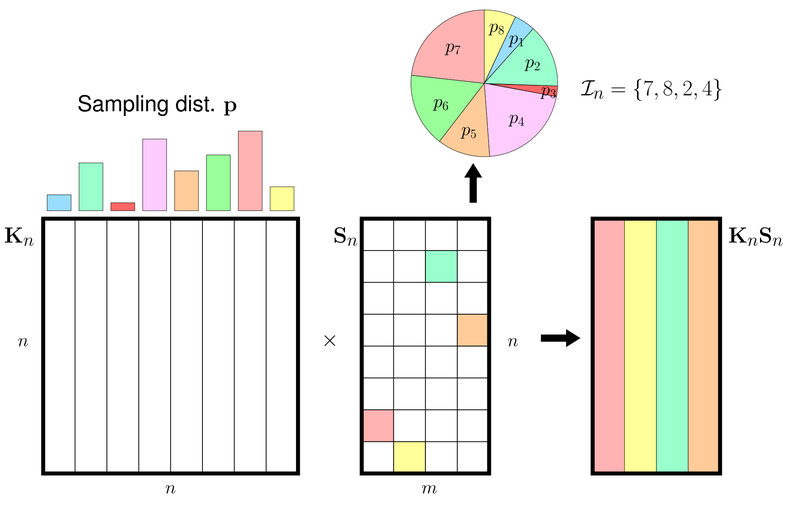
⇒ **Black-box applicability to many downstream tasks**

Nyström Approximation

Subsampling

1 Select a subset (dictionary) $\mathcal{I}_n$ of $\boldsymbol{m}$ representative samples

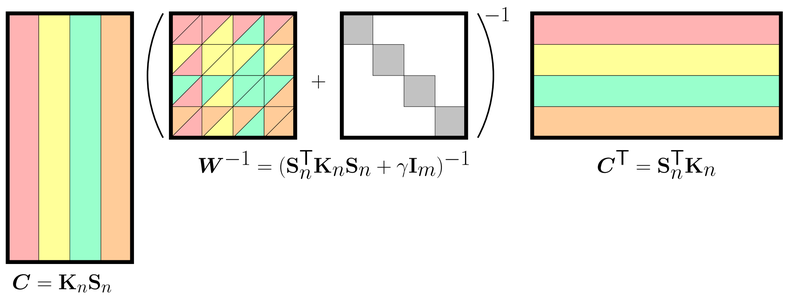
2 Constructs a sparse matrix S_n to select and reweight the columns associated with the points in $\mathcal{I}_n$



Low-Rank Approximation

3 Compute approximate, low-rank matrix $\widetilde{\mathsf{K}}_n = \mathsf{C}\mathsf{W}^{-1}\mathsf{C}^\mathsf{T}$ as

$$\widetilde{\mathsf{K}}_n = \mathsf{C}\mathsf{W}^{-1}\mathsf{C}^\mathsf{T} = \mathsf{K}_n\mathsf{S}_n\left(\mathsf{S}_n^\mathsf{T}\mathsf{K}_n\mathsf{S}_n + \boldsymbol{\gamma}\mathsf{I}_m\right)^{-1}\mathsf{S}_n^\mathsf{T}\mathsf{K}_n$$



E cient Solution

4 Compute approximate solution

Kernel Ridge Regression

$$\widetilde{\mathsf{w}}_n = (\widetilde{\mathsf{K}}_n + \mu\mathsf{I})^{-1}\mathsf{y}_n = \frac{1}{\mu}\left(\mathsf{y}_n - \mathsf{C}\left(\mathsf{C}^\mathsf{T}\mathsf{C} + \mu\mathsf{W}\right)^{-1}\mathsf{C}^\mathsf{T}\mathsf{y}_n\right)$$

Kernel K-Means

$$\min_{\mathbf{A}} \mathrm{Tr}(\mathsf{K}_n - \mathsf{A}\mathsf{A}^\mathsf{T}\mathsf{K}_n\mathsf{A}\mathsf{A}^\mathsf{T}) \sim \min_{\mathbf{A}} \mathrm{Tr}(\widetilde{\mathsf{K}}_n - \mathsf{A}\mathsf{A}^\mathsf{T}\widetilde{\mathsf{K}}_n\mathsf{A}\mathsf{A}^\mathsf{T})$$

Kernel PCA

$$\min_{\mathbf{Z}} \left\|\mathsf{K}_n - \mathsf{Z}\mathsf{Z}^\mathsf{T}\mathsf{K}_n\right\|_F \sim \min_{\mathbf{Z}} \left\|\widetilde{\mathsf{K}}_n - \mathsf{Z}\mathsf{Z}^\mathsf{T}\widetilde{\mathsf{K}}_n\right\|_F$$

Also Kernel CCA, Kernel [Your downstream problem here]

Scalability now depends on $\boldsymbol{m}$

Space: ~~$\mathcal{O}(n^2)$~~ $\Rightarrow \mathcal{O}(nm)$, **Time:** ~~$\mathcal{O}(n^3)$~~ $\Rightarrow \mathcal{O}(nm^2 + m^3)$

Problems:

? How to choose the sampling distribution?

? How to choose m?

References

[Alaoui and Mahoney (2015)] A. El Alaoui and M. W. Mahoney. Fast randomized kernel methods with statistical guarantees. In *NIPS*, 2015.

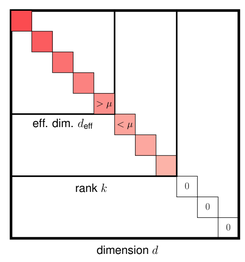
[Bach (2013)] F. Bach. Sharp analysis of low-rank kernel matrix approximations. In *COLT*, 2013.

[Calandriello et al. (2016)] D. Calandriello, A. Lazaric, and M. Valko. Analysis of Nyström method with sequential ridge leverage scores. In *UAI*, 2016.

[Rudi et al. (2015)] A. Rudi, R. Camoriano, and L. Rosasco. Less is more: Nyström computational regularization. In *NIPS*, 2015.

[Musco and Musco (2016)] C. Musco and C. Musco. Provably useful kernel matrix approximation in linear time. In arXiv, 2016.

Kernel Ridge Leverage Scores (RLS) Sampling



Definition 1. *Given a kernel matrix $\mathsf{K}_n \in \mathbb{R}^{n\times n}$, define*

γ-ridge leverage score

$$\tau_{n,i}(\gamma) = \mathsf{e}_{n,i}\mathsf{K}_n^\mathsf{T}(\mathsf{K}_n + \gamma\mathsf{I}_n)^{-1}\mathsf{e}_{n,i} = \phi(\mathsf{x}_i)^\mathsf{T}(\phi(\mathsf{X}_n)\phi(\mathsf{X}_n)^\mathsf{T} + \gamma\mathsf{I})^{-1}\phi(\mathsf{x}_i) \quad (1)$$

effective dimension

$$d_{\mathit{eff}}(\gamma)_n = \sum_{i=1}^n \tau_{n,i}(\gamma) = \mathrm{Tr}\left(\mathsf{K}_n(\mathsf{K}_n + \gamma\mathsf{I}_n)^{-1}\right) \quad (2)$$

Proposition 1 (Alaoui, Mahoney, 2015). *Let $\boldsymbol{\varepsilon}$ be the accuracy, δ the confidence. If the regularized Nystrom approximation $\widetilde{\mathsf{K}}_n$ is computed using a sampling distribution proportional to $\boldsymbol{\tau_{n,i}}$, and at least*

$$m \geq \left(\frac{2\boldsymbol{d_{\mathit{eff}}(\gamma)_n}}{\boldsymbol{\varepsilon^2}}\right)\log\left(\frac{n}{\delta}\right)$$

columns, then with probability $1-\delta$, $0 \preceq \mathsf{K}_n - \widetilde{\mathsf{K}}_n \preceq \frac{\gamma}{1-\varepsilon}\mathsf{I}_n$.

Intuitively: $\tau_{n,i}$ *sensitivity* of prediction on point x_i
$\Rightarrow \widehat{y}_{n,i} = \mathsf{e}_i^\mathsf{T}(\mathsf{K}_n\widehat{\mathsf{w}}_n) = \mathsf{e}_i^\mathsf{T}\mathsf{K}_n(\mathsf{K}_n + \mu\mathsf{I})^{-1}\mathsf{y}_n$

Pros: + m scales with the e□ective dimension

Cons: - computing $\tau_{n,i}(\mu)$ is as di□cult as solving the original problem
- the probabilities need be **recomputed at any new sample** (=multipass)

SQUEAK

Lemma 1. *Assume that the dictionary $\mathcal{I}_{t-1}$ induces a γ-approx. $\widetilde{\mathsf{K}}_{t-1}$, and let $\overline{\mathsf{S}}_t$ be constructed by adding $\overline{q}$ copies of $(\overline{q})^{-1/2}\mathsf{e}_{t,t}$ to the selection matrix. Then, denoting $\alpha = (1+\varepsilon)/(1-\varepsilon)$, for all i such that $i \in \{\mathcal{I}_{t-1} \cup \{t\}\}$,*

$$\widetilde{\tau}_{t,i} = \frac{1+\varepsilon}{\alpha\gamma}\left(k_{i,i} - \mathsf{k}_{t,i}\overline{\mathsf{S}}\left(\overline{\mathsf{S}}^\mathsf{T}\mathsf{K}_t\overline{\mathsf{S}} + \gamma\mathsf{I}\right)^{-1}\overline{\mathsf{S}}^\mathsf{T}\mathsf{k}_{t,i}\right), \quad (3)$$

is an α-approximation of the RLS $\tau_{t,i}$, that is $\tau_{t,i}(\gamma)/\alpha \leq \widetilde{\tau}_{t,i} \leq \tau_{t,i}(\gamma)$.

SQUEAK

Input: $\mathcal{D}$, regularization $\gamma, \overline{q}, \varepsilon$, **Output**: $\mathcal{I}_n$

1: Initialize $\mathcal{I}_0$ as empty, $\widetilde{p}_{1,0} = 1$
2: **for** $t = 1, \ldots, n$ **do**
3: Receive new sample x_t
4: Compute α-app. RLS $\{\widetilde{\tau}_{t,i} : i \in \mathcal{I}_{t-1} \cup \{t\}\}$, using $\mathcal{I}_{t-1}$, x, and Eq. 3
5: Set $\widetilde{\mathsf{p}}_{\mathbf{t,i}} = \min\{\widetilde{\tau}_{\mathbf{t,i}}, \widetilde{\mathsf{p}}_{\mathbf{t-1,i}}\}$
6: Initialize $\mathcal{I}_t = \emptyset$
7: **for all** $j \in \{1, \ldots, t-1\}$ **do** } Shrink
8: **if** $q_{t-1,j} \neq 0$ **then** } Shrink
9: $\mathsf{q}_{\mathbf{t,j}} \sim \mathcal{B}(\widetilde{\mathsf{p}}_{\mathbf{t,j}}/\widetilde{\mathsf{p}}_{\mathbf{t-1,j}}, \mathsf{q}_{\mathbf{t-1,j}})$ } Shrink
10: Add $(j, \phi_j, q_{t,j}, \widetilde{p}_{t,j})$ to $\mathcal{I}_t$. } Shrink
11: **end if** } Shrink
12: **end for** } Shrink
13: $\mathsf{q}_{\mathbf{t,t}} \sim \mathcal{B}(\widetilde{\mathsf{p}}_{\mathbf{t,t}}, \overline{\mathsf{q}})$ } Expand
14: Add $q_{t,t}$ copies of $(t, \phi_t, q_{t,t}, \widetilde{p}_{t,t})$ to $\mathcal{I}_t$ } Expand
15: **end for** } Dict-Update

Theorem 1. *Let $\alpha = (\frac{1+\varepsilon}{1-\varepsilon})$ and $\gamma > 1$. For any $0 \leq \varepsilon \leq 1$, and $0 \leq \delta \leq 1$, if we run* SQUEAK *with $\overline{q} = \mathcal{O}(\frac{\alpha}{\varepsilon^2}\log(\frac{n}{\delta}))$, then w.p. $1-\delta$, for all $t \in [n]$*

(1) $\widetilde{\mathsf{K}}_t$ computed with $\mathcal{I}_t$ is a γ-approximation of K_t.

(2) $|\mathcal{I}_t| = \sum_i Q_{t,i} \leq \mathcal{O}(\overline{q}d_{\mathit{eff}}(\gamma)_t) \leq \mathcal{O}(\frac{\alpha}{\varepsilon^2}\boldsymbol{d_{\mathit{eff}}(\gamma)_n}\log(\frac{\boldsymbol{n}}{\boldsymbol{\delta}}))$.

Accuracy and space/time anytime guarantees, matches exact RLS sampling.

Using unnormalized $\widetilde{p}_{t,i}$, no need for appr. $d_{\mathrm{eff}}(\gamma)_t$

Only need to compute RLS for points in $\mathcal{I}_t$, never recompute after dropping
↳Never construct the whole K_n, subquadratic runtime ~~$\mathcal{O}(n^2|\mathcal{I}_n|^2)$~~ $\Rightarrow \mathcal{O}(n|\mathcal{I}_n|^3)$

Store points directly in the dictionary
↳$\mathcal{O}(\boldsymbol{d}_{\mathrm{eff}}(\gamma)_n^2 + \boldsymbol{d}_{\mathrm{eff}}(\gamma)_n\boldsymbol{d})$ space constant in n
Single pass over the dataset (streaming)

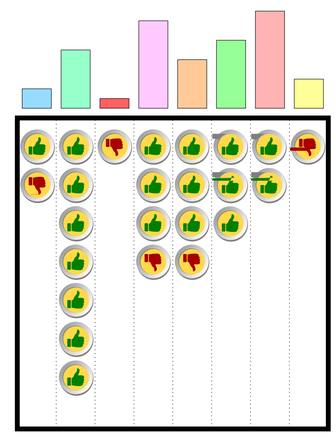
Dictionary changes a lot between iteration, total runtime $\mathcal{O}(n|\mathcal{I}_n|^3)$

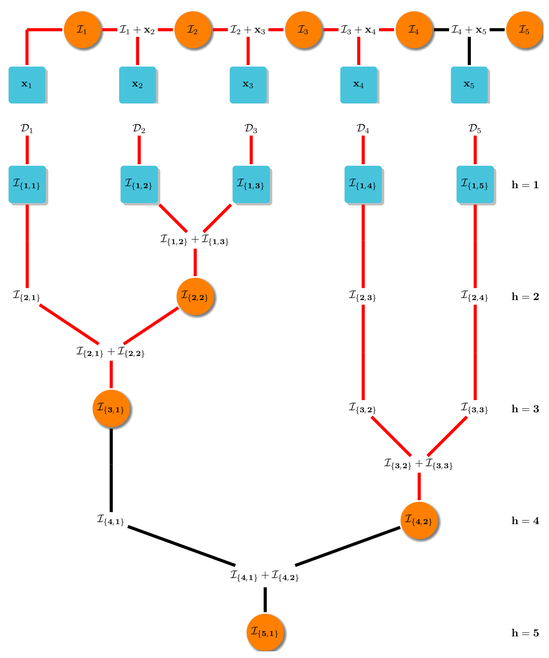
Extend Dict-Update (point + dict.) to Dict-Merge (dict. + dict.)
↳Distributed SQUEAK, multiple workers in parallel, without sharing memory
Recursive merging to build dictionary, $\mathcal{O}(\log(n)|\mathcal{I}_n|^3)$ time, $\mathcal{O}(n|\mathcal{I}_n|^3)$ work

- $\widetilde{\tau}_{t,i} = \mathsf{e}_i^\mathsf{T}\widetilde{\mathsf{K}}_t(\widetilde{\mathsf{K}}_t + \gamma\mathsf{I})^{-1}\mathsf{e}_i$ would fail
- Instead, approximate $\tau_{t,i}$ directly in RKHS $\widetilde{\tau}_{t,i} = \phi(\mathsf{x}_i)^\mathsf{T}(\phi(\mathsf{X}_t)\overline{\mathsf{S}}\overline{\mathsf{S}}^\mathsf{T}\phi(\mathsf{X}_t)^\mathsf{T} + \gamma\mathsf{I})^{-1}\phi(\mathsf{x}_i)$ and then reformulate using kernel trick
- $\widetilde{\tau}_{t,i}$ can be computed in $\mathcal{O}(|\mathcal{I}_t|^2)$ space and $\mathcal{O}(|\mathcal{I}_t|^3)$ time, independent from t.
- $\widetilde{\tau}_{t,i}$ for samples in $\mathcal{I}_t$ can be computed using only samples contained in $\mathcal{I}_t$.
- The formulation of $\widetilde{\tau}_{t,i}$ is not incremental

Proposition 2. *For any kernel matrix K_{t-1} and its bordering K_t,*

$$\tau_{t,i} \leq \tau_{t-1,i}, \quad d_{\mathit{eff}}(\gamma)_t \geq d_{\mathit{eff}}(\gamma)_{t-1}.$$





	Time	$\lvert\mathcal{I}_n\rvert$	Incr.
Exact	n^3	n	-
Bach'13	$\frac{nd_{\max,n}^2}{\varepsilon} + \frac{d_{\max,n}^3}{\varepsilon}$	$\frac{d_{\max,n}}{\varepsilon}$	No
A&M'15	$n(\lvert\mathcal{I}_n\rvert)^2$	$\left(\frac{\lambda_{\min}+n\mu\varepsilon}{\lambda_{\min}-n\mu\varepsilon}\right)d_{\mathrm{eff}}(\gamma)_n$	No
INK (C&al'16)	$\frac{\lambda_{\max}^2}{\gamma^2}\frac{n^2 d_{\mathrm{eff}}(\gamma)_n^2}{\varepsilon^2}$	$\frac{\lambda_{\max}}{\gamma}\frac{d_{\mathrm{eff}}(\gamma)_n}{\varepsilon^2}$	Yes
SQUEAK	$\frac{n^2 d_{\mathrm{eff}}(\gamma)_n^2}{\varepsilon^2}$	$\frac{d_{\mathrm{eff}}(\gamma)_n}{\varepsilon^2}$	Yes

Downstream guarantees (Musco & Musco 2016)

RLS sampling preserves well the projection on K_n's range $\mathsf{P} = \mathsf{K}_n^{1/2}(\mathsf{K}_n + \gamma\mathsf{I})^{-1}\mathsf{K}_n^{1/2} = \phi(\mathsf{X}_n)^\mathsf{T}(\phi(\mathsf{X}_n)\phi(\mathsf{X}_n)^\mathsf{T} + \gamma\mathsf{I})^{-1}\phi(\mathsf{X}_n)$

Kernel ridge regression:
$\widehat{y}_{n,i} = \mathsf{e}_i^\mathsf{T}\mathsf{K}_n(\mathsf{K}_n + \mu\mathsf{I})^{-1}\mathsf{y}_n = \mathsf{e}_i^\mathsf{T}\mathsf{P}\mathsf{y}_n$
$\widetilde{w}_n = (\widetilde{\mathsf{K}}_n + \gamma\mathsf{I})^{-1}\mathsf{y}_n$

$$R(\widetilde{w}_n) \leq \left(1 + \frac{1}{1-\varepsilon}\right)R(\widehat{w}_n)$$

Kernel PCA:
$\mathsf{K}_n = \mathsf{U}\Lambda\mathsf{U}^\mathsf{T}$, $\mathsf{P} = \mathsf{U}\Lambda(\Lambda + \gamma\mathsf{I})^{-1}\mathsf{U}^\mathsf{T}$
$\widetilde{\mathsf{Z}}$ computed using $\widetilde{\mathsf{K}}_n = \widetilde{\mathsf{U}}\widetilde{\Lambda}\widetilde{\mathsf{U}}^\mathsf{T}$

$$\left\|\mathsf{K}_n - \widetilde{\mathsf{Z}}\widetilde{\mathsf{Z}}^\mathsf{T}\mathsf{K}_n\right\|_F \leq (1+2\varepsilon)\left\|\mathsf{K}_n - \mathsf{Z}^*\mathsf{Z}^{*\mathsf{T}}\mathsf{K}_n\right\|_F$$

Kernel K-Means:
$\widetilde{\mathsf{A}}$ ρ-optimal cluster assignment for $\widetilde{\mathsf{K}}_n$
A^* optimal cluster assignment for K_n
$\xi = (1+\varepsilon)(1+\rho)$

$$\mathrm{Tr}(\mathbf{K}_n - \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}^\mathsf{T}\mathbf{K}_n\widetilde{\mathbf{A}}\widetilde{\mathbf{A}}^\mathsf{T}) \leq \xi\,\mathrm{Tr}(\mathbf{K}_n - \mathbf{A}^\square\mathbf{A}^{\square\mathsf{T}}\mathbf{K}_n\mathbf{A}^\square\mathbf{A}^{\square\mathsf{T}})$$