On Submodular Set Cover Problems for Near-Optimal Online Kernel Basis Selection

Hrusikesha Pradhan*, Alec Koppel†, and Ketan Rajawat*

***Abstract*—Non-parametric function approximators provide a principled way to fit nonlinear statistical models while affording formal performance guarantees. However, their complexity drawbacks are well-understood: they define a statistical representation whose complexity scales with the sample size through the fact that they retain all past samples. In the case of streaming data, this complexity may grow unbounded. One is faced with the question of how to suitably trade off representational complexity with statistical accuracy, which may be addressed with various approximation methods. In this work, we formalize that greedy-based approximations, under suitably chosen compression statistics, can admit near-optimal representations. The key driver of this result is a novel connection between the reproducing kernel Hilbert Space (RKHS) norm and the log-determinant of the kernel matrix, which has been shown to be a submodular set function of a collection of points. This allows us to design a constructive variant of a greedy subspace projections in [1], [2] according to a submodular set cover (SSC) problem, which provably picks at most logarithmically more elements than the optimal one. We validate our constructive approach by doing simulation on real ocean data from the Gulf of Mexico [3].**

***Index terms*—** Non-parametric function learning, submodularity, online learning, kernel regression

I. Introduction

We consider an expected risk minimization (ERM) problem, where the objective is to learn the regressor by minimizing the loss function, thereby quantifying the merit of the statistical model. Our focus is on the case where the regressor is not a vector-valued parameter $w \in \mathbb{R}^p$ but rather is a function f belonging to a reproducing kernel Hilbert space (RKHS) [4]. These non-linear statistical models provide descriptive richness owing to their universal approximation properties [5] as compared to linear models. Unfortunately because of the setting expected value minimization, their representational complexity (via the Representer Theorem [6]) scales quadratically with the sample size, which may be unbounded [7].

To minimize such expected risks without any concern for the complexity of the feasible set, one may employ first-order methods such as gradient descent or its stochastic variants [8]. However, with the RKHS representation, one must co-design first-order method with the complexity of the function class. To do so, one may either do online sparsification in a way that is either tied to the optimization procedure (supervised [9], [10]) or solely based on finding a sparse representation of the function (unsupervised [11]–[15]). We note that the favorable performance in theory and practice of tethering the rule for complexity reduction to the stochastic gradient update is well-documented in [1], where a greedily constructed [16] subspace projection is composed with functional variant of SGD. That the learned sequence finiteness of the function representation (model) complexity is established in [1], with follow-up work providing a non-asymptotic bound in [17].

However, the guarantees for these methods are only for the sub-optimality of the RKHS element, but not on whether the function's parameterization in terms of past points are close to the optimal set of points of a fixed size. To address this gap, we identify that the point selection problem associated with these subspace projections can be cast as a submodular set cover (SSC) problem [18]. SSC has been extensively studied in combinatorial optimization, where one seeks to obtain the smallest subset satisfying a certain utility, given by a monotone function [19]–[22] that satisfies submodularity [23], which formalizes the notion of diminishing returns. It is well-known that constructive greedy subset selection returns a solution of at most $(1 + \log \gamma)k^\star$, where $k^\star$ is the size of the optimal solution set and γ subsumes problem parameters – see [18].

Our main contribution is to make a link between the RKHS norm as a compression statistic and the log-determinant of the kernel matrix which has shown to be a submodular set function in [22], [24], and thereby cast the point selection task as a SSC problem. In doing so, we provide for the first time formal guarantees for the retained points of RKHS subspace projections that are executed constructively. Experimentally, we demonstrate on real data that we indeed retain near-optimal collections of points. We note that similar lines of reasoning are applicable to other areas of approximate Bayesian inference, under the hypothesis that invocations of submodularity are lurking behind dimensionality reduction methodologies.

II. Problem Statement

We formulate the ERM problem by considering a convex loss function $\ell : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ which characterizes the goodness-of-fit of the estimator $f \in \mathcal{H}$ evaluated at $\mathbf{x}$ averaged over all possible training samples, and we also consider adding a Tikhonov regularizer $\|f\|_{\mathcal{H}}^2$ for ensuring stability and write the problem as

$$f^* = \operatorname*{argmin}_{f \in \mathcal{H}} \mathbb{E}_{\mathbf{x},\mathbf{y}} \Big[\ell(f(\mathbf{x}), y) \Big] + \frac{\lambda}{2} \|f\|_{\mathcal{H}}^2. \tag{1}$$

The problem (1) is intractable in general. However, for the case when $\mathcal{H}$ is equipped with a reproducing kernel $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, we can transform the functional optimization in (1) to the parametric form using the famous Representer Theorem [7], [25]. In particular, if $\mathcal{H}$ is equipped with a unique *kernel function*, $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, such that:

$$(i)\ \langle f, \kappa(\mathbf{x}, \cdot) \rangle_{\mathcal{H}} = f(\mathbf{x}),\ (ii)\ \mathcal{H} = \overline{\text{span}\{\kappa(\mathbf{x}, \cdot)\}}, \tag{2}$$

for all $\mathbf{x} \in \mathcal{X}$, and $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denotes the Hilbert inner product for $\mathcal{H}$. We further assume that the kernel is positive semidefinite, i.e. $\kappa(\mathbf{x}, \mathbf{x}') \geq 0$ for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$. The function spaces

*Department of Elect. Engg., Indian Institute of Technology Kanpur, Kanpur, Uttar Pradesh, India. Email: {hpradhan, ketan}@iitk.ac.in.

†Supply Chain Optimization Technologies, Amazon, Bellevue, WA 98004. Email: aekoppel@amazon.com. Work completed while at the U.S. Army Research Laboratory in Adelphi, MD 20783.

equipped with such structure are called RKHS [26]. For kernelized and regularized empirical risk minimization, we can write the function $f \in \mathcal{H}$ in terms of kernels evaluated only at training samples using the Representer Theorem [4], [6] as

$$f(\mathbf{x}) = \sum_{n=1}^{N} w_n \kappa(\mathbf{x}_n, \mathbf{x}) \ . \tag{3}$$

where $\mathbf{w} = [w_1, \cdots, w_N]^T \in \mathbb{R}^N$ denotes a set of weights. The upper summand index N in (3) is henceforth referred to as the model order. We proceed with writing out the functional stochastic gradient (FSGD) method [1] to solve (1) iteratively

$$f_{t+1} = (1 - \eta_t \lambda) f_t - \eta_t \ell'(f_t(\mathbf{x}_t), y_t) \kappa(\mathbf{x}_t, \cdot) \ , \tag{4}$$

where $\eta_t > 0$ is an algorithm step-size. Now with $\lambda > 0$, step-size $\eta_t < 1/\lambda$ and initialization $f_0 = 0 \in \mathcal{H}$, we write the function f_t at time t using the Representer Theorem given in (3) in terms of samples seen so far as $f_t(\mathbf{x}) = \sum_{n=1}^{t-1} w_n \kappa(\mathbf{x}_n, \mathbf{x}) = \mathbf{w}_t^T \boldsymbol{\kappa}_{\mathbf{X}_t}(\mathbf{x})$, where $\mathbf{X}_t = [\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}] \in \mathbb{R}^{p \times (t-1)}$ and $\boldsymbol{\kappa}_{\mathbf{X}_t}(\cdot) = [\kappa(\mathbf{x}_1, \cdot), \ \ldots \ , \kappa(\mathbf{x}_{t-1}, \cdot)]^T$. The Representer Theorem together with the stochastic functional update (4) then permits us to rewrite the functional update in terms of a growing data matrix called a kernel dictionary $\mathbf{X}$ and coefficients $\mathbf{w}$:

$$\mathbf{X}_{t+1} = [\mathbf{X}_t, \mathbf{x}_t], \ \mathbf{w}_{t+1} = [(1 - \eta_t \lambda)\mathbf{w}_t, -\eta_t \ell'(f_t(\mathbf{x}_t), y_t)] \tag{5}$$

Observe that $\mathbf{X}_{t+1}$ has one more column than $\mathbf{X}_t$, an instance of the curse of kernelization. We define the *model order* as number of data points M_t in the dictionary at time t (the number of columns of $\mathbf{X}_t$). FSGD is such that $M_t = t - 1$, and hence grows unbounded with iteration index t.

Subspace Projections. To address this unbounded memory growth, one may project the function sequence onto a lower dimensional subspace $\mathcal{H}_{\mathbf{D}}$ ($\mathcal{H}_{\mathbf{D}} \subseteq \mathcal{H}$), where $\mathcal{H}_{\mathbf{D}}$ is represented by compressed dictionary $\mathbf{D} = [\mathbf{d}_1, \ \ldots, \ \mathbf{d}_M] \in \mathbb{R}^{p \times M}$. Thus, instead of considering $\mathbf{D} = \mathbf{X}_{t+1}$, we consider a different dictionary $\mathbf{D} = \mathbf{D}_{t+1} \in \mathbb{R}^{p \times M_{t+1}}$, which is extracted from the data points observed thus far, at each iteration, with $M_{t+1} \ll t$. Considering this, we first write the parametric representation of the functional update (4) in terms of these parameters as

$$\tilde{\mathbf{D}}_{t+1} = [\mathbf{D}_t, \mathbf{x}_t], \ \tilde{\mathbf{w}}_{t+1} = [(1 - \eta_t \lambda)\mathbf{w}_t, -\eta_t \ell'(f_t(\mathbf{x}_t), y_t)], \tag{6}$$

Previously, a destructive variant of kernel orthogonal matching pursuit algorithm (KOMP) [27] has been employed to select the sequence of dictionary and coefficient parameters in [1]. In this work, we depart from this approach by instead developing a constructive approach, which we rigorously motivate through its links to combinatorial optimization, specifically, SSC problems [18]. Thus, next we present the dictionary selection problem as a SSC problem and the constructive version of KOMP [27] algorithm to solve the problem (7) in Alg. 1.

A. Submodular Set cover under Functional Settings

We now present the function approximation problem of finding dictionary $\mathbf{D}$ as a submodular set cover problem:

$$\mathbf{D}_{t+1}^{\text{opt}} = \underset{\mathbf{D} \subset \tilde{\mathbf{D}}_{t+1}}{\operatorname{argmin}} \ |\mathbf{D}|, \ s.t. \ F(\mathbf{D}) \geq Q, \tag{7}$$

where the motivation and exact form of $F(\mathbf{D})$ and Q (24) will be discussed in Lemma 2. Thus at every iteration t, we receive sample $\mathbf{x}_t$ and have dictionary $\tilde{\mathbf{D}}_{t+1} = [\mathbf{D}_t, \ \mathbf{x}_t]$ and we solve the above problem (7) to obtain the smallest dictionary $\mathbf{D} \subset \tilde{\mathbf{D}}_{t+1}$ which approximates the function with ϵ error. Next we present the algorithm of obtaining the dictionary $\mathbf{D}_{t+1}$ in Algorithm 1. The Algorithm 1 starts with an empty dictionary, i.e., $\mathbf{D} = \emptyset$ and we keep on adding a dictionary element to $\mathbf{D}$ till the function $F(\mathbf{D})$ (see (24)) achieves the threshold Q (see (24)). And the moment we have the inequality $F(\mathbf{D}) \geq Q$, we exit out of the loop which adds elements to the dictionary $\mathbf{D}$ (see the while loop in Algorithm 1). This algorithm is called as constructive since it starts from an empty dictionary and keeps on adding element to the dicitonary in comparison to the KOMP algorithm in [1] which at every step removes element from the original dictionary till the stopping criterion is met (see Algorithm 1 in [1]). Then, next we compute the weights defined by dictionary $\mathbf{D}$. Thus, to summarize the CFG algorithm, at every iteration index t, it basically does:

$$(f_{t+1}, \mathbf{D}_{t+1}, \mathbf{w}_{t+1}) = \mathbf{CFG}(\tilde{f}_{t+1}, \tilde{\mathbf{D}}_{t+1}, \tilde{\mathbf{w}}_{t+1}, Q). \tag{8}$$

Next, we present the standard SSC problem.

Preliminaries: Standard SSC Problem - We consider a function F which can be a performance measure function or some accuracy measuring function that measures the quality of a given subset $\mathcal{A} \subseteq \mathcal{V} = \{1, \ldots, V\}$. We define the marginal gain associated with a given element $e \in \mathcal{V}$ w.r.t some set $\mathcal{A} \subseteq \mathcal{V}$ as $\rho_e(\mathcal{A}) \coloneqq F(\mathcal{A} \cup \{e\}) - F(\mathcal{A})$. The function F is considered to be -

(i) **Monotone**: For all $\mathcal{S}$, $\mathcal{T}$ such that $\mathcal{S} \subseteq \mathcal{T} \subseteq \mathcal{V}$, we have $F(\mathcal{S}) \leq F(\mathcal{T})$;

(ii) **Submodular**: For all all $\mathcal{S}$, $\mathcal{T}$ such that $\mathcal{S} \subseteq \mathcal{T}$, and for all $e \in \mathcal{V} \setminus \mathcal{T}$, we have $\rho_e(\mathcal{S}) \geq \rho_e(\mathcal{T})$.

Now, we present the famous SSC problem [18], where the objective is to find the smallest subset $\mathcal{A} \subset \mathcal{V}$ that satisfies a certain utility Q [20]–[22], i.e.,

$$\mathcal{A}^* = \underset{\mathcal{A} \subset \mathcal{V}}{\operatorname{argmin}} \ |\mathcal{A}|, \ s.t. \ F(\mathcal{A}) \geq Q. \tag{9}$$

With these preliminaries set, we next present the relation between RKHS norm as a compression measure and the log-determinant of the kernel matrix, and move onto to showing how the problem (7) was formulated.

Algorithm 1 Constructive Functional Greedy (CFG)

Require: Dictionary $\tilde{\mathbf{D}} \in \mathbb{R}^{p \times \tilde{M}}$ representing function $\tilde{f}$, model order $\tilde{M}$ coeffs. $\tilde{\mathbf{w}} \in \mathbb{R}^{\tilde{M}}$, threshold Q

initialize $i = 1$, $\mathbf{D}^0 = \emptyset$(empty set), model order $M = 0$.

while $F(\mathbf{D}^{i-1}) < Q$ and $|\mathbf{D}^{i-1}| < |\tilde{\mathbf{D}}|$ **do**

$\mathbf{d}^* \coloneqq \operatorname{argmax}_{\mathbf{d} \in \tilde{\mathbf{D}} \setminus \mathbf{D}^{i-1}} F(\mathbf{D}^{i-1} \cup \{\mathbf{d}\})$

Update dictionary: $\mathbf{D}^i \leftarrow \mathbf{D}^{i-1} \cup \{\mathbf{d}^*\}$

Update model complexity and i: $M = M + 1, i = i + 1$

end while

Compute weights $\mathbf{w}$ defined by final dict. $\mathbf{D} = \mathbf{D}^{i-1}$

$$\mathbf{w} = \underset{\mathbf{w} \in \mathbb{R}^M}{\operatorname{argmin}} \|\tilde{f}(\cdot) - \mathbf{w}^T \boldsymbol{\kappa}_{\mathbf{D}}(\cdot)\|_{\mathcal{H}}$$

return $f, \mathbf{D}, \mathbf{w}$ of model order $M \leq \tilde{M}$ s.t. $\|f - \tilde{f}\|_{\mathcal{H}} \leq \epsilon$

III. Non-parametric learning: SSC framework

With problem (9) presented above, our objective now is to formulate the problem of finding the smallest dictionary $\mathbf{D}$ representing function f satisfying $\|f - \tilde{f}\|_{\mathcal{H}} \leq \epsilon$ as a SSC problem. Before going into those details we first present Lemma 1, which allows us to relate the RKHS-norm difference to the logarithm of the determinant of the kernel matrix $\mathbf{K}_{\mathbf{D},\mathbf{D}}$ as a function of the kernel dictionary $\mathbf{D}$ criterion. We abbreviate this quantity as *logdet* subsequently, and it will be our specification of F in the following subsections.

Lemma 1. *The Hilbert norm difference of approximating $\tilde{f}$ by a function $f_{\mathbf{D},\mathbf{w}}$ represented by dictionary $\mathbf{D}$ and weights $\mathbf{w}$ can be upper-bounded as:*

$$\min_{\mathbf{D}} \min_{\mathbf{w}} \|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2 \leq \tilde{M} \min_{\mathbf{D}} \sum_{i=1}^{\tilde{M}} v_i^2 \kappa(\mathbf{x}_i, \mathbf{x}_i) - v_i^2 \mathbf{k}_i^T (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I})^{-1} \mathbf{k}_i. \quad (10)$$

Proof. Let $\tilde{f}$ be the function defined by dictionary points $\tilde{\mathbf{D}} = \{\mathbf{x}_i\}_{i=1}^{\tilde{M}}$ and weights $\{v_i\}_{i=1}^{\tilde{M}}$. We want to approximate $\tilde{f}$ by a function $f_{\mathbf{D},\mathbf{w}}$ with a compressed dictionary $\mathbf{D} = \{\mathbf{d}_j\}_{j=1}^{M}$ having M number of dictionary points and weights $\{w_j\}_{j=1}^{M}$. We formulate the RKHS norm minimization problem as:

$$\min_{\mathbf{D}} \min_{\mathbf{w}} \|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2 = \min_{\mathbf{D}} \min_{\mathbf{w}} \| \sum_{i=1}^{\tilde{M}} v_i \phi(\mathbf{x}_i) - \sum_{j=1}^{M} w_j \phi(\mathbf{d}_j) \|_{\mathcal{H}}^2,$$

where $\phi(\cdot)$ is the nonlinear mapping that assigns to each $\mathbf{x}$ the kernel function $\kappa(\cdot, \mathbf{x})$. We introduce dummy variables w_j^i such that $w_j = \sum_{i=1}^{\tilde{M}} w_j^i$ and write the above problem as

$$= \min_{\mathbf{D}} \min_{\mathbf{w}} \| \sum_{i=1}^{\tilde{M}} \Big[v_i \phi(\mathbf{x}_i) - \sum_{j=1}^{M} w_j^i \phi(\mathbf{d}_j) \Big] \|_{\mathcal{H}}^2. \quad (11)$$

Taking the summation outside, we upper bound (11) as

$$(11) \leq \tilde{M} \min_{\mathbf{D}} \min_{\mathbf{w}} \sum_{i=1}^{\tilde{M}} \| \Big[v_i \phi(\mathbf{x}_i) - \sum_{j=1}^{M} w_j^i \phi(\mathbf{d}_j) \Big] \|_{\mathcal{H}}^2$$

$$= \tilde{M} \min_{\mathbf{D}} \sum_{i=1}^{\tilde{M}} \min_{w_1^i, \ldots, w_M^i} \| v_i \phi(\mathbf{x}_i) - \sum_{j=1}^{M} w_j^i \phi(\mathbf{d}_j) \|_{\mathcal{H}}^2. \quad (12)$$

For $\mathbf{k}_i = [\kappa(\mathbf{x}_i, \mathbf{d}_1), \ldots, \kappa(\mathbf{x}_i, \mathbf{d}_M)]$, we write the inner minimization term in (12) by considering regularization as $\min_{\mathbf{w}^i} \| v_i \phi(\mathbf{x}_i) - \sum_{j=1}^{M} w_j^i \phi(\mathbf{d}_j) \|_{\mathcal{H}}^2 + \mu \|\mathbf{w}^i\|^2$

$$= \min_{\mathbf{w}^i} \mathbf{w}^{i^T} (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I}) \mathbf{w}^{i^T} - 2 v_i \mathbf{w}^{i^T} \mathbf{k}_i + v_i^2 \kappa(\mathbf{x}_i, \mathbf{x}_i). \quad (13)$$

Solving the inner minimization problem yields $\mathbf{w}^i = v_i (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I})^{-1} \mathbf{k}_i$. Now using the value of $\mathbf{w}^i$ in (13), we write the optimization problem over dictionary $\mathbf{D}$ and get

$$(12) \leq \tilde{M} \min_{\mathbf{D}} \sum_{i=1}^{\tilde{M}} \min_{w_1^i, \ldots, w_M^i} \| v_i \phi(\mathbf{x}_i) - \sum_{j=1}^{M} w_j^i \phi(\mathbf{d}_j) \|_{\mathcal{H}}^2 + \mu \|\mathbf{w}^i\|^2$$

$$\leq \tilde{M} \min_{\mathbf{D}} \sum_{i=1}^{\tilde{M}} v_i^2 \kappa(\mathbf{x}_i, \mathbf{x}_i) - v_i^2 \mathbf{k}_i^T (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I})^{-1} \mathbf{k}_i. \quad (14)$$

□

With this result stated, we may shift to expanding upon the relationship between the RKHS-norm error and $\log\det$.

A. Function approximation as log-determinant formulation

Before moving into the key results of our work, we bring out the difference between our work and [28]. In [28], the authors have also solved the dictionary (representing a function) selection problem by framing it as a cardinality constrained submodular maximization problem, where given a constraint l, the goal is to choose at most l elements that attain the largest possible utility. However, the approach of our work and the algorithm to solve it is very different from [28], since we frame it as a SSC problem, where given a utility Q, the goal is to find the minimum number of elements that can achieve it.

Next, we frame the dictionary selection problem w.r.t compression budget ϵ, i.e., $\|f - \tilde{f}\|_{\mathcal{H}} \leq \epsilon$ as a equivalent *logdet* problem, and present the exact forms of $F(\mathbf{D})$ and Q introduced in (7). This result is presented below in Lemma 2. Next, we go on to show how this formulation leads us to cast the dictionary learning problem as the functional SSC problem presented in (7).

Lemma 2. *The smallest dictionary representing the approximated function $f_{\mathbf{D},\mathbf{w}}$ automatically satisfies the criteria $\|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}} \leq \epsilon$ if the dictionary $\mathbf{D}$ satisfies the inequality*

$$F(\mathbf{D}) \geq Q, \quad (15)$$

where $F(\mathbf{D}) \coloneqq \log\det(\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu\mathbf{I})$, and $Q \coloneqq \log\Big[\int_{\mathcal{X}} d\mathbf{x} - \frac{\epsilon}{C}\Big]$.

Proof. Let us denote f^* to be the optimal function. We can write the inequality

$$\min_{\mathbf{D}} \min_{\mathbf{w}} \|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2 \leq \min_{\mathbf{D}} \min_{\mathbf{w}} \|f^* - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2, \quad (16)$$

using the fact that f^* is the optimal function ($f^* = \int_{\mathcal{X}} v(\mathbf{x}) \phi(\mathbf{x}) d\mathbf{x}$) and $\tilde{f}$ is the function represented by using a compressed dictionary. From Lemma 1, we have

$$\min_{\mathbf{D}} \min_{\mathbf{w}} \|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2 \leq \tilde{M} \min_{\mathbf{D}} \sum_{i=1}^{\tilde{M}} v_i^2 \kappa(\mathbf{x}_i, \mathbf{x}_i) - v_i^2 \mathbf{k}_i^T (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I})^{-1} \mathbf{k}_i. \quad (17)$$

Similar to (17), we can upper bound the right hand side of (16) as (17) but note here that f^* is the optimal function when infinite observation space $\mathcal{X}$ is considered. Thus, we can write the upper bound as

$$\min_{\mathbf{D}} \min_{\mathbf{w}} \|f^* - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2 \leq C \Big[\min_{\mathbf{D} \subset \mathcal{X}} \int_{\mathcal{X}} v(\mathbf{x})^2 \kappa(\mathbf{x}, \mathbf{x}) d\mathbf{x} - \int_{\mathcal{X}} v(\mathbf{x})^2 \mathbf{k}(\mathbf{x}, \mathbf{D})^T (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I})^{-1} \mathbf{k}(\mathbf{x}, \mathbf{D}) d\mathbf{x} \Big]. \quad (18)$$

where constant C is analogous to $\tilde{M}$ (see the inequality in (12)) in continuous domain. Thus using (16) and (18), we get:

$$\min_{\mathbf{D}} \min_{\mathbf{w}} \|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|_{\mathcal{H}}^2 \leq C \Big[\int_{\mathcal{X}} v(\mathbf{x})^2 \kappa(\mathbf{x}, \mathbf{x}) d\mathbf{x} - \min_{\mathbf{D} \subset \mathcal{X}} \int_{\mathcal{X}} v(\mathbf{x})^2 \mathbf{k}(\mathbf{x}, \mathbf{D})^T (\mathbf{K}_{\mathbf{D},\mathbf{D}} + \mu \mathbf{I})^{-1} \mathbf{k}(\mathbf{x}, \mathbf{D}) d\mathbf{x} \Big] \quad (19)$$

Now, lets consider normalized kernels, i.e., $\kappa(\mathbf{z}, \mathbf{z}) = 1$. If kernel vector $\mathbf{k}(\mathbf{x}, \mathbf{D})$ maps to a unique $\mathbf{k} \in [0, 1]^M$, and also

the function $\mathbf{k}(\cdot,\mathbf{D})^M$, then for $v(\mathbf{x})=1$, we can simplify the second integral term in (19) as [28],

$$\int_{\mathcal{X}} v(\mathbf{x})^2 \mathbf{k}(\mathbf{x},\mathbf{D})^T(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I})^{-1}\mathbf{k}(\mathbf{x},\mathbf{D})d\mathbf{x}$$
$$= \int \mathbf{k}^T(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I})^{-1}\mathbf{k}d\mathbf{k} = \det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I}). \quad (20)$$

The first integral term in (19) can now be simplified as

$$\int_{\mathcal{X}} v(\mathbf{x})^2\kappa(\mathbf{x},\mathbf{x})d\mathbf{x} = \int_{\mathcal{X}} d\mathbf{x} \quad (21)$$

using the fact that $\kappa(\mathbf{z},\mathbf{z})=1$ and $v(\mathbf{x})=1$.

Now using (21) and (20), we can write the right hand side of (19) as $C\Big[\int_{\mathcal{X}} d\mathbf{x} - \min_{\mathbf{D}\subset\mathcal{X}} \det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I})\Big]$. Now if we can have ensure that

$$C\Big[\int_{\mathcal{X}} d\mathbf{x} - \min_{\mathbf{D}\subset\mathcal{X}} \det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I})\Big] \leq \epsilon, \quad (22)$$

then we can say that $\min_{\mathbf{D}} \min_{\mathbf{w}} \|\tilde{f} - f_{\mathbf{D},\mathbf{w}}\|^2_{\mathcal{H}} \leq \epsilon$ from (19).

Re-ordering (22), we can write $\int_{\mathcal{X}} d\mathbf{x} - \epsilon/C \leq \min_{\mathbf{D}\subset\mathcal{X}} \det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I})$ and thereby we can also write:

$$\int_{\mathcal{X}} d\mathbf{x} - \epsilon/C \leq \min_{\mathbf{D}\subset\mathcal{X}} \det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I}) \leq \det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I}). \quad (23)$$

Taking log on both the sides of (23), we denote the resulting log terms on the left and right hand side of the inequality as

$$F(\mathbf{D}) := \log\det(\mathbf{K}_{\mathbf{D},\mathbf{D}}+\mu\mathbf{I}), \text{and } Q := \log\left[\int_{\mathcal{X}} d\mathbf{x} - \frac{\epsilon}{C}\right]. \quad (24)$$

Thus, using this definition (24), we get the inequality $F(\mathbf{D}) \geq Q$, thereby completing the proof. □

The *logdet* function is monotone and submodular [22], [24]. Thus, the inequality $F(\mathbf{D}) \geq Q$ leads us to formulating the functional submodular cover problem (7) and designing the greedy Algorithm 1 which ensures that we obtain a dictionary $\mathbf{D}$ satisfying the ϵ norm ball criterion, i.e., $\|f - \tilde{f}\|_{\mathcal{H}} \leq \epsilon$.

Next, we present the main result of the work, where we have generalized the algorithm in [18] for online non-parametric function learning settings. Theorem 1 encapsulates the result that using the Alg. 1 will yield a solution that picks at most logarithmically more elements than the optimal one.

Theorem 1. *Considering the submodular and monotone function F, for every iteration t, the dictionary obtained by Alg. 1 will guarantee that we satisfy the approximation budget ϵ, and also characterizes the cardinality of the dictionary obtained from Alg. 1 in comparison to the cardinality of dictionary obtained from the optimal algorithm as:*

$$M_{t+1} \leq (1+\log\min\{C_1,C_2\})\ M^{opt}_{t+1} \quad (25)$$

where $M_{t+1} = |\mathbf{D}_{t+1}|$ and $M^{opt}_{t+1} = |\mathbf{D}^{opt}_{t+1}|$, and C_1, C_2 are problem parameters.

Thm. 1 suggests that the logarithmic approximations will yield a model complexity which will not be very far from the optimal one. This is the first of its kind result in the non-parametric function learning literature. The important fact to note here is that, we can't establish such bounds (25) using destructive variants of KOMP algorithm proposed in [1], [2], and thus the destructive approach lacks the optimal model complexity guarantees as given in (25). For space restrictions, we have deferred the proof of Thm. 1 to [29].

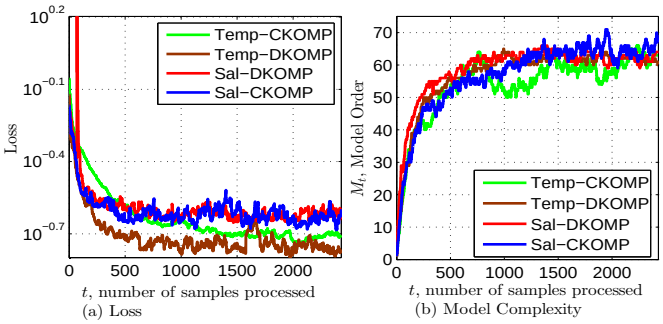

Fig. 1: In Fig. 1(a) we show the convergence of the loss function and in Fig. 1(b) we show the model order growth for temperature and salinity data.

IV. Numerical Experiments

In this section, we present the numerical performance of the constructive algorithm in comparison to the destructive algorithm on a real world ocean data obtained from the Gulf of Mexico [3] for estimating temperature and salinity at varying depths. In practice to implement Alg. 1 is difficult, since the value of Q is unknown because we can't find out the value of the integral term and the constant C inside the log term in Q in (24). However, we can still establish the efficacy of the constructive approach, by using the constructive version (denoted as "CKOMP") of the destructive variant (denoted as "DKOMP") of the KOMP [27] algorithm proposed in [1] and compare them. For the above comparison, we use the functional learning framework of [1]. The objective of this comparison is to show that the constructive algorithm also chooses the most favourable set of points and approximates the function closely as the destructive variant does.

We solve problem (1) by minimizing the regularized quadratic loss over function f using both CKOMP and DKOMP. Thus we predict the statistical mean of the temperature and salinity fields at varying depths and compare it with the real data. We consider a Gaussian kernel set at bandwidth 50 and the regularizer $\lambda = 10^{-5}$. The step-size η and the parsimony constant P (measures the compression budget: $\epsilon = P\eta^{3/2}$) for both the algorithms are set such that both have same model complexity. The parameters for using CKOMP algorithm for estimating temperature (denoted as "Temp-CKOMP") are $\eta = 0.05$, $P = 0.15$, and for estimating salinity (denoted as "Sal-CKOMP") are $\eta = 0.15$, $P = 0.03$. Similarly, the parameters for using DKOMP algorithm for estimating temperature (denoted as "Temp-DKOMP") are $\eta = 0.13$, $P = 3.5$, and for estimating salinity (denoted as "Sal-DKOMP") are $\eta = 0.135$, $P = 3.5$. Thus, it can be observed from Fig.1(a) that for comparable model order growth (see Fig.1(b)), we have approximately same convergence of objective function, thereby validating the effectiveness of the constructive approach of the algorithm. Hence, the performance of the constructive approach is comparable to the destructive algorithm. Thus, the constructive approach allows us to obtain the bound on model complexity relating it with the optimal model complexity (Theorem 1) and also gives favourable performance (see Fig.1) for both the temperature and salinity fields when compared with destructive algorithm.

REFERENCES

[1] A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "Parsimonious online learning with kernels via sparse projections in function space," *The Journal of Mach. Learn. Research*, vol. 20, no. 1, pp. 83–126, 2019.

[2] A. Koppel, H. Pradhan, and K. Rajawat, "Consistent online gaussian process regression without the sample complexity bottleneck," *Statistics and Computing*, vol. 31, no. 6, pp. 1–18, 2021.

[3] T. P. Boyer, M. Biddle, M. Hamilton, A. V. Mishonov, C. Paver, D. Seidov, and M. . Zweng, "Gulf of mexico regional climatology (NCEI Accession 0123320)," *Version 1.1. NOAA National Centers for Environmental Inf.*.

[4] G. Kimeldorf and G. Wahba, "Some results on tchebycheffian spline functions," *J. Math. Anal. Appl.*, vol. 33, no. 1, pp. 82–95, 1971.

[5] V. Tikhomirov, "On the representation of continuous functions of several variables as superpositions of continuous functions of one variable and addition," in *Selected Works of AN Kolmogorov*. Springer, 1991.

[6] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," *Subseries of Lect. Notes in Comput. Sci. Edited by JG Carbonell and J. Siekmann*, p. 416, 2001.

[7] V. Norkin and M. Keyzer, "On stochastic optimization and statistical learning in reproducing kernel hilbert spaces by support vector machines (svm)," *Informatica*, vol. 20, no. 2, pp. 273–292, 2009.

[8] A. Shapiro, D. Dentcheva, and A. Ruszczyński, *Lectures on stochastic programming: modeling and theory*. SIAM, 2009.

[9] B. Dai, B. Xie, N. He, Y. Liang, A. Raj, M.-F. F. Balcan, and L. Song, "Scalable kernel methods via doubly stochastic gradients," in *Advances in Neural Information Processing Systems*, 2014, pp. 3041–3049.

[10] T. Le, V. Nguyen, T. D. Nguyen, and D. Phung, "Nonparametric budgeted stochastic gradient descent," in *Artificial Intelligence and Statistics*, 2016, pp. 654–572.

[11] Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm," vol. 52, no. 8, pp. 2275–2285, Aug 2004.

[12] C. Richard, J. C. M. Bermudez, and P. Honeine, "Online prediction of time series data with kernels," vol. 57, no. 3, pp. 1058–1067, 2009.

[13] J. Kivinen, A. J. Smola, and R. C. Williamson, "Online Learning with Kernels," vol. 52, pp. 2165–2176, August 2004.

[14] O. Dekel, S. Shalev-Shwartz, and Y. Singer, "The forgetron: A kernel-based perceptron on a fixed budget," in *Advances in Neural Information Processing Systems 18*. MIT Press, 2006, p. 259266. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=78226

[15] J. Zhu and T. Hastie, "Kernel Logistic Regression and the Import Vector Machine," *Journal of Computational and Graphical Statistics*, vol. 14, no. 1, pp. 185–205, 2005.

[16] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition," in *Proc. of the Asilomar Conf. on Signals, Sys. Comput.*, 1993.

[17] H. Pradhan, A. S. Bedi, A. Koppel, and K. Rajawat, "Adaptive kernel learning in heterogeneous networks," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 7, pp. 423–437, 2021.

[18] L. A. Wolsey, "An analysis of the greedy algorithm for the submodular set covering problem," *Combinatorica*, vol. 2, no. 4, pp. 385–393, 1982.

[19] R. K. Iyer and J. A. Bilmes, "Submodular optimization with submodular cover and submodular knapsack constraints," in *Advances in Neural Information Processing Systems*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., vol. 26. Curran Associates, Inc., 2013. [Online]. Available: https://proceedings.neurips.cc/paper/2013/file/a1d50185e7426cbb0acad1e6ca74b9aa-Paper.pdf

[20] A. Krause, H. B. McMahan, C. Guestrin, and A. Gupta, "Robust submodular observation selection." *Journal of Machine Learning Research*, vol. 9, no. 12, 2008.

[21] A. Guillory and J. Bilmes, "Interactive submodular set cover," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 2010, pp. 415–422.

[22] A. Norouzi-Fard, A. Bazzi, I. Bogunovic, M. El Halabi, Y.-P. Hsieh, and V. Cevher, "An efficient streaming algorithm for the submodular cover problem," *Advances in Neural Information Processing Systems*, vol. 29, pp. 4493–4501, 2016.

[23] S. Tschiatschek, R. K. Iyer, H. Wei, and J. A. Bilmes, "Learning mixtures of submodular functions for image collection summarization," in *Advances in neural information processing systems*, 2014, pp. 1413–1421.

[24] A. Krause and D. Golovin, "Submodular function maximization." *Tractability*, vol. 3, pp. 71–104, 2014.

[25] R. Wheeden and A. Zygmund, *Measure and Integral: An Introduction to Real Analysis*, ser. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 1977.

[26] K. Müller, T. Adali, K. Fukumizu, J. C. Principe, and S. Theodoridis, "Special issue on advances in kernel-based learning for signal processing," vol. 30, no. 4, pp. 14–15, 2013.

[27] P. Vincent and Y. Bengio, "Kernel matching pursuit," *Mach. Learn.*, vol. 48, no. 1, pp. 165–187, 2002.

[28] M. Schlegel, Y. Pan, J. Chen, and M. White, "Adapting kernel representations online using submodular maximization," in *International Conference on Machine Learning*. PMLR, 2017, pp. 3037–3046.

[29] H. Pradhan, A. Koppel, and K. Rajawat, "On submodular set cover problems for near-optimal online kernel basis selection," 2021. [Online]. Available: https://drive.google.com/file/d/1BIhpsn6knE1flKj_qnDlUcuLS8MNgzur/view