Mixed Variable Bayesian Optimization with Frequency Modulated Kernels

Changyong Oh[1] **Efstratios Gavves**[1] **Max Welling**[1,2]

[1]Informatics Institute, University of Amsterdam, Amsterdam, The Netherlands
[2]Qualcomm AI Research Netherlands, Amsterdam, The Netherlands

1 POSITIVE DEFINITE FM KERNELS

For a weighted undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with graph Laplacian $L(\mathcal{G}) = U\Lambda U^T$. Frequency modulating kernels are defined as

$$k((\mathbf{c}, v), (\mathbf{c}', v')\|\, \theta, \beta) = [\sum_{i=1}^{\|\,\mathcal{V}\,\|} [U]_{:,i} f(\lambda_i, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta)[U]_{:,i}]_{v,v'} \tag{1}$$

where $[U]_{:,i}$ are eigenvectors of $L(\mathcal{G})$ which are columns of U and $\lambda_i = [\Lambda]_{ii}$ are corresponding eigenvalues. $\mathbf{c}$ and $\mathbf{c}'$ are continuous variables in $\mathbb{R}^{D_c}$, $\theta \in \mathbb{R}^{D_c}$ is a kernel parameter similar to the lengthscales in the RBF kernel. $\beta \in \mathbb{R}$ is a kernel parameter from kernels derived from the graph Laplacian.

Theorem 1.1. *If $f(\lambda, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta)$ defines a positive definite kernel on $(\mathbf{c}, \mathbf{c}') \in \mathbb{R}^{D_c} \times \mathbb{R}^{D_c}$, then a FreMod kernel defined with such f is positive definite jointly on $(\mathbf{c}, v)$.*

Proof.

$$k((\mathbf{c}, v), (\mathbf{c}', v')\|\, \theta, \beta) = \Big[\sum_{i=1}^{\|\,\mathcal{V}\,\|} [U]_{:,i} f(\lambda_i, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta)[U]_{:,i} \Big]_{v,v'} = \sum_{i=1}^{\|\,\mathcal{V}\,\|} [U]_{v,i} f(\lambda_i, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta)[U]_{v',i} \tag{2}$$

Since a sum of positive definite(PD) kernels is PD, we prove PD of frequency modulating kernels by showing that $k_i((\mathbf{c}, v), (\mathbf{c}', v')\|\, \theta, \beta) = [U]_{v,i} f(\lambda_i, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta)[U]_{v',i}$ is PD.

Let us consider $\mathbf{a} \in \mathbb{R}^S$, $\mathcal{D} = \{(\mathbf{c}_1, v_1), \cdots, (\mathbf{c}_S, v_S)\}$, then

$$\begin{aligned}
&\mathbf{a}^T \begin{bmatrix} [U]_{v_1,i} f(\lambda_i, \|\, \mathbf{c}_1 - \mathbf{c}_1 \,\|_\theta|\beta)[U]_{v_1,i} & \cdots & [U]_{v_1,i} f(\lambda_i, \|\, \mathbf{c}_1 - \mathbf{c}_S \,\|_\theta|\beta)[U]_{v_S,i} \\ [U]_{v_2,i} f(\lambda_i, \|\, \mathbf{c}_2 - \mathbf{c}_1 \,\|_\theta|\beta)[U]_{v_1,i} & \cdots & [U]_{v_2,i} f(\lambda_i, \|\, \mathbf{c}_2 - \mathbf{c}_S \,\|_\theta|\beta)[U]_{v_S,i} \\ \vdots & \cdots & \vdots \\ [U]_{v_S,i} f(\lambda_i, \|\, \mathbf{c}_S - \mathbf{c}_1 \,\|_\theta|\beta)[U]_{v_1,i} & \cdots & [U]_{v_S,i} f(\lambda_i, \|\, \mathbf{c}_S - \mathbf{c}_S \,\|_\theta|\beta)[U]_{v_S,i} \end{bmatrix} \mathbf{a} \\
&= (\mathbf{a} \circ [U]_{:,i})^T \begin{bmatrix} f(\lambda_i, \|\, \mathbf{c}_1 - \mathbf{c}_1 \,\|_\theta \|\beta) & \cdots & f(\beta\lambda_i, \|\, \mathbf{c}_1 - \mathbf{c}_S \,\|_\theta|\beta) \\ f(\lambda_i, \|\, \mathbf{c}_2 - \mathbf{c}_1 \,\|_\theta \|\beta) & \cdots & f(\beta\lambda_i, \|\, \mathbf{c}_2 - \mathbf{c}_S \,\|_\theta|\beta) \\ \vdots & \cdots & \vdots \\ f(\lambda_i, \|\, \mathbf{c}_S - \mathbf{c}_1 \,\|_\theta \|\beta) & \cdots & f(\beta\lambda_i, \|\, \mathbf{c}_S - \mathbf{c}_S \,\|_\theta|\beta) \end{bmatrix} (\mathbf{a} \circ [U]_{:,i})
\end{aligned} \tag{3}$$

where $\circ$ is Hadamard(elementwise) product and $[U]_{:,i} = [[U]_{v_1,i}, \cdots, [U]_{v_S,i}]^T$.

By letting $\mathbf{a}' = \mathbf{a} \circ [U_i]_{\pi_i(v_:),n}$, since $f(\lambda_i, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta)$ is PD, we show that $k_i((\mathbf{c}, v), (\mathbf{c}', v')\|\, \theta, \beta) = u_{i,v} f(\lambda_i, \|\, \mathbf{c} - \mathbf{c}' \,\|_\theta \|\beta) u_{i,v'}$ is PD. □

2 NONNEGATIVE VALUED FM KERNELS

Theorem 2.1. *For a connected and undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with non-negative weights on edges, define a kernel $k(v, v') = [Uf(\Lambda)U^T]_{v,v'}$ where U and Λ are eigenvectors and eigenvalues of the graph Laplacian $L(\mathcal{G}) = U\Lambda U^T$. If f is any non-negative and strictly decreasing convex function on $[0, \infty)$, then $K(v, v') \geq 0$ for all $v, v' \in \mathcal{V}$.*

Proof. For a connected and weighted undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the graph Laplacian $L(G)$ has exactly one 0 eigenvalue and the corresponding eigenvector $1/\sqrt{D}[1, \cdots, 1]^T$ when $|\mathcal{V}| = D$.

We show that

$$\min_{v,v'} k_{\mathcal{G}}(v, v') = \min_{p,q=1,\cdots,D} [Uf(\Lambda)U^T]_{p,q} \geq 0 \tag{4}$$

for an arbitrary connected and weighted undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $|\mathcal{V}| = D$ and $L(\mathcal{G}) = U\Lambda U^T$.

For a connected graph, there is only one zero eigenvalue

$$0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_D \quad \text{where} \quad \lambda_i = [\Lambda]_{i,i} \tag{5}$$

and the corresponding eigenvector is given as

$$U_{1,q} = \frac{1}{\sqrt{D}} (q = 1, \cdots, D). \tag{6}$$

From the definition of eigendecomposition, we have

$$U^T U = UU^T = I. \tag{7}$$

Importantly, from the definition of the graph Laplacian

$$[U\Lambda U^T]_{p,q} \leq 0 \quad \text{when} \quad p \neq q. \tag{8}$$

For a given diagonal matrix Λ such that $0 = \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_D$ where $\lambda_i = [\Lambda]_{i,i}$, we solve the following minimization problem

$$\min_{[U]_{p,i},[U]_{q,i}} \frac{f(0)}{D} + \sum_{i=2}^{D} f(\lambda_i)[U]_{p,i}[U]_{q,i} \tag{9}$$

with the constraints

$$\sum_{i=2}^{D} \lambda_i [U]_{p,i}[U]_{q,i} \leq 0 (p \neq q), \quad \sum_{i=2}^{D} [U]_{p,i}^2 = \sum_{i=2}^{D} [U]_{q,i}^2 = 1 - \frac{1}{D}, \quad \sum_{i=2}^{D} [U]_{p,i}[U]_{q,i} = -\frac{1}{D} (p \neq q) \tag{10}$$

When $p = q$, eq.9 is nonnegative because f is nonnegative valued. From now on, we consider the case $p \neq q$.

Lagrange multiplier is given as

$$\begin{aligned} L_{KKT}([U]_{p,i}, [U]_{q,i}, \eta, a, b, c) = \frac{f(0)}{D} + \sum_{i=2}^{D} f(\lambda_i)[U]_{p,i}[U]_{q,i} + \eta\Big(\sum_{i=2}^{D} \lambda_i [U]_{p,i}[U]_{q,i}\Big) \\ + a\Big(\sum_{i=2}^{D} [U]_{p,i}^2 - (1 - \frac{1}{D})\Big) + b\Big(\sum_{i=2}^{D} [U]_{q,i}^2 - (1 - \frac{1}{D})\Big) + c\Big(\sum_{i=2}^{D} [U]_{p,i}[U]_{q,i} + \frac{1}{D}\Big) \end{aligned} \tag{11}$$

with $\eta \geq 0$.

The stationary conditions are given as

$$\frac{\partial L_{KKT}}{\partial [U]_{p,i}} = f(\lambda_i)[U]_{q,i} + \eta\lambda_i [U]_{q,i} + c[U]_{q,i} + 2a[U]_{p,i} = 0 \tag{12}$$

$$\frac{\partial L_{KKT}}{\partial [U]_{q,i}} = f(\lambda_i)[U]_{p,i} + \eta\lambda_i [U]_{p,i} + c[U]_{p,i} + 2b[U]_{q,i} = 0 \tag{13}$$

from which, we have

$$(f(\lambda_i) + \eta\lambda_i + c)[U]_{q,i} = -2a[U]_{p,i} \tag{14}$$
$$(f(\lambda_i) + \eta\lambda_i + c)[U]_{p,i} = -2b[U]_{q,i} \tag{15}$$

By using

$$\sum_{i=2}^{D} \frac{\partial L_{KKT}}{\partial [U]_{p,i}} [U]_{p,i} = \sum_{i=2}^{D} \frac{\partial L_{KKT}}{\partial [U]_{q,i}} [U]_{q,i} = 0 \tag{16}$$

we have $a = b$.

From eq.(14) and eq.(15), we get

$$((f(\lambda_i) + \eta\lambda_i + c)^2 - 4ab)[U]_{q,i} = 0 \tag{17}$$
$$((f(\lambda_i) + \eta\lambda_i + c)^2 - 4ab)[U]_{p,i} = 0 \tag{18}$$

If $i \in \{i|(f(\lambda_i) + \eta\lambda_i + c)^2 - 4ab \neq 0\}$, we have $[U]_{p,i} = [U]_{q,i} = 0$. On the other hand, if $(f(\lambda_i) + \eta\lambda_i + c)^2 - 4ab = 0$, then we have $f(\lambda_i) + \eta\lambda_i + c = -2a$ or $f(\lambda_i) + \eta\lambda_i + c = 2a$ because $a = b$.

We define three index sets

$$I_0 = \{i|(f(\lambda_i) + \eta\lambda_i + c)^2 - 4a^2 \neq 0\} \tag{19}$$
$$I_+ = \{i|f(\lambda_i) + \eta\lambda_i + c + 2a = 0\} - \{1\} \tag{20}$$
$$I_- = \{i|f(\lambda_i) + \eta\lambda_i + c - 2a = 0\} \tag{21}$$

from eq.(14) and eq.(15), we have

$$i \in I_0 \Rightarrow [U]_{p,i} = [U]_{q,i} = 0 \tag{22}$$
$$i \in I_+ \Rightarrow [U]_{p,i} = [U]_{q,i} \tag{23}$$
$$i \in I_- \Rightarrow [U]_{p,i} = -[U]_{q,i} \tag{24}$$

With these conditions, the constraints can be expressed as

$$\sum_{i_+ \in I_+} \lambda_{i_+} [U]_{p,i}^2 - \sum_{i_- \in I_-} \lambda_{i_-} [U]_{p,i}^2 \leq 0, \quad \sum_{i_+ \in I_+} [U]_{p,i_+}^2 = \frac{1}{2} - \frac{1}{D}, \quad \sum_{i_- \in I_-} [U]_{p,i_-}^2 = \frac{1}{2} \tag{25}$$

We divide cases according to the number of solutions $g(\lambda) = f(\lambda) + \eta\lambda$ can have. *i)* $f(\lambda) + \eta\lambda$ can have at most one solution, *ii)* $f(\lambda) + \eta\lambda$ may have two solutions. Note that $g(\lambda)$ is convex as sum of two convex functions. Since a convex function can have at most two zeros unless it is constantly zero, these two cases are exhaustive. When $\eta = 0$, $f(\lambda)$ is strictly decreasing function and, thus $g(\lambda)$ has at most one solution. Also, when $\eta \geq -f'(0) = \max_\lambda -f'(\lambda)$, $f'(\lambda) + \eta$ is positive except for $\lambda = 0$ and $g(\lambda)$ has at most one solution.

Case *i)* $f(\lambda) + \eta\lambda$ can have at most one solution. ($\eta = 0$ or $\eta \geq -f'(0) = \max_\lambda -f'(\lambda)$)

Let us denote λ^E the unique solution of $f(\lambda_i) + \eta\lambda_i + c + 2a = 0$ and λ^N the unique of $f(\lambda_i) + \eta\lambda_i + c - 2a = 0$.

Therefore $\lambda_{i_+} = \lambda^E, \forall i_+ \in I_+$ and $\lambda_{i_-} = \lambda^N, \forall i_- \in I_-$. The minimization objective becomes

$$\begin{aligned} \frac{f(0)}{D} + \sum_{i=2}^{D} f(\lambda_i)[U]_{p,i}[U]_{q,i} &= \frac{f(0)}{D} + f(\lambda^E) \sum_{i_+ \in I_+} [U]_{p,i}^2 - f(\lambda_N) \sum_{i_- \in I_-} [U]_{p,i}^2 \\ &= \frac{f(0)}{D} + \Big(\frac{1}{2} - \frac{1}{D}\Big) f(\lambda^E) - \frac{1}{2} f(\lambda^N) \end{aligned} \tag{26}$$

The inequality constraint becomes

$$\sum_{i=2}^{D} \lambda_i [U]_{p,i}[U]_{q,i} = \frac{f(0)}{D} + \lambda^E \sum_{i_+ \in I_+} [U]_{p,i_+}^2 - \lambda_N \sum_{i_- \in I_-} [U]_{p,i_-}^2 = \Big(\frac{1}{2} - \frac{1}{D}\Big)\lambda^E - \frac{1}{2}\lambda^N \leq 0 \tag{27}$$

Since $\lambda^E, \lambda^N \in \{\lambda_2, \cdots, \lambda_D\}$, there is maximum value with respect to the choice of λ^E, λ^N. We consider continuous relaxation of the minimization problem with respect to λ^E, λ^N. By showing that the objective is nonnegative when $\lambda^E \geq 0, \lambda^N \geq 0$, we prove our claim. When we consider continuous optimization problem over λ^E, λ^N, the minimum is obtained when the inequality constraint becomes equality constraints. If $\left(\frac{1}{2} - \frac{1}{D}\right)\lambda^E - \frac{1}{2}\lambda^N < 0$ by increasing λ^E by $\delta > 0$ so that $\left(\frac{1}{2} - \frac{1}{D}\right)(\lambda^E + \delta) - \frac{1}{2}\lambda^N = 0$, $f(\lambda^E)$ is decreased to $f(\lambda^E + \delta)$, thus the minimum is obtained when the inequality constraint is equality. When $\eta > 0$, the inequality constraint automatically becomes an equality constraint by the slackness condition of the Karush-Kuhn-Tucker conditions.

With the inequality condition the objective becomes

$$\frac{f(0)}{D} + \left(\frac{1}{2} - \frac{1}{D}\right)f(\lambda^E) - \frac{1}{2}f\left((1 - \frac{2}{D})\lambda^E\right) \tag{28}$$

taking derivative with respect to λ_E, we have

$$\left(\frac{1}{2} - \frac{1}{D}\right)\left(f'(\lambda^E) - f'((1 - \frac{2}{D})\lambda^E)\right) \tag{29}$$

By the convexity of f, the derivative is always nonnegative with respect to $\lambda^E \geq 0$.

Since

$$\lim_{\lambda_E \to 0} \frac{f(0)}{D} + \left(\frac{1}{2} - \frac{1}{D}\right)f(\lambda^E) - \frac{1}{2}f((1 - \frac{2}{D})\lambda^E) = 0 \tag{30}$$

The minimum is nonnegative.

Case *ii)* $f(\lambda) + \eta\lambda$ may have two solutions. $(0 < \eta < -f'(0) = \max_\lambda -f'(\lambda))$

By the slackness condition, the inequality constraint becomes an equality constraint. Since $f(\lambda) + \eta\lambda$ is convex, it has at most two solutions. Let us denote $\lambda_1^E < \lambda_2^E$ two solutions of $f(\lambda) + \eta\lambda + c + 2a = 0$ and $\lambda_1^N < \lambda_2^N$ two solutions of $f(\lambda) + \eta\lambda + c - 2a = 0$ Then

$$f(\lambda_1^E) + \eta\lambda_1^E + c + 2a = 0 \tag{31}$$
$$f(\lambda_2^E) + \eta\lambda_2^E + c + 2a = 0 \tag{32}$$
$$f(\lambda_1^N) + \eta\lambda_1^N + c - 2a = 0 \tag{33}$$
$$f(\lambda_2^N) + \eta\lambda_2^N + c - 2a = 0 \tag{34}$$

The objective becomes

$$\begin{aligned}\frac{f(0)}{D} &+ f(\lambda_1^E) \sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_1^E} [U]_{p,i_+}^2 + f(\lambda_2^E) \sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_2^E} [U]_{p,i_+}^2 \\ &- f(\lambda_1^N) \sum_{i_- \in I_- : \lambda_{i_-} = \lambda_1^N} [U]_{p,i_-}^2 - f(\lambda_2^N) \sum_{i_- \in I_- : \lambda_{i_-} = \lambda_2^N} [U]_{p,i_-}^2\end{aligned} \tag{35}$$

with the constraints

$$\sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_1^E} [U]_{p,i_+}^2 + \sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_2^E} [U]_{p,i_+}^2 = \frac{1}{2} - \frac{1}{D} \tag{36}$$

$$\sum_{i_- \in I_- : \lambda_i = \lambda_1^N} [U]_{p,i_-}^2 + \sum_{i_- \in I_- : \lambda_i = \lambda_2^N} [U]_{p,i_-}^2 = \frac{1}{2} \tag{37}$$

$$\lambda_1^E \sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_1^E} [U]_{p,i_+}^2 + \lambda_2^E \sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_2^E} [U]_{p,i_+}^2 - \lambda_1^N \sum_{i_- \in I_- : \lambda_{i_-} = \lambda_1^N} [U]_{p,i_-}^2 - \lambda_2^N \sum_{i_- \in I_- : \lambda_{i_-} = \lambda_2^N} [U]_{p,i_-}^2 = 0 \tag{38}$$

Let

$$A^E = \sum_{i_+ \in I_+ : \lambda_{i_+} = \lambda_1^E} [U]_{p,i_+}^2 \in [0, \frac{1}{2} - \frac{1}{D}], \quad A^N = \sum_{i_- \in I_- : \lambda_{i_-} = \lambda_1^N} [U]_{p,i_-}^2 \in [0, \frac{1}{2}] \tag{39}$$

Then the objective becomes

$$\frac{f(0)}{D} + f(\lambda_1^E)A^E + f(\lambda_2^E)(\frac{1}{2} - \frac{1}{D} - A^E) - f(\lambda_1^N)A^N - f(\lambda_2^N)(\frac{1}{2} - A^N) \tag{40}$$

Taking derivatives

$$\frac{\partial}{\partial A^E} \Rightarrow f(\lambda_1^E) - f(\lambda_2^E) > 0 \tag{41}$$

$$\frac{\partial}{\partial A^N} \Rightarrow -f(\lambda_1^N) + f(\lambda_2^N) < 0 \tag{42}$$

$$\tag{43}$$

Thus the minimum is obtained at the boundary point where $A^E = 0$ and $A^N = \frac{1}{2}$ which falls back to Case *i)* whose minimum is bounded below by zero.

□

Remark. Theorem 3.2 holds for weighted undirected graphs, that is, for any arbitrary graph with arbitrary symmetric nonnegative edge weights.

Remark. Note that in numerical simulations, you may observe small negative values ($\approx 10^{-7}$) due to numerical instability.

Remark. In numerical simulations, the convexity condition does not appear to be necessary for complete graphs where $\max_{p \neq q}[L(\mathcal{G})]_{p,q} < -\epsilon$ for some $\epsilon > 0$. For complete graphs, the convexity condition may be relaxed, at least, in a stochastic sense.

Corollary 2.1.1. *The random walk kernel derived from normalized Laplacian Smola and Kondor [2003] and the diffusion kernels Kondor and Lafferty [2002], the ARD diffusion kernel Oh et al. [2019] and the regularized Laplacian kernel Smola and Kondor [2003] derived from normalized and unnormalized Laplacian are all positive valued kernels.*

Proof. The condition that off-diagonal entries are nonpositive holds for both normalized and unnormalized graph Laplacian. Therefore for normalized graph Laplacian, the proof in the above theorem can be applied without modification. The positivity of kernel value also holds for kernels derived from normalized Laplacian as long as it satisfies the conditions in Thm.3.2. □

Remark. In numerical simulations with nonconvex functions and arbitrary connected and weighted undirected graphs, negative values easily occur. For example, the inverse cosine kernel Smola and Kondor [2003] does not satisfies the convexity condition and has negative values.

3 EXAMPLES OF FM KERNELS

In this section, we first review the definition of conditionally negative definite(CND) and relations between positive definite(PD). Utilizing relations between PD and CND and properties of PD and CND, we provide an example of a flexible family of frequency modulating functions.

Definition 3.1 (3.1.1 [Berg et al., 1984])**.** A symmetric function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a conditionally negative definite(CND) kernel if $\forall n \in \mathbb{N}$, $x_1, \cdots, x_n \in \mathcal{X}$ $a_1, \cdots, a_n \in \mathbb{R}$ such that $\sum_{i=1}^n a_i = 0$

$$\sum_{i,j=1}^{n} a_i k(x_i, x_j) a_j \leq 0 \tag{44}$$

Please note that CND requires the condition $\sum_{i=1}^n a_i = 0$.

Theorem 3.1 (3.2.2 [Berg et al., 1984])**.** *$K(x, x')$ is conditionally negative definite if and only if $e^{-tK(x,x')}$ is positive definite for all $t > 0$.*

As mentioned in p.75 [Berg et al., 1984], from Thm. 3.1, we have

Theorem 3.2. *$K(x, x')$ is conditionally negative definite and $K(x, x')) \geq 0$ if and only if $(t + K(x, x'))^{-1}$ is positive definite for all $t > 0$.*

Theorem 3.3 (3.2.10 [Berg et al., 1984]). *If $K(x, x')$ is conditionally negative definite and $K(x, x) \geq 0$, then $(K(x, x'))^a$ for $0 < a < 1$ and $\log K(x, x')$ are conditionally negative definite.*

Theorem 3.4 (3.2.13 [Berg et al., 1984]). *$K(x, x') = \|x - x'\|^p$ is conditionally negative definite for all $0 < p \leq 2$.*

Using above theorems, we provide a quite flexible family of frequency modulating functions

Proposition 1. *For $S \in (0, \infty)$, a finite measure μ on $[0, S]$ and μ-measurable $\tau : [0, S] \to [0, 2]$ and $\rho : [0, S] \to \mathbb{N}$,*

$$f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta) = \int_0^S \frac{1}{(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^{\tau(s)})^{\rho(s)}} \mu(ds) \tag{45}$$

is a frequency modulating function.

Proof. First we show that

$$f^{p,t}(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta) = \frac{1}{(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^t)^p} \tag{46}$$

is a frequency modulating function for $t \in (0, 2]$ and $p \in \mathbb{N}$.

Property **FM-P1** on $f^{p,t}$) $f^{p,t}(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ is positive valued and decreasing with respect to λ.

Property **FM-P2** on $f^{p,t}$) $\| \mathbf{c} - \mathbf{c}' \|_\theta$ is conditionally negative definite by Thm.3.4 Then by Thm.3.2, $\frac{1}{(1+\beta\lambda+\alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^t)}$ is positive definite with respect to $\mathbf{c}$ and $\mathbf{c}'$. Since the product of positive definite kernels is positive definite, $f^{p,t}(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ is positive definite.

Property **FM-P3** on $f^{p,t}$) Let $h^{p,t} = f^{p,t}(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta) - f^{p,t}(\lambda, \|\tilde{\mathbf{c}} - \tilde{\mathbf{c}}'\|_\theta | \alpha, \beta)$, then

$$\begin{aligned} h^{p,t}_\lambda &= \frac{\partial h^{p,t}}{\partial \lambda} = -p\beta \Big(\frac{1}{(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^t)^{p+1}} - \frac{1}{(1 + \beta\lambda + \alpha \|\tilde{\mathbf{c}} - \tilde{\mathbf{c}}'\|_\theta^t)^{p+1}} \Big) \\ h^{p,t}_{\lambda\lambda} &= \frac{\partial^2 h^{p,t}}{\partial \lambda^2} = p(p+1)\beta^2 \Big(\frac{1}{(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^t)^{p+2}} - \frac{1}{(1 + \beta\lambda + \alpha \|\tilde{\mathbf{c}} - \tilde{\mathbf{c}}'\|_\theta^t)^{p+2}} \Big) \end{aligned} \tag{47}$$

For $\| \mathbf{c} - \mathbf{c}' \|_\theta < \|\tilde{\mathbf{c}} - \tilde{\mathbf{c}}'\|_\theta$, $h > 0$, $h_\lambda < 0$ and $h_{\lambda\lambda} > 0$, therefore this satisfies the frequency modulation principle.

Now we show that

$$f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta) = \int_0^S \frac{1}{(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^{\tau(s)})^{\rho(s)}} \mu(ds) \tag{48}$$

satisfies all 3 conditions.

Property **FM-P1**) Trivial from the definition.

Property **FM-P2**) Since a measurable function can be approximated by simple functions [Folland, 1999], we approximate $f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ with following increasing sequence

$$\begin{aligned} f_n(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta) = & \sum_{i=1}^{2^n} \sum_{j=1}^{n} \frac{\mu(A_{i,j})}{(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^{\frac{i-1}{2^n} 2})^j} \\ & \text{where} \quad A_{i,j} = \{s | \frac{i-1}{2^n} 2 < \rho(s) \leq \frac{i}{2^n} 2, \tau(s) = j\} \end{aligned} \tag{49}$$

Each summand $\mu(A_{i,j})/(1 + \beta\lambda + \alpha \| \mathbf{c} - \mathbf{c}' \|_\theta^{\frac{i-1}{2^n} 2})^j$ is positive definite as shown above and sum of positive definite kernels is positive definite. Therefore, $f_n(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ is positive definite. Since the pointwise limit of positive definite kernels is a kernel [Fukumizu, 2010], we show that $f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ is positive definite.

Property **FM-P3**) If we show that $\frac{\partial}{\partial \lambda}$ and $\int \mu(ds)$ are interchangeable, from the Condition #3 on $f_{p,t}$, we show that $f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ satisfies the frequency modulating principle.

Let $h = f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta) - f(\lambda, \|\tilde{\mathbf{c}} - \tilde{\mathbf{c}}'\|_\theta | \alpha, \beta)$. There is a constant $A > 0$ such that

$$\Big| \frac{h^{\tau(s),\rho(s)}(\lambda + \delta) - h^{\tau(s),\rho(s)}(\lambda)}{\delta} \Big| < \Big| \frac{\partial h^{\tau(s),\rho(s)}}{\partial \lambda} \Big| + A < \Big| \frac{\partial h^{0,1}}{\partial \lambda} \Big| + A \tag{50}$$

For a finite measure, $\left|\frac{\partial h^{0,1}}{\partial \lambda}\right| + A$ is integrable. Therefore, $\frac{\partial}{\partial \lambda}$ and $\int \mu(ds)$ are interchangeable by dominated convergence theorem [Folland, 1999]. With the same argument, $\frac{\partial^2}{\partial \lambda^2}$ and $\int \mu(ds)$ are interchangeable.

Now, we have

$$h_\lambda = \frac{\partial h}{\partial \lambda} = \int_0^S \frac{\partial h^{\tau(s),\rho(s)}}{\partial \lambda} \mu(ds)$$
$$h_{\lambda\lambda} = \frac{\partial^2 h}{\partial \lambda^2} = \int_0^S \frac{\partial^2 h^{\tau(s),\rho(s)}}{\partial \lambda^2} \mu(ds)$$

From the Condition #3 on $f^{p,t}$, $h_\lambda < 0$ and $h_{\lambda\lambda} > 0$ follow and thus we show that $f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ satisfies the frequency modulating principle.

$f(\lambda, \| \mathbf{c} - \mathbf{c}' \|_\theta | \alpha, \beta)$ is a frequency modulating function. □

Proposition 2. *If $k_\mathcal{H} : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ on a RKHS $\mathcal{H}$ is bounded above by $u > 0$, then for any $\delta > 0$*

$$f(\lambda, k_\mathcal{H}(h, h') | \alpha, \beta) = \frac{1}{\delta + u + \beta\lambda - k_\mathcal{H}(h, h')} \tag{51}$$

is positive definite on $(h, h') \in \mathcal{H} \times \mathcal{H}$.

Proof. The negation of a positive definite kernel is conditionally negative definite by Supp. Def. 3.1. Also, by definition, a constant plus a conditionally negative definite kernel is conditionally negative definite. Therefore, $u - k_\mathcal{H}(h, h')$ is conditionally negative definite.

Using Supp. Thm.3.2, we show that $1/(\delta + u + \beta\lambda - k_\mathcal{H}(h, h'))$ is positive definite on $(h, h') \in \mathcal{H} \times \mathcal{H}$. □

4 EXPERIMENTAL DETAILS

In this section, we provide the details of each component of BO pipeline, the surrogate model and how it is fitted to evaluation data, the acquisition function and how it is optimized. We also provide each experiment specific details including the search spaces, evaluation detail, run time analysis and etc. The code used for the experiments will be released upon acceptance.

4.1 ACQUISITION FUNCTION OPTIMIZATION

We use Expected Improvement (EI) acquisition function [Donald, 1998]. Since, in mixed variable BO, acquisition function optimization is another mixed variable optimization task, we need a procedure to perform an optimization of acquisition functions on mixed variables.

Acquisition Function Optimization Similar to [Daxberger et al., 2019], we alternatively call continuous optimizer and discrete optimizer, which is similar to coordinate-wise ascent, and, in this case, it is so-called type-wise ascent. For continuous variables, we use L-BFGS-B [Zhu et al., 1997] and for discrete variables, we use hill climbing [Skiena, 1998]. Since the discrete part of the search space is represented by graphs, hill climbing is amount to greedy ascent in neighborhood. We alternate one discrete update using hill climbing call and one continuous update by calling $scipy.optimize.minimize(method = \text{"L-BFGS-B"}, maxiter = 1)$.

Spray Points Acquisition functions are highly multi-modal and thus initial points with which the optimization of acquisition functions starts have an impact on exploration-exploitation trade-off. In order to encourage exploitation, spray points [Snoek et al., 2012, Garnett et al., 2010, Oh et al., 2018], which are points in the neighborhood of the current optimum (e.g, optimum among the collected evaluations), has been widely used.

Initial points for acquisition function optimization On 50 spray points and 100000 randomly sampled points, acquisition values are computed, and the highest 40 are used as initial points to start acquisition function optimization.

4.2 JOINT OPTIMIZATION OF NEURAL ARCHITECTURE AND SGD HYPERPARAMETER

Discrete Part of the Search Space The discrete part of the search space, $\mathcal{A}$, is modified from the NASNet search space [Zoph and Le, 2016]. Each block consists of 4 states S_1, S_2, S_3, S_4 and takes two inputs S_{-1}, S_0 from a previous block. For each state, two inputs are chosen from the previous states, Then two operations are chosen and the state finishes its process by summing up two results of the chosen operation For example, if two inputs S_{-1}, S_2 and two operations $OP_3^{(1)}, OP_3^{(2)}$ are chosen for S_3, we have $(S_{-1}, S_2) \xrightarrow{S_3} OP_3^{(1)}(S_{-1}) + OP_3^{(2)}(S_2)$.

Operations are chosen from 8 types below

- ID
- Conv3×3
- Separable Conv3×3
- Max Pooling3×3
- Conv1×1
- Conv5×5
- Separable Conv5×5
- Max Pooling5×5

Two inputs for each state are chosen from states with smaller subscript(e.g S_i is allowed to have S_j as an input if $j < i$). By choosing S_4 and one of S_1, S_2, S_3 as outputs of the block, the configuration of a block is completed.

In MODLAP, it is required to specify graphs for discrete variables. For graphs representing operation types, we use complete graphs. For graphs representing inputs of each states, we use graphs which reflect the ordering structure. In a graph representing inputs of each state, each vertex is represented by a tuple, for the graph representing inputs of S_3, it has a vertex set of $\{(-1,0),(-1,1),(-1,2),(0,1),(0,2),(1,2)\}$. For example, choosing $(-1,0)$ means S_3 takes S_{-1}(input 1 of the block) and S_0(input 2 of the block) as inputs of the cell and choosing $(0,2)$ means S_3 takes S_0(input 2 of the block) and S_2(cell 2) as inputs. There exists an edge between vertices as long as one input is shared and two distinct inputs differ by one. For example, there is an edge between $(-1,0)$ and $(-1,1)$ because -1 is shared and $|0-1| = 1$ and there is no edge between $(-1,0)$ and $(-1,2)$ because $|0-1| \neq 1$ even though -1 is shared. Note that in the graph representing inputs for S_4, we exclude the vertex $(-1,0)$ to avoid the identity block. For graphs representing outputs of the block, we use the path graph with 3 vertices since we restrict the output is one of $(1,4),(2,4),(3,4)$. By defining graphs corresponding variables in this way, a prior knowledge about the search space can be infused and be of help to Bayesian optimization.

Continuous Part of the Search Space The space of continuous hyperparameters $\mathcal{H}$ comprises 6 continuous hyperparameters of the SGD with a learning rate scheduler: learning rate, momentum, weight decay, learning rate reduction factor, 1st reduction point ratio1 and 2nd reduction point ratio. The ranges for each hyperparameter are given in Supp. Table 1.

Table 1: SGD Hyperparameter Range

SGD hyperparameter	Transformation	Range
Learning Rate	log	$[\log(0.001), \log(0.1)]$
Momentum	·	$[0.8, 1.0]$
Weight Decay	log	$[\log(10^{-6}), \log(10^{-2})]$
Learning Rate Reduction Factor	·	$[0.1, 0.9]$
1st Reduction Point Ratio	·	$[0, 1]$
2nd Reduction Point Ratio	·	$[0, 1]$

For a given learning rate l, learning rate reduction factor γ, 1st reduction point ratio r_1 and 2nd reduction point ratio r_2, then learning rate scheduling is given in Supp. Table 2.

Table 2: Learning Rate Scheduling. In the experiment, the number of epochs E is set to 25.

Begin Epoch($<$)	($\leq$)End Epoch	Learning Rate
0	$E \times r_1$	l
$E \times r_1$	$E \times (r_1 + (1-r_1)r_2)$	$l \cdot \gamma$
$E \times (r_1 + (1-r_1)r_2)$	E	$l \cdot \gamma^2$

Evaluation For a given block configuration $a \in \mathcal{A}$, the model is built by stacking 3 blocks with downsampling between blocks. Note that there are two inputs and two outputs of the blocks. Therefore, the downsampling is applied separately to each output. The two outputs of the last block are concatenated after max pooling and then fed to the fully connected layer.

The model is trained with the hyperparameter $h \in \mathcal{H}$ on a half of FashionMNIST [Xiao et al., 2017] training data for 25 epochs and the validation error is computed on the rest half of training data. To reduce the high noise in validation error, the validation error is averaged over 4 validation errors from models trained with different random initialization. With the batch size of 32, each evaluation takes 12$\sim$21 minutes on a single GTX 1080 Ti depending on architectures

Regularized Evolution Hyperparameters RE has hyperparameters, the population size and the sample size. We set to 50 and 15, respectively, to make those similar to the optimal choice in [Real et al., 2019, Oh et al., 2019]. Accordingly, RE starts with a population with 50 random initial points. In each run of 4 runs, the first 10 initial points of 50 random initial points are shared with 10 initial points used in GP-BO.

Another hyperparameter is the mutation rule. In addition to the mutation of architectures used in [Real et al., 2019], for continuous variables, a randomly chosen single continuous variable is mutated by Gaussian noise with small variance. In each round, one continuous variable and one discrete variable are altered.

Wall-clock Run Time The total run time of MODLAP(200), 61.44 ± 4.09 hours, is sum of 9.27 ± 2.60 hours for BO suggestions and 52.16 ± 1.79 hours for evaluations. BO suggestions were run on Intel Xeon Processor E5-2630 v3 and evaluations were run on GTX 1080 Ti.

In the actual execution of RE, two different types of GPUs were used, GTX 1080 Ti(fast) and GTX 980(slow). Therefore, the evaluation time for RE is estimated by assuming that RE were also run on GTX 1080 Ti(fast) only. During the total run time of MODLAP(200), 61.44 ± 4.09 hours, RE is estimated to collects 230 evaluations. $230 \approx 61.44/52.16 \times (200 - 10) + 10$ where 10 is adjusted because the evaluation time for 10 random initial points was not measured.

Since in both RE and BOHB, we assume zero seconds to acquire new hyperparameters and only consider times spent for evaluations, the wall-clock runtime of BOHB is estimated to be equal to wall-clock runtime of RE.

5 EXPERIMENT: RESULTS

In this section, in addition to the results reported in Sec. 5, we provide additional results.

On 3 synthetic problems and 2 hyperparameter optimization problems, along with the frequency modulation, we also compare other kernel combinations such as the kernel addition and the kernel product as follows.

PRODLAP : $k_{RBF} \times k_{Lap}$	ADDLAP : $k_{RBF} + k_{Lap}$	MODLAP : Eq.5 with $f = f_{Lap}$
PRODDIF : $k_{RBF} \times k_{Dif}$	ADDDIF : $k_{RBF} + k_{Dif}$	MODDIF : Eq.5 with $f = f_{Dif}$

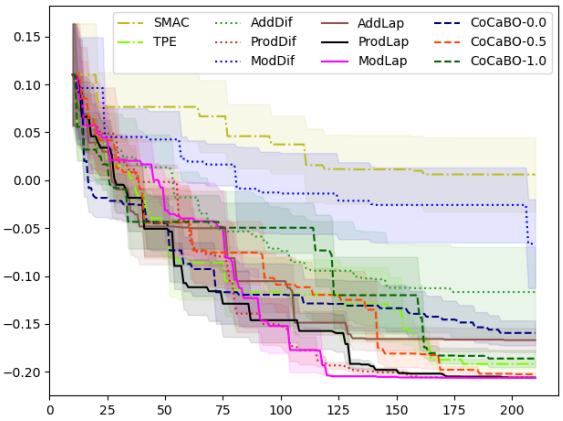
where k_{RBF} is the RBF kernel and

$$k_{Lap}(\mathbf{v}, \mathbf{v}') = \prod_{p=1}^{P} \sum_{i=1}^{|\mathcal{V}_p|} [U^p]_{v_p,i} \frac{1}{1 + \beta_p \lambda_i^p} [U^p]_{v'_p,i} \quad k_{Dif}(\mathbf{v}, \mathbf{v}') = \prod_{p=1}^{P} \sum_{i=1}^{|\mathcal{V}_p|} [U^p]_{v_p,i} \exp(-\beta_p \lambda_i^p) [U^p]_{v'_p,i} \tag{52}$$

We make following observations with this additional comparison. Firstly, MODDIF which does not respect the similarity measure behavior, sometimes severely degrades BO performance. Secondly, the kernel product often performs better than the kernel addition. Thirdly, MODLAP shows the equally good final results as the kernel product and finds the better solution faster than the kernel product consistently. This can be clearly shown by comparing the area above the mean curve of BO runs using different kernels. The area above the mean curve of BO using MODLAP is larger than the are above the mean curve of BO using the kernel product. Moreover, the gap between the area from MODLAP and the area from kernel product increases in problems with larger search spaces. Even on the smallest search space, Func2C, MODLAP lags behind the kernel product up to around 90th evaluation and outperforms after it. The benefit of MODLAP modeling complex dependency among mixed variables is more prominent in higher dimension problems.

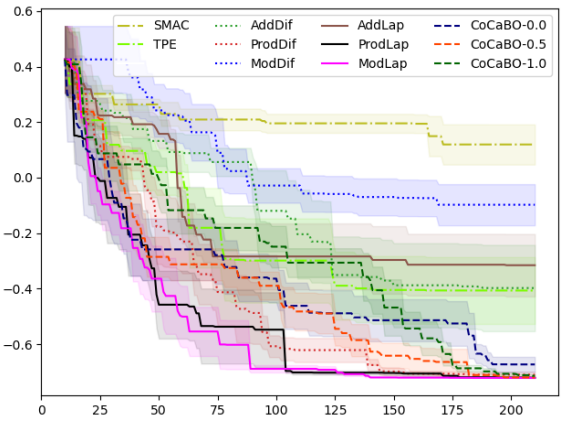
On the joint optimization of SGD hyperparameters and architecture, we show the additional result where RE and BOHB are continued 600 evaluations.

5.1 FUNC2C



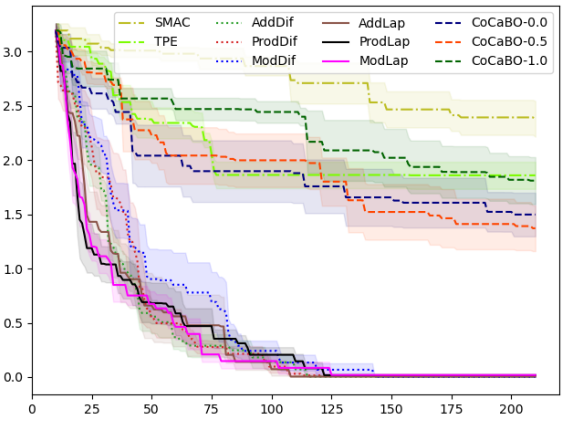
Method	Mean±Std.Err.
SMAC	$+0.0060 \pm 0.0387$
TPE	-0.1917 ± 0.0053
AddDif	-0.1167 ± 0.0472
ProdDif	-0.2060 ± 0.0002
ModDif	-0.0662 ± 0.0463
AddLap	-0.1669 ± 0.0127
ProdLap	-0.2060 ± 0.0001
ModLap	-0.2063 ± 0.0000
CoCaBO-0.0	-0.1594 ± 0.0130
CoCaBO-0.5	-0.2025 ± 0.0018
CoCaBO-1.0	-0.1861 ± 0.0090

5.2 FUNC3C



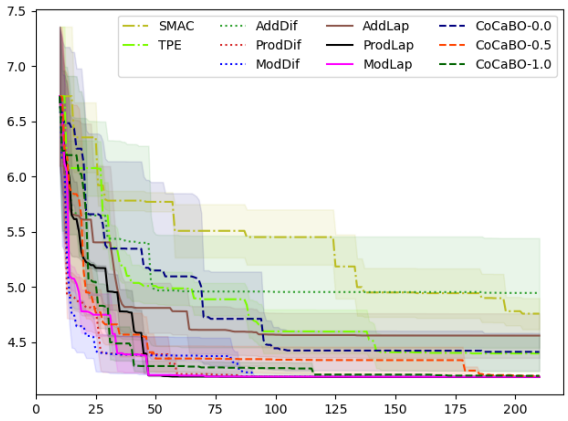
Method	Mean±Std.Err.
SMAC	$+0.1194 \pm 0.0723$
TPE	-0.4068 ± 0.1204
AddDif	-0.3979 ± 0.1555
ProdDif	-0.7100 ± 0.0106
ModDif	-0.0977 ± 0.0742
AddLap	-0.3156 ± 0.1125
ProdLap	-0.7213 ± 0.0005
ModLap	-0.7215 ± 0.0004
CoCaBO-0.0	-0.6730 ± 0.0274
CoCaBO-0.5	-0.7202 ± 0.0016
CoCaBO-1.0	-0.7139 ± 0.0051

5.3 ACKLEY5C



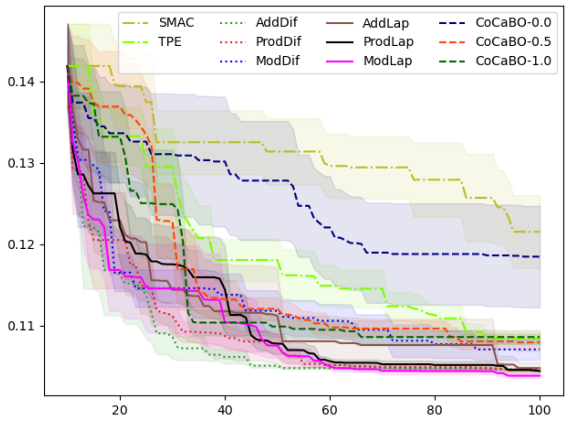
Method	Mean±Std.Err.
SMAC	$+2.3809 \pm 0.1648$
TPE	$+1.8601 \pm 0.1248$
AddDif	$+0.0040 \pm 0.0015$
ProdDif	$+0.0152 \pm 0.0044$
ModDif	$+0.0008 \pm 0.0003$
AddLap	$+0.0042 \pm 0.0018$
ProdLap	$+0.0177 \pm 0.0038$
ModLap	$+0.0186 \pm 0.0057$
CoCaBO-0.0	$+1.4986 \pm 0.2012$
CoCaBO-0.5	$+1.3720 \pm 0.2110$
CoCaBO-1.0	$+1.8114 \pm 0.2168$

5.4 SVM HYPERPARAMETER OPTIMIZATION



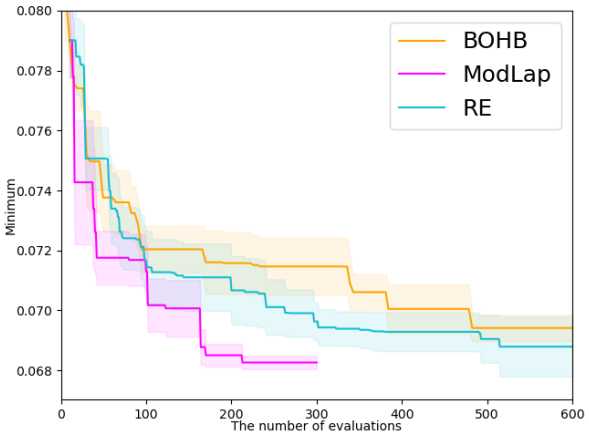
Method	Mean±Std.Err.
SMAC	$+4.7588 \pm 0.1414$
TPE	$+4.3986 \pm 0.1632$
AddDif	$+4.9463 \pm 0.4960$
ProdDif	$+4.1857 \pm 0.0017$
ModDif	$+4.1876 \pm 0.0012$
AddLap	$+4.5600 \pm 0.2014$
ProdLap	$+4.1856 \pm 0.0012$
ModLap	$+4.1864 \pm 0.0015$
CoCaBO-0.0	$+4.4122 \pm 0.1703$
CoCaBO-0.5	$+4.1957 \pm 0.0040$
CoCaBO-1.0	$+4.1958 \pm 0.0037$

5.5 XGBOOST HYPERPARAMETER OPTIMIZATION



Method	Mean±Std.Err.
SMAC	$+0.1215 \pm 0.0045$
TPE	$+0.1084 \pm 0.0007$
AddDif	$+0.1046 \pm 0.0001$
ProdDif	$+0.1045 \pm 0.0003$
ModDif	$+0.1071 \pm 0.0013$
AddLap	$+0.1048 \pm 0.0007$
ProdLap	$+0.1044 \pm 0.0001$
ModLap	$+0.1038 \pm 0.0003$
CoCaBO-0.0	$+0.1184 \pm 0.0062$
CoCaBO-0.5	$+0.1079 \pm 0.0010$
CoCaBO-1.0	$+0.1086 \pm 0.0008$

5.6 JOINT OPTIMIZATION OF SGD HYPERPARAMETERS AND ARCHITECTURE.



Method(#Eval.)	Mean±Std.Err.
BOHB(200)	$7.158 \times 10^{-2} \pm 1.0303 \times 10^{-3}$
BOHB(230)	$7.151 \times 10^{-2} \pm 9.8367 \times 10^{-4}$
BOHB(600)	$6.941 \times 10^{-2} \pm 4.4320 \times 10^{-4}$
RE(200)	$7.067 \times 10^{-2} \pm 1.1417 \times 10^{-3}$
RE(230)	$7.061 \times 10^{-2} \pm 1.1329 \times 10^{-3}$
RE(400)	$6.929 \times 10^{-2} \pm 6.4804 \times 10^{-4}$
RE(600)	$6.879 \times 10^{-2} \pm 1.0039 \times 10^{-3}$
ModLap(200)	$6.850 \times 10^{-2} \pm 3.7914 \times 10^{-4}$
ModLap(230)	$6.826 \times 10^{-2} \pm 2.2317 \times 10^{-4}$
ModLap(300)	$6.826 \times 10^{-2} \pm 2.2317 \times 10^{-4}$

6 EXPERIMENT: ABLATION STUDY

We run a regression task on 3 different UCI datasets.

Table 3: Regression Datasets

Dataset	# of points	Continuous Dim.	Categorical Dim.
Meta-data	528	16	3
Servo	167	2	2
Optical Intercon. Net.	640	2	2

On 20 different random splits (training:test=8:2), negative log likelihood(NLL) and RMSE on test set are reported in Table 4.

Table 4: Regression

NLL	Meta-data	Servo	Optical Intercon. Net.
AddDif	16.0224 ± 3.9906	4.2362 ± 0.6115	7.5504 ± 0.4867
ProdDif	9.5198 ± 3.7116	0.9579 ± 0.4758	0.2132 ± 0.2050
ModDif	5.9377 ± 1.9872	503.9973 ± 486.4679	10.0005 ± 0.2934
AddLap	1.6805 ± 0.1847	3.7083 ± 0.5001	7.5568 ± 0.4897
ProdLap	1.3236 ± 0.3539	0.7008 ± 0.3385	0.2135 ± 0.1928
ModLap	1.1218 ± 0.2987	1.0790 ± 0.4607	0.1521 ± 0.2265
RMSE	**Meta-data**	**Servo**	**Optical Intercon. Net.**
AddDif	1.0223 ± 0.1601	0.5696 ± 0.0310	0.2577 ± 0.0052
ProdDif	1.1537 ± 0.1654	0.3023 ± 0.0408	0.1413 ± 0.0060
ModDif	1.4074 ± 0.2027	0.7308 ± 0.0910	0.7881 ± 0.0069
AddLap	1.0199 ± 0.1588	0.5709 ± 0.0311	0.2577 ± 0.0052
ProdLap	1.0898 ± 0.1642	0.2971 ± 0.0405	0.1417 ± 0.0059
ModLap	1.0920 ± 0.1626	0.3046 ± 0.0412	0.1400 ± 0.0063

In terms of NLL, which takes into account uncertainty, ModLap is the best in Meta-data and Optical Intercon. Net. In Servo, ProdLap/ProdDif perform the best, so we conjecture that this dataset has an approximate product structure. In terms of RMSE, ModLap and ProdLap are equally good. We conclude that the frequency modulation has the benefit beyond the addition/product of good basis kernels. Also, the importance of respecting the similarity measure behavior is observed on the regression task.

References

Christian Berg, Jens Peter Reus Christensen, and Paul Ressel. *Harmonic analysis on semigroups: theory of positive definite and related functions*, volume 100. Springer, 1984.

Erik Daxberger, Anastasia Makarova, Matteo Turchetta, and Andreas Krause. Mixed-variable bayesian optimization. *arXiv preprint arXiv:1907.01329*, 2019.

R Jones Donald. Efficient global optimization of expensive black-box function. *J. Global Optim.*, 13:455–492, 1998.

Gerald B Folland. *Real analysis: modern techniques and their applications*. Wiley, 1999.

Kenji Fukumizu. Kernel method: Data analysis with positive definite kernels. *Graduate University of Advanced Studies*, 2010.

Roman Garnett, Michael A Osborne, and Stephen J Roberts. Bayesian optimization for sensor set selection. In *Proceedings of the 9th ACM/IEEE international conference on information processing in sensor networks*, pages 209–219, 2010.

Risi Imre Kondor and John Lafferty. Diffusion kernels on graphs and other discrete structures. In *ICML*, 2002.

ChangYong Oh, Efstratios Gavves, and Max Welling. Bock: Bayesian optimization with cylindrical kernels. In *ICML*, pages 3868–3877, 2018.

Changyong Oh, Jakub Tomczak, Efstratios Gavves, and Max Welling. Combinatorial bayesian optimization using the graph cartesian product. In *NeurIPS*, pages 2910–2920, 2019.

Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pages 4780–4789, 2019.

Steven S Skiena. *The algorithm design manual: Text*, volume 1. Springer Science & Business Media, 1998.

Alexander J Smola and Risi Kondor. Kernels and regularization on graphs. In *Learning theory and kernel machines*, pages 144–158. Springer, 2003.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *NeurIPS*, pages 2951–2959, 2012.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560, 1997.

Barret Zoph and Quoc V Le. Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1611.01578*, 2016.