Learning Deep Kernel Networks: Application to Efficient and Robust Structured Prediction

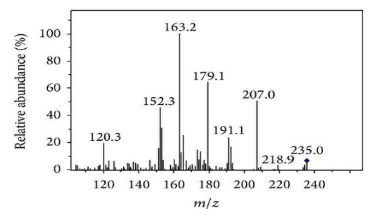
PhD Defense, Tamim El Ahmad

July 9th, 2024

Télécom Paris

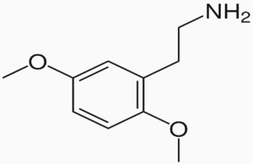
Structured prediction

Emblematic example of metabolite identification (Brouard et al., 2016a; Schymanski et al., 2017):



x

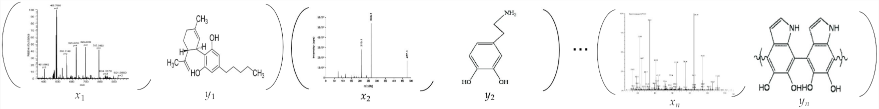
$$\xrightarrow{f\ =\ ?}$$



y

Structured prediction in supervised settings

Supervised settings: n i.i.d. training sample $(x_i, y_i)_{i=1}^{n} \in (\mathcal{X}, \mathcal{Y})^n \sim \rho$

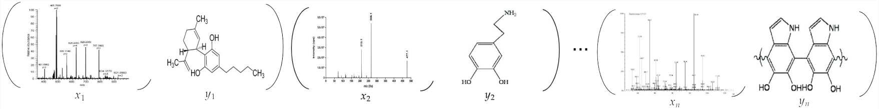


Given a loss function $\Delta : \mathcal{Y}^2 \to \mathbb{R}$

$$f^* = \underset{f:\mathcal{X}\to\mathcal{Y}}{\arg\inf}\ \mathbb{E}_{(x,y)\sim\rho}[\Delta(f(x), y)] \approx \underset{f:\mathcal{X}\to\mathcal{Y}}{\arg\inf}\ \frac{1}{n}\sum_{i=1}^{n}\Delta(f(x_i), y_i) = \hat{f}$$

Structured prediction in supervised settings

Supervised settings: n i.i.d. training sample $(x_i, y_i)_{i=1}^{n} \in (\mathcal{X}, \mathcal{Y})^n \sim \rho$



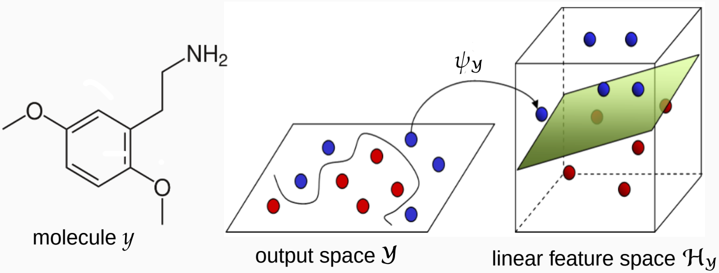
Given a loss function $\Delta : \mathcal{Y}^2 \to \mathbb{R}$

$$f^* = \operatorname*{arg\,inf}_{f:\mathcal{X}\to\mathcal{Y}} \mathbb{E}_{(x,y)\sim\rho}[\Delta(f(x), y)] \approx \operatorname*{arg\,inf}_{f:\mathcal{X}\to\mathcal{Y}} \frac{1}{n}\sum_{i=1}^{n} \Delta(f(x_i), y_i) = \hat{f}$$

How to design a loss Δ taking into account the structure of $\mathcal{Y}$?

Kernel methods: output representation

Linear method after embedding through feature map $\psi_{\mathcal{Y}} : \mathcal{Y} \to \mathcal{H}_{\mathcal{Y}}$:
choice of kernel $\iff$ choice of representation


NH2
O
O
$\psi_{\mathcal{Y}}$
molecule y
output space $\mathcal{Y}$
linear feature space $\mathcal{H}_{\mathcal{Y}}$

$\langle \psi_{\mathcal{Y}}(y), \psi_{\mathcal{Y}}(y') \rangle_{\mathcal{H}_{\mathcal{Y}}} = k_{\mathcal{Y}}(y, y')$: relevant similarity measure over $\mathcal{Y}$

Output Kernel Regression for structured prediction

$$\implies \boldsymbol{\Delta}(y, y') = \|\boldsymbol{\psi_{\mathcal{Y}}}(y) - \boldsymbol{\psi_{\mathcal{Y}}}(y')\|^2_{\boldsymbol{\mathcal{H}_{\mathcal{Y}}}} = 2 - 2\boldsymbol{k_{\mathcal{Y}}}(y, y')$$

($\forall\, y \in \mathcal{Y}$, $\|\boldsymbol{\psi_{\mathcal{Y}}}\|_{\boldsymbol{\mathcal{H}_{\mathcal{Y}}}} = 1$ without loss of generality)

Versatility: tackle various tasks through an appropriate choice of $\boldsymbol{\psi_{\mathcal{Y}}}$ (e.g. SOTA performance on metabolite identification (Brouard et al., 2016a) and label ranking (Korba et al., 2018) datasets)

Output Kernel Regression: a surrogate approach

Surrogate (2-step) method (Weston et al., 2003; Cortes et al., 2005; Brouard et al., 2011; Kadri et al., 2013):

1. $\hat{h} = \underset{h:\mathcal{X}\to\boldsymbol{\mathcal{H}_\mathcal{Y}}}{\arg\min}\ \frac{1}{n}\sum_{i=1}^{n}\|h(x_i) - \boldsymbol{\psi_\mathcal{Y}}(y_i)\|^2_{\boldsymbol{\mathcal{H}_\mathcal{Y}}}$ (training step)
2. $\hat{f}(x) = d \circ \hat{h}(x) = \underset{y\in\mathcal{Y}}{\arg\min}\ \|\hat{h}(x) - \boldsymbol{\psi_\mathcal{Y}}(y)\|^2_{\boldsymbol{\mathcal{H}_\mathcal{Y}}}$ (inference step)

$$\begin{tikzcd}
 & \mathcal{H}_\mathcal{Y} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur, "\hat{h}"] \arrow[r, "\hat{f} = d \circ \hat{h}"'] & \mathcal{Y} \arrow[u, "\psi_\mathcal{Y}"']
\end{tikzcd}$$

Theoretical guarantees: for measurable $h : \mathcal{X} \to \boldsymbol{\mathcal{H}_\mathcal{Y}}$ and $f = d \circ h$, **f's excess risk is bounded by h's excess risk** (Ciliberto et al., 2020)

Output Kernel Regression: linear estimator

$$\hat{h} : x \mapsto \sum_{i=1}^{n} \hat{\boldsymbol{\alpha}}(x)_i \boldsymbol{\psi}_{\boldsymbol{\mathcal{Y}}}(y_i)$$

where $\hat{\boldsymbol{\alpha}} : \mathcal{X} \to \mathbb{R}^n$ usually obtained by **non-parametric methods** (e.g. **input kernel $\boldsymbol{k}_{\boldsymbol{\mathcal{X}}}$** (**Input Output Kernel Regression**) (Brouard et al., 2016b), input tree (Geurts et al., 2006))

Output Kernel Regression: linear estimator

$$\hat{h} : x \mapsto \sum_{i=1}^{n} \hat{\boldsymbol{\alpha}}(x)_i \boldsymbol{\psi}_{\boldsymbol{\mathcal{Y}}}(y_i)$$

where $\hat{\boldsymbol{\alpha}} : \mathcal{X} \to \mathbb{R}^n$ usually obtained by **non-parametric methods** (e.g. **input kernel $k_{\boldsymbol{\mathcal{X}}}$ (Input Output Kernel Regression)** (Brouard et al., 2016b), input tree (Geurts et al., 2006))

$$\begin{tikzcd}
 & \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left=60, "d"] \\
\mathcal{X} \arrow[ru, "\hat{h}(x) = \sum_{i=1}^{n} \hat{\alpha}(x)_i \psi_{\mathcal{Y}}(y_i)"] \arrow[r, "\hat{f}(x) = d \circ \hat{h}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} \hat{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y)"'] & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"']
\end{tikzcd}$$

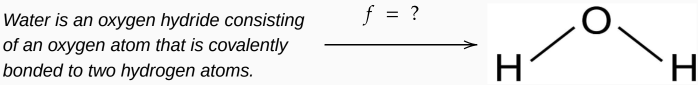
Challenges raised by IOKR

1. **Scalability:** obtain $\tilde{f} = d \circ \tilde{h}$, **computationally efficient** version of $\hat{f} = d \circ \hat{h}$, when learning from **big data**, i.e. **large n**

2. **Theory:** obtain **excess risk bound** of $\tilde{f} = d \circ \tilde{h}$

3. **Loss:** what if $\boldsymbol{\Delta}(y, y') = c(\|\boldsymbol{\psi_{\mathcal{Y}}}(y) - \boldsymbol{\psi_{\mathcal{Y}}}(y')\|^2_{\boldsymbol{\mathcal{H}_{\mathcal{Y}}}})$?

4. **Expressiveness:**



Water is an oxygen hydride consisting of an oxygen atom that is covalently bonded to two hydrogen atoms.

$$\xrightarrow{f\ =\ ?}$$

Key tool for scalability: Random Fourier Features vs Sketching

a) **Random Fourier Features** (Rahimi and Recht, 2007; Sriperumbudur and Szabó, 2015): for $m_{\mathcal{Y}} \ll n$,

$$\langle \psi_{\mathcal{Y}}(y), \psi_{\mathcal{Y}}(y') \rangle_{\mathcal{H}_{\mathcal{Y}}} \approx \langle \tilde{\psi}_{\mathcal{Y}}(y), \tilde{\psi}_{\mathcal{Y}}(y') \rangle_{\mathbb{R}^{m_{\mathcal{Y}}}}$$

$$\implies \boldsymbol{\Delta}(y, y') = \|\boldsymbol{\psi}_{\mathcal{Y}}(y) - \boldsymbol{\psi}_{\mathcal{Y}}(y')\|^2_{\mathcal{H}_{\mathcal{Y}}} \approx \|\tilde{\boldsymbol{\psi}}_{\mathcal{Y}}(y) - \tilde{\boldsymbol{\psi}}_{\mathcal{Y}}(y')\|^2_{\mathbb{R}^{m_{\mathcal{Y}}}} = \widetilde{\boldsymbol{\Delta}}(y, y')$$

$$\implies \widetilde{\boldsymbol{\Delta}} \text{ approximated loss}$$

Key tool for scalability: Random Fourier Features vs Sketching

a) **Random Fourier Features** (Rahimi and Recht, 2007; Sriperumbudur and Szabó, 2015): for $m_{\mathcal{Y}} \ll n$,

$$\langle \psi_{\mathcal{Y}}(y), \psi_{\mathcal{Y}}(y') \rangle_{\mathcal{H}_{\mathcal{Y}}} \approx \langle \tilde{\psi}_{\mathcal{Y}}(y), \tilde{\psi}_{\mathcal{Y}}(y') \rangle_{\mathbb{R}^{m_{\mathcal{Y}}}}$$

$$\implies \boldsymbol{\Delta}(y, y') = \|\boldsymbol{\psi}_{\mathcal{Y}}(y) - \boldsymbol{\psi}_{\mathcal{Y}}(y')\|^2_{\mathcal{H}_{\mathcal{Y}}} \approx \|\tilde{\boldsymbol{\psi}}_{\mathcal{Y}}(y) - \tilde{\boldsymbol{\psi}}_{\mathcal{Y}}(y')\|^2_{\mathbb{R}^{m_{\mathcal{Y}}}} = \widetilde{\boldsymbol{\Delta}}(y, y')$$

$$\implies \widetilde{\boldsymbol{\Delta}} \text{ approximated loss}$$

b) **Sketching** (Williams and Seeger, 2001; Rudi et al., 2015; Yang et al., 2017): for $m_{\mathcal{Y}} \ll n$, $R_{\mathcal{Y}} \in \mathbb{R}^{m_{\mathcal{Y}} \times n}$

$$\text{span}\left((\psi_{\mathcal{Y}}(y_i))_{i=1}^{n} \right) \leftarrow \text{span}\left(\left(\sum_{j=1}^{n} [R_{\mathcal{Y}}]_{ij} \psi_{\mathcal{Y}}(y_j) \right)_{i=1}^{m_{\mathcal{Y}}} \right)$$

$$\implies \boldsymbol{\Delta} \text{ remains unchanged!}$$

Outline of the thesis

Method	Scalability	Theory	Loss	Express.	Output dim.
RFF (Li et al., 2021)	✓	✓	✓		1
Nyström (Rudi et al., 2015)	✓	✓			1
Sketching (Yang et al., 2017)	✓	(✓)			1
Sketching (Lacotte and Pilanci, 2022)	✓	(✓)	✓		1
1. *p*-sparsified (El Ahmad et al., 2023)	✓	✓	✓		$d \geq 1$
ORFF (Brault et al., 2016)	✓		✓		∞
ILE (Ciliberto et al., 2020)		✓			∞
2. SISOKR (El Ahmad et al., 2024)	✓	✓			$\boldsymbol{\infty}$
MMR (Brouard et al., 2016b)			✓		∞
Double Rep. (Laforgue et al., 2020)			✓		∞
MOVKL (Kadri et al., 2012)				(✓)	∞
3. DSOKR (El Ahmad et al., 2024)	✓		✓	✓	$\boldsymbol{\infty}$

p-sparsified sketches for fast kernel methods with Lipschitz losses

Motivation

$\mathcal{Y} = \mathbb{R}$ (take a step aside from structured prediction)

Given $k_{\mathcal{X}}$ and its associated RKHS $\mathcal{H}_{\mathcal{X}}$, $\lambda > 0$

$$\min_{f \in \mathcal{H}_{\mathcal{X}}} \frac{1}{n} \sum_{i=1}^{n} \Delta(f(x_i), y_i) + \frac{\lambda}{2} \|f\|_{\mathcal{H}_{\mathcal{X}}}^2 .$$

Motivation

$\mathcal{Y} = \mathbb{R}$ (take a step aside from structured prediction)

Given $k_{\mathcal{X}}$ and its associated RKHS $\mathcal{H}_{\mathcal{X}}$, $\lambda > 0$

$$\min_{f \in \mathcal{H}_{\mathcal{X}}} \frac{1}{n} \sum_{i=1}^{n} \Delta(f(x_i), y_i) + \frac{\lambda}{2} \|f\|^2_{\mathcal{H}_{\mathcal{X}}} \,.$$

Representer Theorem (Kimeldorf and Wahba, 1971):
$\hat{f} = \sum_{j=1}^{n} \hat{\alpha}_i \langle \psi_{\mathcal{X}}(\cdot), \psi_{\mathcal{X}}(x_i) \rangle_{\mathcal{H}_{\mathcal{X}}}$, where

$$\hat{\alpha} = \operatorname*{arg\,min}_{\alpha \in \mathbb{R}^n} \frac{1}{n} \sum_{i=1}^{n} \Delta\left(\left[\underbrace{K_X}_{n \times n} \alpha \right]^{\top}_{i:}, y_i \right) + \frac{\lambda}{2} \alpha^{\top} \underbrace{K_X}_{n \times n} \alpha \,.$$

Optimisation problem on n parameters and n^2-matrix to store: can we reduce n?

Sub-sampling, i.e. Nyström approximation

Let $m_{\mathcal{X}} \ll n$ and $\{(\tilde{x}_i)_{i=1}^{m_{\mathcal{X}}}\} \subset \{(x_i)_{i=1}^{n}\}$ **(Sample $m_{\mathcal{X}}$ training data)**

$\text{span}((\psi_{\mathcal{X}}(x_i)_{i=1}^{n}) \leftarrow \text{span}((\psi_{\mathcal{X}}(\tilde{x}_i)_{i=1}^{m_{\mathcal{X}}})$ **(Hypothesis space reduction)**

$\implies \tilde{f} = \sum_{i=1}^{m_{\mathcal{X}}} \tilde{\gamma}_i \langle \psi_{\mathcal{X}}(\cdot), \psi_{\mathcal{X}}(x_i) \rangle_{\mathcal{H}_{\mathcal{X}}}$ where

$$\tilde{\gamma} = \operatorname*{arg\,min}_{\gamma \in \mathbb{R}^{m_{\mathcal{X}}}} \frac{1}{n} \sum_{i=1}^{n} \Delta \left(\left[\underbrace{K_{nm_{\mathcal{X}}}}_{n \times m_{\mathcal{X}}} \gamma \right]_{i:}^{\top}, y_i \right) + \frac{\lambda}{2} \gamma^{\top} \underbrace{K_{m_{\mathcal{X}} m_{\mathcal{X}}}}_{m_{\mathcal{X}} \times m_{\mathcal{X}}} \gamma$$

Sub-sampling, i.e. Nyström approximation

Let $m_{\mathcal{X}} \ll n$ and $\{(\tilde{x}_i)_{i=1}^{m_{\mathcal{X}}}\} \subset \{(x_i)_{i=1}^{n}\}$ **(Sample $m_{\mathcal{X}}$ training data)**

$\mathrm{span}((\psi_{\mathcal{X}}(x_i)_{i=1}^{n}) \leftarrow \mathrm{span}((\psi_{\mathcal{X}}(\tilde{x}_i)_{i=1}^{m_{\mathcal{X}}})$ **(Hypothesis space reduction)**

$\implies \tilde{f} = \sum_{i=1}^{m_{\mathcal{X}}} \tilde{\gamma}_i \langle \psi_{\mathcal{X}}(\cdot), \psi_{\mathcal{X}}(x_i) \rangle_{\mathcal{H}_{\mathcal{X}}}$ where

$$\tilde{\gamma} = \underset{\gamma \in \mathbb{R}^{m_{\mathcal{X}}}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \Delta \left(\left[\underbrace{K_{nm_{\mathcal{X}}}}_{n \times m_{\mathcal{X}}} \gamma \right]_{i:}^{\top}, y_i \right) + \frac{\lambda}{2} \gamma^{\top} \underbrace{K_{m_{\mathcal{X}} m_{\mathcal{X}}}}_{m_{\mathcal{X}} \times m_{\mathcal{X}}} \gamma$$

Sampling the wrong data can lead to poor results $\implies$ data-dependent sampling schemes (e.g. leverage scores) (Alaoui and Mahoney, 2015; Rudi et al., 2018; Cherfaoui et al., 2022)

Can we use a more robust and data-independent approximation scheme?

Johnson-Lindenstrauss lemma

Lemma (Johnson and Lindenstrauss, 1984)

Given $0 < \varepsilon < 1$, a set $\mathcal{S}$ of n points in $\mathbb{R}^{D}$, and an integer $d > 8(\log n)/\varepsilon^2$, there is a linear map $h : \mathbb{R}^{D} \to \mathbb{R}^{d}$ such that

$$(1-\varepsilon)\left\|u-v\right\|^2 \leq \left\|h(u)-h(v)\right\|^2 \leq (1+\varepsilon)\left\|u-v\right\|^2 ,$$

for all $u, v \in \mathcal{S}$.

Johnson-Lindenstrauss lemma

Lemma (Johnson and Lindenstrauss, 1984)

Given $0 < \varepsilon < 1$, a set $\mathcal{S}$ of n points in $\mathbb{R}^{D}$, and an integer $d > 8(\log n)/\varepsilon^2$, there is a linear map $h : \mathbb{R}^{D} \to \mathbb{R}^{d}$ such that

$$(1-\varepsilon)\left\|u-v\right\|^2 \leq \left\|h(u)-h(v)\right\|^2 \leq (1+\varepsilon)\left\|u-v\right\|^2 ,$$

for all $u, v \in \mathcal{S}$.

Proof (Boucheron et al., 2013):

1. take $h = \frac{1}{\sqrt{d}} R \in \mathbb{R}^{d \times D}$, where R_{ij} i.i.d. **sub-Gaussian** random variables
2. prove the above equation with high probability thanks to the Bernstein inequality

Gaussian sketching then?

Let $\boldsymbol{R_{\mathcal{X}}} \in \mathbb{R}^{\boldsymbol{m_{\mathcal{X}}} \times \boldsymbol{n}}$ be a **Gaussian** sketching matrix
$\tilde{f} = \sum_{i=1}^{n} [R_{\mathcal{X}}^{\top} \tilde{\gamma}]_i \langle \psi_{\mathcal{X}}(\cdot), \psi_{\mathcal{X}}(x_i) \rangle_{\mathcal{H}_{\mathcal{X}}}$

$$\hat{\gamma} = \underset{\gamma \in \mathbb{R}^{m_{\mathcal{X}}}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \Delta \left(\left[K_X R_{\mathcal{X}}^{\top} \gamma \right]_i, y_i \right) + \frac{\lambda}{2} \gamma^{\top} R_{\mathcal{X}} K_X R_{\mathcal{X}}^{\top} \gamma \,.$$

Gaussian sketching then?

Let $R_{\mathcal{X}} \in \mathbb{R}^{m_{\mathcal{X}} \times n}$ be a **Gaussian** sketching matrix
$\tilde{f} = \sum_{i=1}^{n} [R_{\mathcal{X}}^\top \tilde{\gamma}]_i \langle \psi_{\mathcal{X}}(\cdot), \psi_{\mathcal{X}}(x_i) \rangle_{\mathcal{H}_{\mathcal{X}}}$

$$\hat{\gamma} = \underset{\gamma \in \mathbb{R}^{m_{\mathcal{X}}}}{\arg\min} \ \frac{1}{n} \sum_{i=1}^{n} \Delta \left(\left[K_X R_{\mathcal{X}}^\top \gamma \right]_i, y_i \right) + \frac{\lambda}{2} \gamma^\top R_{\mathcal{X}} K_X R_{\mathcal{X}}^\top \gamma \,.$$

Problems:

1. computing $R_{\mathcal{X}} K_X$: $\mathcal{O}\left(n^2 m_{\mathcal{X}}\right)$ time complexity → **still high complexity**
2. storing K_X: $\mathcal{O}\left(n^2\right)$ space complexity → **space complexity does not change**

Definition: p-sparsified sketches

Definition (El Ahmad et al., 2023)

Let $m_{\mathcal{X}} < n$, and $p \in (0, 1]$. A p-sparsified sketch $R_{\mathcal{X}} \in \mathbb{R}^{m_{\mathcal{X}} \times n}$ is composed of i.i.d. entries

$$R_{\mathcal{X}_{ij}} = \frac{1}{\sqrt{m_{\mathcal{X}} p}} B_{ij} G_{ij} ,$$

where $B_{ij} \overset{\text{i.i.d.}}{\sim} \mathrm{Ber}(p)$ and $G_{ij} \overset{\text{i.i.d.}}{\sim} \mathrm{Rad}(\frac{1}{2})$ (p-SR) or $\mathcal{N}(0, 1)$ (p-SG).

Definition: p-sparsified sketches

Definition (El Ahmad et al., 2023)

Let $m_{\mathcal{X}} < n$, and $p \in (0, 1]$. A p-sparsified sketch $R_{\mathcal{X}} \in \mathbb{R}^{m_{\mathcal{X}} \times n}$ is composed of i.i.d. entries

$$R_{\mathcal{X}_{ij}} = \frac{1}{\sqrt{m_{\mathcal{X}} p}} B_{ij} G_{ij} \,,$$

where $B_{ij} \overset{\text{i.i.d.}}{\sim} \text{Ber}(p)$ and $G_{ij} \overset{\text{i.i.d.}}{\sim} \text{Rad}(\frac{1}{2})$ (p-SR) or $\mathcal{N}(0, 1)$ (p-SG).

$\boldsymbol{R_{\mathcal{X}_{ij}}}$ **is $\boldsymbol{\frac{1}{m_{\mathcal{X}} p}}$-sub-Gaussian** $\implies$ ***p*-sparsifed** sketches are Johnsonn-Lindenstrauss compatible sketches

Computational property: *decomposition trick*

Let $m'_{\mathcal{X}} = \sum_{j=1}^{n} \mathbb{I}\{S_{:j} \neq 0\}$, $R_{\mathcal{X}} = \underbrace{R_{\mathcal{X}_{\mathrm{SG}}}}_{m_{\mathcal{X}} \times m'_{\mathcal{X}}} \underbrace{R_{\mathcal{X}_{\mathrm{SS}}}}_{m'_{\mathcal{X}} \times n}$

Example: $\begin{pmatrix} 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$

Computational property: *decomposition trick*

Let $m'_{\mathcal{X}} = \sum_{j=1}^{n} \mathbb{I}\{S_{:j} \neq 0\}$, $R_{\mathcal{X}} = \underbrace{R_{\mathcal{X}_{\mathrm{SG}}}}_{m_{\mathcal{X}} \times m'_{\mathcal{X}}} \underbrace{R_{\mathcal{X}_{\mathrm{SS}}}}_{m'_{\mathcal{X}} \times n}$

Example: $\begin{pmatrix} 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$

$m'_{\mathcal{X}} \sim \mathrm{Binom}\left(n, 1-(1-p)^{m_{\mathcal{X}}}\right) \implies \mathbb{E}\left[m'_{\mathcal{X}}\right] = n(1-(1-p)^{m_{\mathcal{X}}}) \underset{p \to 0}{\sim} \boldsymbol{n m_{\mathcal{X}} p}$

Best of both worlds

Table 1: Complexities of $R_{\mathcal{X}} K_X$

Sketch	Time	Space
Gaussian	$\mathcal{O}\left(n^2 + n^2 m_{\mathcal{X}}\right)$	$\mathcal{O}\left(n^2\right)$
p-sparsified	$\mathcal{O}\left(n^2 m_{\mathcal{X}} p + n^2 m_{\mathcal{X}}^2 p\right)$	$\mathcal{O}\left(n^2 m_{\mathcal{X}} p\right)$

$\implies$ p-sparsified more efficient if $m_{\mathcal{X}} p < 1$!

Best of both worlds

Table 1: Complexities of $R_{\mathcal{X}} K_X$

Sketch	Time	Space
Gaussian	$\mathcal{O}\left(n^2 + n^2 m_{\mathcal{X}}\right)$	$\mathcal{O}\left(n^2\right)$
p-sparsified	$\mathcal{O}\left(n^2 m_{\mathcal{X}} p + n^2 m_{\mathcal{X}}^2 p\right)$	$\mathcal{O}\left(n^2 m_{\mathcal{X}} p\right)$

$\Longrightarrow$ **p-sparsified more efficient if $m_{\mathcal{X}} p < 1$!**

p-sparsified sketch's goal → best of both worlds with **data-independent distribution**:

1. **computational efficiency of sub-sampling sketch**
2. **statistical accuracy of Rademacher or Gaussian sketch**

Scalability ✓!

Related work: Accumulation sketching (Chen and Yang, 2021)

Excess risk bound for Lipschitz losses

Corollary

Assume that $\sigma_i(K_X/n) \propto i^{-t}$ for $t > 1$ (polynomial decay). Then, for a L-Lipschitz loss Δ, $\lambda \propto n^{-\frac{t}{1+t}}$ and a p-sparsified sketching matrix $R_{\mathcal{X}}$ such that, for any $\delta \in (0,1)$,

$$m_{\mathcal{X}} \gtrsim \max(n^{\frac{1}{1+t}}, \log(1/\delta)) \,,$$

with probability $1-\delta$

$$\mathbb{E}_{(x,y)\sim\rho}\left[\Delta(\tilde{f}(x), y)\right] - \mathbb{E}_{(x,y)\sim\rho}\left[\Delta(f_{\mathcal{H}}(x), y)\right] \lesssim \log(1/\delta) n^{-\frac{t}{2(1+t)}} \,.$$

Theory ✓, loss ✓!

Scalar regression with synthetic dataset: settings

1) $n = 10\,000$, $(x_i, y_i) \in \mathbb{R}^{10} \times \mathbb{R}$

2) Inhomogeneous input data distribution

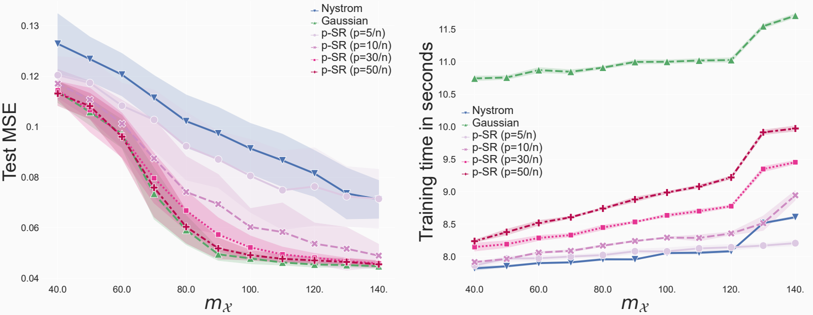
$$x_i \sim \begin{cases} \mathcal{U}\left([0_{10}, \mathbb{1}_{10}]\right)\,, & \text{if } i = 1, \ldots, 9\,900\,, \\ \mathcal{N}\left(1.5\mathbb{1}_{10}, 0.25 I_{10}\right)\,, & \text{if } i = 9\,901, \ldots, 10\,000\,, \end{cases}$$

3) $y = f^{\star}(x) + \epsilon$, where $\epsilon \sim \mathcal{N}\left(0, 1\right)$ and

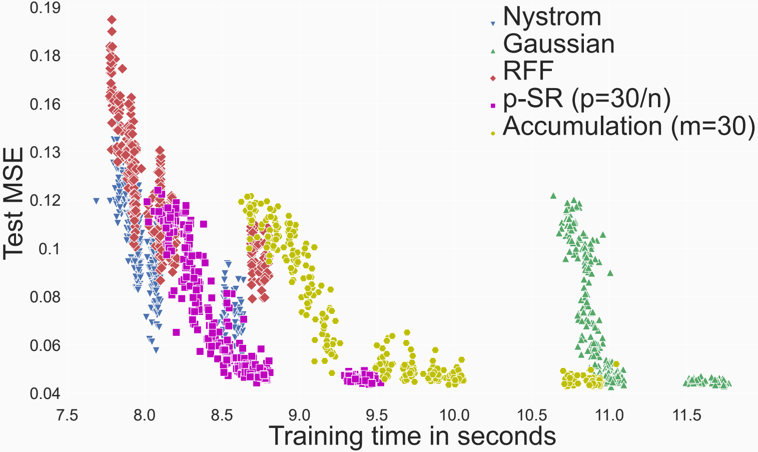
$$f^{\star}(x) = 0.1 \exp\left(4x^1\right) + \frac{4}{1 + \exp\left(-20\left(x^2 - 0.5\right)\right)} + 3x^3 + 2x^4 + x^5\,.$$

4) loss: κ-Huber

Interpolation between Nyström approximation and Gaussian sketching



Optimal computational/statistical trade-off



Sketched Input Sketched Output Kernel Regression

Motivation

$$\begin{tikzcd}
 & \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur, "\widehat{h}(x) = \sum_{i=1}^{n} \widehat{\alpha}(x)_i \psi_{\mathcal{Y}}(y_i)"] \arrow[r, "\widehat{f}(x) = d \circ \widehat{h}(x) = \underset{y \in \mathcal{Y}}{\arg\max} \sum_{i=1}^{n} \widehat{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y)"'] & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"']
\end{tikzcd}$$

Motivation

$$\begin{tikzcd}
\mathcal{H}_{\mathcal{X}} \arrow[rr, "\widehat{H}"] & & \mathcal{H}_{\mathcal{Y}} \arrow[dd, bend left, "d"] \\
& & \\
\mathcal{X} \arrow[uu, "\psi_{\mathcal{X}}"] \arrow[uurr, "\widehat{h}(x) = \sum_{i=1}^{n} \widehat{\alpha}(x)_i \psi_{\mathcal{Y}}(y_i) = \widehat{H}\psi_{\mathcal{X}}(x)"] \arrow[rr, "\widehat{f}(x) = d \circ \widehat{h}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} \widehat{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y)"'] & & \mathcal{Y} \arrow[uu, "\psi_{\mathcal{Y}}"']
\end{tikzcd}$$

IOKR: Weston et al. (2003); Cortes et al. (2005); Brouard et al. (2011); Kadri et al. (2013); Brouard et al. (2016b); Korba et al. (2018)

Motivation

$$\begin{tikzcd}
\mathcal{H}_{\mathcal{X}} \arrow[rr, "\widehat{H}"] & & \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[u, "\psi_{\mathcal{X}}"] \arrow[urr, "\hat{h}(x) = \sum_{i=1}^{n} \widehat{\alpha}(x)_i \psi_{\mathcal{Y}}(y_i) = \widehat{H}\psi_{\mathcal{X}}(x)"] \arrow[rr] & & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"']
\end{tikzcd}$$

$$\hat{f}(x) = d \circ \hat{h}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} \widehat{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y)$$

Motivation: build a **low-rank** approximation $\tilde{h}$ of $\hat{h}$ thanks to **input and output** random projectors $\widetilde{P}_X$ and $\widetilde{P}_Y$ to obtain a scalable predictor $\tilde{f}$ together with an excess risk bound

IOKR: training and inference complexities

1. Training: $\hat{\alpha}(x) = (\underbrace{K_X + n\lambda I_n}_{n \times n})^{-1} k_X^x = \widehat{\Omega} k_X^x$

$\implies \mathcal{O}\left(n^3\right)$ time and $\mathcal{O}\left(n^2\right)$ space complexity

IOKR: training and inference complexities

1. Training: $\hat{\alpha}(x) = \underbrace{(K_X + n\lambda I_n)}_{n\times n}{}^{-1} k_X^x = \widehat{\Omega} k_X^x$

$\implies \mathcal{O}\left(n^3\right)$ time and $\mathcal{O}\left(n^2\right)$ space complexity

2. Inference: $\hat{f}(x) = \underset{y\in\mathcal{Y}}{\arg\max} \sum_{i=1}^{n} \hat{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y) = {k_X^x}^T \widehat{\Omega} k_Y{}^y$

- **Test set:** $X^{te} = \{x_1^{\mathrm{te}}, \ldots, x_{n_{\mathrm{te}}}^{\mathrm{te}}\}$ of size n_{te}
- **Candidate set:** $Y^{\mathrm{c}} = \{y_1^{\mathrm{c}}, \ldots, y_{n_{\mathrm{c}}}^{\mathrm{c}}\}$ of size n_{c}

$$\underbrace{K_X^{\mathbf{te,tr}}}_{n_{\mathrm{te}}\times n}\underbrace{\widehat{\Omega}}_{n\times n}\underbrace{K_Y^{\mathbf{tr,c}}}_{n\times n_{\mathrm{c}}}$$

$$\hat{f}(x_i^{\mathrm{te}}) = y_j^{\mathrm{c}} \quad \text{where} \quad j = \underset{1\leq j\leq n_{\mathrm{c}}}{\arg\max}\, [K_X^{\mathbf{te,tr}}\widehat{\Omega}K_Y^{\mathbf{tr,c}}]_{ij}$$

$\implies \mathcal{O}\left(n_{\mathrm{te}} n n_{\mathrm{c}}\right)$ time and $\mathcal{O}\left(n n_{\mathrm{c}}\right)$ space complexity if $n_{\mathrm{te}} < n \leq n_{\mathrm{c}}$

Some notations

For an i.i.d. sample $(z_i)_{i=1}^n \in \mathcal{Z}^n \sim \rho_Z$:

- $S_Z : f \in \mathcal{H}_\mathcal{Z} \mapsto (1/\sqrt{n})(\langle f, \psi_\mathcal{Z}(z_1)\rangle_{\mathcal{H}_\mathcal{Z}}, \dots, \langle f, \psi_\mathcal{Z}(z_n)\rangle_{\mathcal{H}_\mathcal{Z}})^\top \in \mathbb{R}^n$ sampling operator
- $S_Z^\# : \alpha \in \mathbb{R}^n \mapsto (1/\sqrt{n}) \sum_{i=1}^n \alpha_i \psi_\mathcal{Z}(z_i) \in \textbf{span}\left((\psi_\mathcal{Z}(z_i))_{i=1}^n\right)$ its adjoint
- $C_\mathcal{Z} = \mathbb{E}_z[\psi_\mathcal{Z}(z) \otimes \psi_\mathcal{Z}(z)]$ covariance operator
- $\widehat{C}_Z = (1/n) \sum_{i=1}^n \psi_\mathcal{Z}(z_i) \otimes \psi_\mathcal{Z}(z_i) = S_Z^\# S_Z$ its empirical counterpart:

  $\widehat{C}_Z : \mathcal{H}_\mathcal{Z} \to \textbf{span}\left((\psi_\mathcal{Z}(z_i))_{i=1}^n\right)$

Low-rank estimator: from IOKR to SISOKR

$$\begin{tikzcd}
\mathcal{H}_\mathcal{X} \arrow[r, "\widehat{H} = S_Y^\# S_X \left(S_X^\# S_X + \lambda I_{\mathcal{H}_\mathcal{X}}\right)^{-1}"] & \mathcal{H}_\mathcal{Y} \arrow[d, bend left=60, "d"] \\
\mathcal{X} \arrow[u, "\psi_\mathcal{X}"] \arrow[ur, "\hat{h}(x) = \sum_{i=1}^{n} \widehat{\alpha}(x)_i \psi_\mathcal{Y}(y_i) = \widehat{H}\psi_\mathcal{X}(x)"] \arrow[r, "\widehat{f}(x) = d \circ \widehat{h}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} \widehat{\alpha}(x)_i k_\mathcal{Y}(y_i, y)"'] & \mathcal{Y} \arrow[u, "\psi_\mathcal{Y}"']
\end{tikzcd}$$

Low-rank estimator: from IOKR to SISOKR

$$\begin{tikzcd}
\mathcal{H}_{\mathcal{X}} \arrow[r, "{\widetilde{H} = \widetilde{P}_Y S_Y^{\#} S_X \widetilde{P}_X \left(\widetilde{P}_X S_X^{\#} S_X \widetilde{P}_X + \lambda I_{\mathcal{H}_{\mathcal{X}}}\right)^{-1}}"] & \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left=60, "d"] \\
\mathcal{X} \arrow[u, "\psi_{\mathcal{X}}"] \arrow[ur, sloped, "{\widetilde{h}(x) = \widetilde{H}\psi_{\mathcal{X}}(x) = \sum_{i=1}^{n} \widetilde{\alpha}_i(x)\psi_{\mathcal{Y}}(y_i)}"] \arrow[r, "{\widetilde{f}(x) = d \circ \widetilde{h}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} \widetilde{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y)}"'] & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"]
\end{tikzcd}$$

$\widetilde{P}_Z : \mathcal{H}_{\mathcal{Z}} \to \widetilde{\mathcal{H}}_{\mathcal{Z}}$ where $\widetilde{\mathcal{H}}_{\mathcal{Z}} := \operatorname{span}\left(\left(\sum_{j=1}^{n} [R_{\mathcal{Z}}]_{ij} \psi_{\mathcal{Z}}(z_j)\right)_{i=1}^{m_{\mathcal{Z}}}\right)$

How to build these projectors?

Construction of the orthogonal projector $\widetilde{P}_Z$

- $\widehat{C}_Z = S_Z^{\#} S_Z = (1/n) \sum_{i=1}^{n} \psi_{\mathcal{Z}}(z_i) \otimes \psi_{\mathcal{Z}}(z_i)$
- $\widetilde{C}_Z = S_Z^{\#} R_{\mathcal{Z}}^{\top} R_{\mathcal{Z}} S_Z = \frac{1}{n} \sum_{l=1}^{m_{\mathcal{Z}}} \left(\sum_{i=1}^{n} R_{\mathcal{Z}_{li}} \psi_{\mathcal{Z}}(z_i)\right) \otimes \left(\sum_{j=1}^{n} R_{\mathcal{Z}_{lj}} \psi_{\mathcal{Z}}(z_j)\right)$
- $\widetilde{K}_Z = R_{\mathcal{Z}} K_Z R_{\mathcal{Z}}{}^{\top}$, and $\left\{\left(\sigma_i(\widetilde{K}_Z), \tilde{\mathbf{u}}_i^Z\right), i \in [m_{\mathcal{Z}}]\right\}$ its eigenpairs
- $p_Z = \operatorname{rank}\left(\widetilde{K}_Z\right)$, and for all $1 \leq i \leq p_Z$, $\tilde{\boldsymbol{e}}_i^Z = \sqrt{\frac{n}{\sigma_i(\widetilde{K}_Z)}} S_Z^{\#} R_{\mathcal{Z}}^{\top} \tilde{\mathbf{u}}_i^Z \in \mathcal{H}_{\mathcal{Z}}$

Construction of the orthogonal projector $\widetilde{P}_Z$

- $\widehat{C}_Z = S_Z^{\#} S_Z = (1/n)\sum_{i=1}^{n} \psi_{\mathcal{Z}}(z_i) \otimes \psi_{\mathcal{Z}}(z_i)$
- $\widetilde{C}_Z = S_Z^{\#} R_{\mathcal{Z}}^{\top} R_{\mathcal{Z}} S_Z = \frac{1}{n} \sum_{l=1}^{m_{\mathcal{Z}}} \left(\sum_{i=1}^{n} R_{\mathcal{Z}_{li}} \psi_{\mathcal{Z}}(z_i)\right) \otimes \left(\sum_{j=1}^{n} R_{\mathcal{Z}_{lj}} \psi_{\mathcal{Z}}(z_j)\right)$
- $\widetilde{K}_Z = R_{\mathcal{Z}} K_Z R_{\mathcal{Z}}{}^{\top}$, and $\left\{\left(\sigma_i(\widetilde{K}_Z), \tilde{\mathbf{u}}_i^Z\right), i \in [m_{\mathcal{Z}}]\right\}$ its eigenpairs
- $p_Z = \operatorname{rank}\left(\widetilde{K}_Z\right)$, and for all $1 \le i \le p_Z$, $\tilde{\mathbf{e}}_i^Z = \sqrt{\frac{n}{\sigma_i(\widetilde{K}_Z)}} S_Z^{\#} R_{\mathcal{Z}}^{\top} \tilde{\mathbf{u}}_i^Z \in \mathcal{H}_{\mathcal{Z}}$

Proposition (El Ahmad et al., 2024)

The $\tilde{\mathbf{e}}_i^Z$s are the **eigenfunctions**, associated to the eigenvalues $\sigma_i(\widetilde{K}_Z)/n$, of $\widetilde{C}_Z$, whose range is **span**$\left(\left(\sum_{j=1}^{n} R_{\mathcal{Z}_{ij}} \psi_{\mathcal{Z}}(z_j)\right)_{i=1}^{m_{\mathcal{Z}}}\right)$. Then, $\widetilde{E}^Z = (\tilde{\mathbf{e}}_1^Z, \ldots, \tilde{\mathbf{e}}_{p_Z}^Z)$ is an **orthonormal basis** of **span**$\left(\left(\sum_{j=1}^{n} R_{\mathcal{Z}_{ij}} \psi_{\mathcal{Z}}(z_j)\right)_{i=1}^{m_{\mathcal{Z}}}\right)$, and $\widetilde{P}_Z$ writes as

$$\widetilde{P}_Z = \sum_{i=1}^{p_Z} \langle \cdot, \tilde{\mathbf{e}}_i^Z \rangle_{\mathcal{H}_{\mathcal{Z}}} \tilde{\mathbf{e}}_i^Z = (R_{\mathcal{Z}} S_Z)^{\#} \left(R_{\mathcal{Z}} S_Z (R_{\mathcal{Z}} S_Z)^{\#}\right)^{\dagger} R_{\mathcal{Z}} S_Z \,.$$

Related works on Nyström: Yang et al. (2012); Rudi et al. (2015)

Sketched Input Sketched Output Kernel Regression estimator

Proposition (El Ahmad et al., 2024)

$$\tilde{h}(x) = \sum_{i=1}^{n} \tilde{\alpha}_i\,(x)\,\psi_{\mathcal{Y}}\,(y_i)\ , \quad \text{where} \quad \tilde{\alpha}\,(x) = R_{\mathcal{Y}}^{\top}\widetilde{\Omega}R_{\mathcal{X}}k_X^x\,,$$

with

$$\widetilde{\Omega} = \underbrace{(R_{\mathcal{Y}}K_YR_{\mathcal{Y}}^{\top})}_{m_{\mathcal{Y}}\times m_{\mathcal{Y}}}{}^{\dagger}R_{\mathcal{Y}}K_YK_XR_{\mathcal{X}}^{\top}\underbrace{(R_{\mathcal{X}}K_X^2R_{\mathcal{X}}^{\top} + n\lambda R_{\mathcal{X}}K_XR_{\mathcal{X}}^{\top})}_{m_{\mathcal{X}}\times m_{\mathcal{X}}}{}^{\dagger}$$

Sketched Input Sketched Output Kernel Regression estimator

Proposition (El Ahmad et al., 2024)

$$\tilde{h}(x) = \sum_{i=1}^{n} \tilde{\alpha}_i(x)\,\psi_{\mathcal{Y}}(y_i)\ , \quad \text{where} \quad \tilde{\alpha}(x) = R_{\mathcal{Y}}^{\top}\widetilde{\Omega}R_{\mathcal{X}}k_X^x\,,$$

with

$$\widetilde{\Omega} = \underbrace{(R_{\mathcal{Y}}K_YR_{\mathcal{Y}}^{\top})}_{m_{\mathcal{Y}}\times m_{\mathcal{Y}}}{}^{\dagger}R_{\mathcal{Y}}K_YK_XR_{\mathcal{X}}^{\top}\underbrace{(R_{\mathcal{X}}K_X^2R_{\mathcal{X}}^{\top} + n\lambda R_{\mathcal{X}}K_XR_{\mathcal{X}}^{\top})}_{m_{\mathcal{X}}\times m_{\mathcal{X}}}{}^{\dagger}$$

Method	Time	Space
IOKR	$\mathcal{O}(n^3)$	$\mathcal{O}(n^2)$
SISOKR (p-SR/SG)	$\mathcal{O}(\mathbf{max}(m_{\mathcal{X}}, m_{\mathcal{Y}})^2pn)$	$\mathcal{O}(\mathbf{max}(m_{\mathcal{X}}, m_{\mathcal{Y}})pn)$

$\Longrightarrow$ **Training complexity reduced thanks to input sketching!**

SISOKR estimator: Inference

$$\tilde{f}(x) = \underset{y\in\mathcal{Y}}{\arg\max}\ \sum_{i=1}^{n} \tilde{\boldsymbol{\alpha}}(x)_i \boldsymbol{k}_{\mathcal{Y}}(y_i, y) = \underset{y\in\mathcal{Y}}{\arg\max}\ {k_X^x}^T R_{\mathcal{X}}^\top \widetilde{\Omega} R_{\mathcal{Y}} \boldsymbol{k}_Y{}^y$$

$$\underbrace{K_X^{\mathbf{te,tr}} R_{\mathcal{X}}^\top}_{n_{\mathbf{te}}\times m_{\mathcal{X}}} \underbrace{\widetilde{\boldsymbol{\Omega}}}_{m_{\mathcal{X}}\times m_{\mathcal{Y}}} \underbrace{R_{\mathcal{Y}} K_Y^{\mathbf{tr,c}}}_{m_{\mathcal{Y}}\times n_{\mathbf{c}}}$$

$$\tilde{f}(x_i^{\mathrm{te}}) = y_j^{\mathrm{c}} \quad \text{where} \quad j = \underset{1\le j\le n_{\mathrm{c}}}{\arg\max}\ [K_X^{\mathbf{te,tr}} R_{\mathcal{X}}^\top \widetilde{\Omega} R_{\mathcal{Y}} K_Y^{\mathbf{tr,c}}]_{ij}$$

SISOKR estimator: Inference

$$\tilde{f}(x) = \arg\max_{y\in\mathcal{Y}} \sum_{i=1}^{n} \tilde{\alpha}(x)_i k_{\mathcal{Y}}(y_i, y) = \arg\max_{y\in\mathcal{Y}} {k_X^x}^T R_{\mathcal{X}}^\top \widetilde{\Omega} R_{\mathcal{Y}} k_Y^y$$

$$\underbrace{K_X^{\mathbf{te,tr}} R_{\mathcal{X}}^\top}_{n_{\mathbf{te}}\times m_{\mathcal{X}}} \underbrace{\widetilde{\Omega}}_{m_{\mathcal{X}}\times m_{\mathcal{Y}}} \underbrace{R_{\mathcal{Y}} K_Y^{\mathbf{tr,c}}}_{m_{\mathcal{Y}}\times n_{\mathbf{c}}}$$

$$\tilde{f}(x_i^{\text{te}}) = y_j^{\text{c}} \quad \text{where} \quad j = \arg\max_{1\leq j\leq n_{\text{c}}} [K_X^{\mathbf{te,tr}} R_{\mathcal{X}}^\top \widetilde{\Omega} R_{\mathcal{Y}} K_Y^{\mathbf{tr,c}}]_{ij}$$

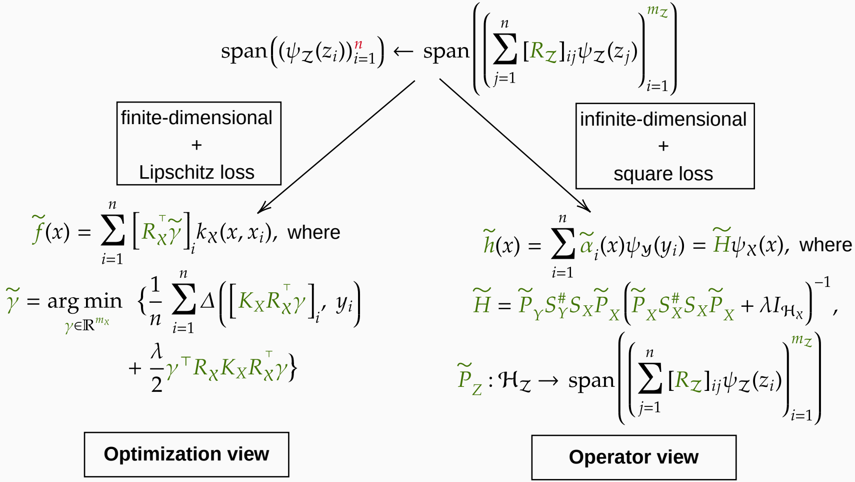
Table 2: If $n_{\text{te}} \leq m_{\mathcal{X}}, m_{\mathcal{Y}} < n \leq n_{\mathbf{c}}$

Method	Time	Space
IOKR	$\mathcal{O}\left(n_{\mathbf{te}} n n_{\mathbf{c}}\right)$	$\mathcal{O}\left(n n_{\mathbf{c}}\right)$
SISOKR (p-SR/SG)	$\mathcal{O}\left(\mathbf{max}(n_{\mathbf{te}}, n m_{\mathcal{Y}} p) m_{\mathcal{Y}} n_{\mathbf{c}}\right)$	$\mathcal{O}\left(n p m_{\mathcal{Y}} n_{\mathbf{c}}\right)$

$\Longrightarrow$ Inference complexity reduced thanks to output sketching!

Scalability ✓!

Sketching for kernel methods: summary



$$\operatorname{span}\Big((\psi_{\mathcal{Z}}(z_i))_{i=1}^{n}\Big) \leftarrow \operatorname{span}\left(\left(\sum_{j=1}^{n}[R_{\mathcal{Z}}]_{ij}\psi_{\mathcal{Z}}(z_j)\right)_{i=1}^{m_{\mathcal{Z}}}\right)$$

finite-dimensional
+
Lipschitz loss

infinite-dimensional
+
square loss

$$\tilde{f}(x) = \sum_{i=1}^{n}\left[R_{\mathcal{X}}^{\top}\tilde{\gamma}\right]_i k_{\mathcal{X}}(x, x_i), \text{ where}$$

$$\tilde{\gamma} = \underset{\gamma\in\mathbb{R}^{m_{\mathcal{X}}}}{\arg\min}\ \Big\{\frac{1}{n}\sum_{i=1}^{n}\Delta\Big(\left[K_{\mathcal{X}}R_{\mathcal{X}}^{\top}\gamma\right]_i,\ y_i\Big) + \frac{\lambda}{2}\gamma^{\top}R_{\mathcal{X}}K_{\mathcal{X}}R_{\mathcal{X}}^{\top}\gamma\Big\}$$

$$\tilde{h}(x) = \sum_{i=1}^{n}\tilde{\alpha}_i(x)\psi_{\mathcal{Y}}(y_i) = \tilde{H}\psi_{\mathcal{X}}(x), \text{ where}$$

$$\tilde{H} = \tilde{P}_{\mathcal{Y}}S_{\mathcal{Y}}^{\#}S_{\mathcal{X}}\tilde{P}_{\mathcal{X}}\left(\tilde{P}_{\mathcal{X}}S_{\mathcal{X}}^{\#}S_{\mathcal{X}}\tilde{P}_{\mathcal{X}} + \lambda I_{\mathcal{H}_{\mathcal{X}}}\right)^{-1},$$

$$\tilde{P}_{\mathcal{Z}} : \mathcal{H}_{\mathcal{Z}} \to \operatorname{span}\left(\left(\sum_{j=1}^{n}[R_{\mathcal{Z}}]_{ij}\psi_{\mathcal{Z}}(z_i)\right)_{i=1}^{m_{\mathcal{Z}}}\right)$$

Optimization view

Operator view

Theoretical guarantees of SISOKR

Let

$$\mathcal{R}(f) = \mathbb{E}_{(x,y)\sim\rho}[\boldsymbol{\Delta}(f(x), y)]\,,$$

and

$$f^* = \underset{f:\mathcal{X}\to\mathcal{Y}}{\arg\inf}\; \mathbb{E}_{(x,y)\sim\rho}[\boldsymbol{\Delta}(f(x), y)]\,,$$

we want to control

$$\mathcal{R}(\tilde{f}) - \mathcal{R}(f^*) \leq \quad \mathbf{?}$$

Assumptions

Asm. 1 (Attainability): Recall that $h^*(x) := \mathbb{E}_Y[\psi_{\mathcal{Y}}(Y) \mid X = x]$. There exists $H : \mathcal{H}_{\mathcal{X}} \to \mathcal{H}_{\mathcal{Y}}$ with $\|H\|_{\mathsf{HS}} < +\infty$ such that

$$h^*(x) = H\psi_{\mathcal{X}}(x) \quad \forall x \in \mathcal{X} \,.$$

Asm. 2 (Bounded kernel): there exists $\kappa_{\mathcal{Z}} > 0$ such that

$$k_{\mathcal{Z}}(z, z) \leq \kappa_{\mathcal{Z}}^2 \quad \forall z \in \mathcal{Z} \,.$$

Asm. 3 (Capacity condition): there exists $\gamma_{\mathcal{Z}} \in [0, 1]$ such that

$$Q_{\mathcal{Z}} := \mathrm{Tr}(C_{\mathcal{Z}}^{\gamma_{\mathcal{Z}}}) < +\infty \,.$$

Asm. 4 (Embedding property): there exists $b_{\mathcal{Z}} > 0$ and $\mu_{\mathcal{Z}} \in [0, 1]$ such that almost surely

$$\psi_{\mathcal{Z}}(z) \otimes \psi_{\mathcal{Z}}(z) \preceq b_{\mathcal{Z}} C_{\mathcal{Z}}^{1-\mu_{\mathcal{Z}}} \,.$$

Asm. 5 (Sub-Gaussian sketches): $R_{\mathcal{Z}} \in \mathbb{R}^{m_{\mathcal{Z}} \times n}$ composed with i.i.d. entries s.t. (i) $\mathbb{E}\left[R_{\mathcal{Z}_{ij}}\right] = 0$, (ii) $\mathbb{E}\left[R_{\mathcal{Z}_{ij}}^2\right] = 1/m_{\mathcal{Z}}$ and (iii) $R_{\mathcal{Z}_{ij}} \sim \frac{\nu_{\mathcal{Z}}^2}{m\mathcal{Z}}$ – sub-Gaussian with $\nu_{\mathcal{Z}} \geq 1$.

Theorem: SISOKR learning rates (El Ahmad et al., 2024)

Under **Asm. 1, 2, 3, 4 and 5**, if for all $y \in \mathcal{Y}$, $\|\psi_{\mathcal{Y}}(y)\|_{\mathcal{H}_{\mathcal{Y}}} = \kappa_{\mathcal{Y}}$, for $\mathcal{Z} \in \{\mathcal{X}, \mathcal{Y}\}$ and for $n \in \mathbb{N}$ sufficiently large such that $\frac{9}{n}\log(n/\delta) \leq n^{-\frac{1}{1+\gamma_{\mathcal{Z}}}} \leq \|C_{\mathcal{Z}}\|_{\text{op}}/2$, and for sketching sizes $m_{\mathcal{Z}}, \in \mathbb{N}$ such that

$$m_{\mathcal{Z}} \gtrsim \max\left(\nu_{\mathcal{Z}}^2 n^{\frac{\gamma_{\mathcal{Z}}+\mu_{\mathcal{Z}}}{1+\gamma_{\mathcal{Z}}}}, \nu_{\mathcal{Z}}^4 \log\left(1/\delta\right)\right),$$

then with probability $1-\delta$

$$\mathbb{E}[\|\tilde{h}(x) - h^*(x)\|^2_{\mathcal{H}_{\mathcal{Y}}}]^{\frac{1}{2}} \lesssim \log\left(4/\delta\right) n^{-\frac{1-\gamma_{\mathcal{X}} \vee \gamma_{\mathcal{Y}}}{2(1+\gamma_{\mathcal{X}} \vee \gamma_{\mathcal{Y}})}},$$

and

$$\mathcal{R}(\tilde{f}) - \mathcal{R}(f^*) \lesssim \mathbb{E}[\|\tilde{h}(x) - h^*(x)\|^2_{\mathcal{H}_{\mathcal{Y}}}]^{\frac{1}{2}} \lesssim \log\left(4/\delta\right) n^{-\frac{1-\gamma_{\mathcal{X}} \vee \gamma_{\mathcal{Y}}}{2(1+\gamma_{\mathcal{X}} \vee \gamma_{\mathcal{Y}})}}.$$

Theory ✓!

Synthetic and real-world experiments: take-home messages

1) a) Input sketching: mainly accelerates the **training** phase

1) b) Output sketching: accelerates the **inference** phase

Synthetic and real-world experiments: take-home messages

1) a) Input sketching: mainly accelerates the **training** phase

1) b) Output sketching: accelerates the **inference** phase

2) Optimal computational/statistical trade-off: statistical performance **converges** when $m_{\mathcal{X}} / m_{\mathcal{Y}}$ **increases** $\implies$ **no need to set them too high!**

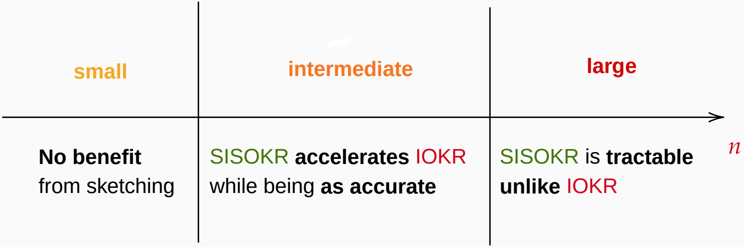
Synthetic and real-world experiments: take-home messages

1) a) Input sketching: mainly accelerates the **training** phase

1) b) Output sketching: accelerates the **inference** phase

2) Optimal computational/statistical trade-off: statistical performance **converges** when $m_{\mathcal{X}}/m_{\mathcal{Y}}$ **increases** $\implies$ **no need to set them too high!**

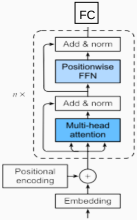
3) Benefits from sketching w.r.t. the number of training data n:



small	**intermediate**	**large**
No benefit from sketching	SISOKR **accelerates** IOKR while being **as accurate**	SISOKR is **tractable unlike** IOKR

Deep Sketched Output Kernel Regression

Motivation

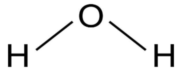


$$h_{\hat{\theta}}(x) = \sum_{i=1}^{n} g_{\hat{\theta}}(x)_i \psi_{\mathcal{Y}}(y_i)$$

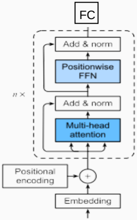
$$\begin{tikzcd}
 & \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur] \arrow[r] & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"']
\end{tikzcd}$$

$$f_{\hat{\theta}}(x) = d \circ h_{\hat{\theta}}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} g_{\hat{\theta}}(x)_i k_{\mathcal{Y}}(y_i, y)$$

"Water is an oxygen hydride consisting of an oxygen atom that is covalently bonded to two hydrogen atoms."



Motivation

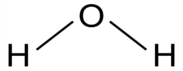


$$h_{\hat{\theta}}(x) = \sum_{i=1}^{n} g_{\hat{\theta}}(x)_i \psi_{\mathcal{Y}}(y_i)$$

$$\begin{tikzcd}
 & \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur] \arrow[r] & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"]
\end{tikzcd}$$

$$f_{\hat{\theta}}(x) = d \circ h_{\hat{\theta}}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{n} g_{\hat{\theta}}(x)_i k_{\mathcal{Y}}(y_i, y)$$

"Water is an oxygen hydride consisting of an oxygen atom that is covalently bonded to two hydrogen atoms."



Motivation: reduce the size of the linear combination to unlock the use of **deep neural networks** within the Output Kernel Regression.

DSOKR: a basis approach

$$\begin{tikzcd}
 & \mathrm{span}\left(\left(\tilde{e}_i^{Y}\right)_{i=1}^{p_Y}\right) \subsetneq \mathcal{H}_{\mathcal{Y}} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur, "h_{\widehat{\theta}}(x) = g_{\widetilde{E}} \circ g_{\widehat{W}}(x) = \sum_{i=1}^{p_Y} g_{\widehat{W}}(x)_i \tilde{e}_i^{Y}"] \arrow[r, "f_{\widehat{\theta}}(x) = d \circ h_{\widehat{\theta}}(x) = \arg\max_{y \in \mathcal{Y}} \sum_{i=1}^{p_Y} g_{\widehat{W}}(x)_i \langle \tilde{e}_i^{Y}, \psi_{\mathcal{Y}}(y) \rangle_{\mathcal{H}_{\mathcal{Y}}}"'] & \mathcal{Y} \arrow[u, "\psi_{\mathcal{Y}}"']
\end{tikzcd}$$

Solving the surrogate problem

$$\min_{W \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^{n} \| g_{\widetilde{E}} \circ g_W(x_i) - \boldsymbol{\psi}_{\boldsymbol{\mathcal{Y}}}(y_i) \|^2_{\boldsymbol{\mathcal{H}}_{\boldsymbol{\mathcal{Y}}}}$$

Solving the surrogate problem

$$\min_{W \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^{n} \|g_{\widetilde{E}} \circ g_W(x_i) - \psi_{\mathcal{Y}}(y_i)\|^2_{\mathcal{H}_{\mathcal{Y}}}$$

$$\left\|g_{\widetilde{E}} \circ g_W(x) - \psi_{\mathcal{Y}}(y)\right\|^2_{\mathcal{H}_{\mathcal{Y}}} = \left\|\sum_{i=1}^{p_Y} g_W(x)_j \tilde{e}_j^Y - \psi_{\mathcal{Y}}(y)\right\|^2_{\mathcal{H}_{\mathcal{Y}}}$$

$$= \left\|g_W(x) - \tilde{\psi}_{\mathcal{Y}}(y)\right\|_2^2 - \left(\left\|\tilde{\psi}_{\mathcal{Y}}(y)\right\|_2^2 + k_{\mathcal{Y}}(y, y)\right)$$

where

- $\tilde{\psi}_{\mathcal{Y}}(y) = \widetilde{D}_{p_Y}{}^{-1/2} \widetilde{U}_{p_Y}{}^{\top} R_{\mathcal{Y}} k_Y{}^y \in \mathbb{R}^{p_Y}$
- $\widetilde{U}_{p_Y} \widetilde{D}_{p_Y} \widetilde{U}_{p_Y}{}^{\top} = \underbrace{\widetilde{K}_Y}_{m_{\mathcal{Y}} \times m_{\mathcal{Y}}} = R_{\mathcal{Y}} K_Y R_{\mathcal{Y}}{}^{\top}$ (SVD of $\widetilde{K}_Y$)
- $k_Y{}^y = (k_{\mathcal{Y}}(y, y_1), \dots, k_{\mathcal{Y}}(y, y_n))$

Deep Sketched Output Kernel Regression: inference

$$f_{\hat{\theta}}(x) = \underset{y \in \mathcal{Y}}{\arg\max} \sum_{i=1}^{p_Y} g_{\hat{W}}(x)_i \langle \tilde{e}_i^Y, \psi_{\mathcal{Y}}(y) \rangle_{\mathcal{H}_{\mathcal{Y}}} = \underset{y \in \mathcal{Y}}{\arg\max}\, g_{\hat{W}}(x)^\top \tilde{\psi}_{\mathcal{Y}}(y)$$

- **Test set:** $X^{te} = \{x_1^{\mathrm{te}}, \dots, x_{n_{\mathrm{te}}}^{\mathrm{te}}\}$ of size n_{te}
- **Candidate set:** $Y^{\mathrm{c}} = \{y_1^{\mathrm{c}}, \dots, y_{n_{\mathrm{c}}}^{\mathrm{c}}\}$ of size n_{c}

$$f_{\hat{\theta}}(x_i^{\mathrm{te}}) = y_j^{\mathrm{c}} \quad \text{where} \quad j = \underset{1 \leq j \leq n_{\mathrm{c}}}{\arg\max}\, g_{\hat{W}}(x_i^{\mathrm{te}})^\top \tilde{\psi}_{\mathcal{Y}}(y_j^{\mathrm{c}})$$

DSOKR: summary

1. Training. a. Computations for the basis $\widetilde{E}$.

   - SVD of $\widetilde{K}_Y = R_{\mathcal{Y}} K_Y R_{\mathcal{Y}}^\top \rightarrow \left\{ \left(\sigma_i(\widetilde{K}_Y), \tilde{\mathrm{u}}_i \right), i \in [m_{\mathcal{Y}}] \right\}$
   - $\widetilde{M} = \widetilde{D}_{p_Y}{}^{-1/2} \widetilde{U}_{p_Y}{}^\top \in \mathbb{R}^{p_Y \times m_{\mathcal{Y}}}$, where $\widetilde{U}_{p_Y} = (\tilde{\mathrm{u}}_1, \ldots, \tilde{\mathrm{u}}_{p_Y})$,
     $\widetilde{D}_{p_Y} = \mathrm{diag}(\sigma_1(\widetilde{K}_Y), \ldots, \sigma_{p_Y}(\widetilde{K}_Y))$

1. Training. b. Solving the surrogate problem.

   - $\{(x_i, y_i)\}_{i=1}^n \leftarrow \{(x_i, \tilde{\psi}_{\mathcal{Y}}(y_i))\}_{i=1}^n$,
     $\{(x_i^{\mathrm{val}}, y_i^{\mathrm{val}})\}_{i=1}^{n_{\mathrm{val}}} \leftarrow \{(x_i, \tilde{\psi}_{\mathcal{Y}}(y_i^{\mathrm{val}}))\}_{i=1}^{n_{\mathrm{val}}}$, where $\tilde{\psi}_{\mathcal{Y}}(y) = \widetilde{M} R_{\mathcal{Y}} k_Y{}^y$
   - $g_{\hat{W}} = \underset{g_W, W \in \mathcal{W}}{\arg\min} \frac{1}{n} \sum_{i=1}^n \left\| g_{\hat{W}}(x_i) - \tilde{\psi}_{\mathcal{Y}}(y_i) \right\|_2^2$

2. Inference.

   - $\{y_i^{\mathrm{c}}\}_{i=1}^{n_{\mathrm{c}}} \leftarrow \{\tilde{\psi}_{\mathcal{Y}}(y_i^{\mathrm{c}})\}_{i=1}^{n_{\mathrm{c}}}$
   - $f_{\hat{\theta}}(x_i^{\mathrm{te}}) = y_j^{\mathrm{c}}$ where $j = \underset{1 \leq j \leq n_{\mathrm{c}}}{\arg\max}\, g_{\hat{W}}(x_i^{\mathrm{te}})^\top \tilde{\psi}_{\mathcal{Y}}(y_j^{\mathrm{c}})$

DSOKR: summary

1. Training. a. Computations for the basis $\widetilde{E}$.
   - SVD of $\widetilde{K}_Y = R_{\mathcal{Y}} K_Y R_{\mathcal{Y}}{}^\top \to \left\{ \left(\sigma_i(\widetilde{K}_Y), \tilde{u}_i \right), i \in [m_{\mathcal{Y}}] \right\}$
   - $\widetilde{M} = \widetilde{D}_{p_Y}{}^{-1/2} \widetilde{U}_{p_Y}{}^\top \in \mathbb{R}^{p_Y \times m_{\mathcal{Y}}}$, where $\widetilde{U}_{p_Y} = (\tilde{u}_1, \dots, \tilde{u}_{p_Y})$, $\widetilde{D}_{p_Y} = \operatorname{diag}(\sigma_1(\widetilde{K}_Y), \dots, \sigma_{p_Y}(\widetilde{K}_Y))$

1. Training. b. Solving the surrogate problem.
   - $\{(x_i, y_i)\}_{i=1}^n \leftarrow \{(x_i, \tilde{\psi}_{\mathcal{Y}}(y_i))\}_{i=1}^n$, $\{(x_i^{\text{val}}, y_i^{\text{val}})\}_{i=1}^{n_{\text{val}}} \leftarrow \{(x_i, \tilde{\psi}_{\mathcal{Y}}(y_i^{\text{val}}))\}_{i=1}^{n_{\text{val}}}$, where $\tilde{\psi}_{\mathcal{Y}}(y) = \widetilde{M} R_{\mathcal{Y}} k_Y{}^y$
   - $g_{\hat{W}} = \underset{g_W, W \in \mathcal{W}}{\arg\min} \ \frac{1}{n} \sum_{i=1}^n c\left(\left\| g_{\widetilde{E}} \circ g_{\hat{W}}(x_i) - \psi_{\mathcal{Y}}(y_i) \right\|_{\mathcal{H}_{\mathcal{Y}}}^2 \right)$

2. Inference.
   - $\{y_i^c\}_{i=1}^{n_c} \leftarrow \{\tilde{\psi}_{\mathcal{Y}}(y_i^c)\}_{i=1}^{n_c}$
   - $f_{\hat{\theta}}(x_i^{\text{te}}) = y_j^c$ where $j = \underset{1 \leq j \leq n_c}{\arg\min} \ c\left(\left\| g_{\widetilde{E}} \circ g_{\hat{W}}(x_i^{\text{te}}) - \psi_{\mathcal{Y}}(y_j^c) \right\|_{\mathcal{H}_{\mathcal{Y}}}^2 \right)$

Scalability ✓, loss ✓, expressiveness ✓!

Synthetic least squares regression

1) $n = 50\,000$, $\mathcal{X} = \mathbb{R}^{2\,000}$, $\mathcal{Y} = \mathbb{R}^{1\,000}$, $k_{\mathcal{Y}}$ linear kernel $\implies$ $\mathcal{H}_{\mathcal{Y}} = \mathcal{Y} = \mathbb{R}^{1\,000}$

Goal: build this dataset such that the outputs lie in **a subspace of $\mathcal{Y}$ of dimension $d = 50 < 1\,000$**

Synthetic least squares regression

1) $n = 50\,000$, $\mathcal{X} = \mathbb{R}^{2\,000}$, $\mathcal{Y} = \mathbb{R}^{1\,000}$, $k_{\mathcal{Y}}$ linear kernel $\implies$ $\mathcal{H}_{\mathcal{Y}} = \mathcal{Y} = \mathbb{R}^{1\,000}$

Goal: build this dataset such that the outputs lie in **a subspace of $\mathcal{Y}$ of dimension $d = 50 < 1\,000$**

2) Draw $H = (H_{ij})_{1 \leq i \leq d, 1 \leq j \leq 2\,000} \in \mathbb{R}^{d \times 2\,000}$ s.t. $H_{ij} \sim \mathcal{N}(0, 1)$, $x_i \sim \mathcal{N}(0, C_{\mathcal{X}})$, where $(\sigma_j(C_{\mathcal{X}}) = j^{-1/2})_{j=1}^{2\,000}$, $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 I_{1\,000})$ with $\sigma^2 = 0.01$,

$$y_i = \boldsymbol{U} H x_i + \varepsilon_i \,,$$

where $U = (u_1, \ldots, u_d) \in \mathbb{R}^{1\,000 \times d}$ and $(u_j)_{j=1}^{d}$ are d randomly drawn orthonormal vectors

Synthetic least squares regression: results

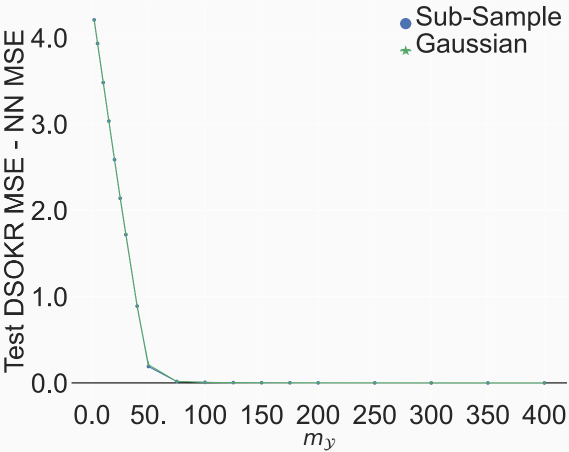


Figure 2: Difference between test MSE of DSOKR and NN w.r.t. $m_{\mathcal{Y}}$.

Text to molecule

ChEBI-20 dataset (Edwards et al., 2021)

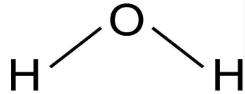
$n = 26\,408$, $n_{\mathbf{te}} = 3\,301$, $n_{\mathbf{c}} = 33\,010$

Inputs: texts (mean/median number of words per description is 55/51)

Outputs: molecules as graphs (mean/median number of atoms per molecule is 32/25)

Water is an oxygen hydride consisting of an oxygen atom that is covalently bonded to two hydrogen atoms.

$$\xrightarrow{f\ =\ ?}$$



Text to molecule

ChEBI-20 dataset (Edwards et al., 2021)

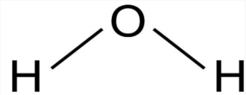
$n = 26\,408$, $n_{\mathrm{te}} = 3\,301$, $n_{\mathrm{c}} = 33\,010$

Inputs: texts (mean/median number of words per description is 55/51)

Outputs: molecules as graphs (mean/median number of atoms per molecule is 32/25)

Water is an oxygen hydride consisting of an oxygen atom that is covalently bonded to two hydrogen atoms.

$\xrightarrow{f \ = \ ?}$



Input neural network: SciBERT (transformer) (Beltagy et al., 2019)

Output kernel: cosine applied to Mol2vec (Jaeger et al., 2018) (for normalization)

Sketching: Sub-Sample and Gaussian, $m_{\mathcal{Y}} = 100$

Text to molecule: results

	Hits@1 ↑	Hits@10 ↑	MRR ↑
SISOKR	0.4%	2.8%	0.015
SciBERT Regression	16.8%	56.9%	0.298
CMAM - MLP	34.9%	84.2%	0.513
CMAM - GCN	33.2%	82.5%	0.495
CMAM - Ensemble (MLP×3)	39.8%	87.6%	0.562
CMAM - Ensemble (GCN×3)	39.0%	87.0%	0.551
CMAM - Ensemble (MLP×3 + GCN×3)	44.2%	**88.7**%	0.597
DSOKR - SubSample Sketch	48.2%	87.4%	0.624
DSOKR - Gaussian Sketch	49.0%	87.5%	0.630
DSOKR - Ensemble (SubSample×3)	**51.0**%	88.2%	**0.642**
DSOKR - Ensemble (Gaussian×3)	50.5%	87.9%	**0.642**
DSOKR - Ensemble (SubSample×3 + Gaussian×3)	50.0%	88.3%	0.640

Conclusion

Conclusion

Challenge	*p*-sparsified
1. Scalability	✓
2. Theory	✓
3. Loss	✓
4. Expressiveness	

- *p*-sparsified sketches: new sketching distributions for an optimal statistical/computational trade-off
- Beyond Nyström approximation with **data-independent** distribution
- Excess risk bounds of sketched vector-valued kernel machines with Lipschitz losses

Conclusion

Challenge	p-sparsified	SISOKR
1. Scalability	✓	✓
2. Theory	✓	✓
3. Loss	✓	
4. Expressiveness		

- SISOKR: sketching on both input/output kernels to accelerate both training/inference steps
- Sketching as a way to build orthogonal projectors onto low-dimensional subspace of the feature space
- Excess risk bound leading to a consistent theoretical analysis of SISOKR
- Experiments: SISOKR accelerates IOKR or make it tractable

Conclusion

Challenge	*p*-sparsified	SISOKR	DSOKR
1. Scalability	✓	✓	✓
2. Theory	✓	✓	
3. Loss	✓		✓
4. Expressiveness			✓

- DSOKR: sketching on the output kernel to unlock the use of Deep Neural Networks within OKR framework
- Various losses thanks to this basis approach
- Experiments: DSOKR outperforms SOTA method on a text-to-molecule dataset
- All codes publicly available

Perspectives

- Incoporate SISOKR and DSOKR in a Python package for structured prediction in collaboration with *HI! PARIS*

- **Excess risk bound for DSOKR**:
  - ▷ SISOKR's error decomposition
  - ▷ excess risk of MLP with ReLU activations (Schmidt-Hieber, 2017)

- **DSOKR for unsupervised learning**:
  - ▷ basis approach on both first and last layers
  - ▷ auto-encoder for **structured objects** (Laforgue et al., 2019)

- **Differentially private kernel methods**:
  - ▷ **data-independent** *p*-sparsified sketches distribution
  - ▷ add less noise to attain privacy

Acknowledgements

- **PhD advisors:** Florence d'Alché-Buc and Pierre Laforgue
- **Co-authors:** Luc Brogat-Motte and Junjie Yang

- Fast Kernel Methods for Generic Lipschitz Losses via *p*-Sparsified Sketches
  with P. Laforgue and F. d'Alché-Buc, TMLR 2023
- Sketch In, Sketch Out: Accelerating both Learning and Inference for Structured Prediction with Kernels
  with L. Brogat-Motte, P. Laforgue and F. d'Alché-Buc, AISTATS 2024
- Deep Sketched Output Kernel Regression for Structured Prediction
  with J. Yang, P. Laforgue and F. d'Alché-Buc, to appear in ECML PKDD 2024

References

Alaoui, A. and Mahoney, M. W. (2015). Fast randomized kernel ridge regression with statistical guarantees. In Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28.

Bartlett, P. L., Bousquet, O., and Mendelson, S. (2005). Local rademacher complexities. *Ann. Statist.*, 33(4):1497–1537.

References ii

Beltagy, I., Lo, K., and Cohan, A. (2019). SciBERT: A Pretrained Language Model for Scientific Text. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Boucheron, S., Lugosi, G., and Massart, P. (2013). *Concentration inequalities: A nonasymptotic theory of independence.* Oxford university press.

Brault, R., Heinonen, M., and Buc, F. (2016). Random fourier features for operator-valued kernels. In *Asian Conference on Machine Learning*, pages 110–125. PMLR.

References iii

Brouard, C., d'Alché-Buc, F., and Szafranski, M. (2011). Semi-supervised penalized output kernel regression for link prediction. In *International Conference on Machine Learning (ICML)*, pages 593–600.

Brouard, C., Shen, H., Dührkop, K., d'Alché-Buc, F., Böcker, S., and Rousu, J. (2016a). Fast metabolite identification with input output kernel regression. *Bioinformatics*, 32(12):28–36.

Brouard, C., Szafranski, M., and D'Alché-Buc, F. (2016b). Input output kernel regression: supervised and semi-supervised structured output prediction with operator-valued kernels. *The Journal of Machine Learning Research*, 17(1):6105–6152.

Caldarelli, E., Chatalic, A., Colomé, A., Molinari, C., Ocampo-Martinez, C., Torras, C., and Rosasco, L. (2024). Linear quadratic control of nonlinear systems with koopman operator learning and the nyström method.

References iv

Chen, Y. and Yang, Y. (2021). Accumulations of projections—a unified framework for random sketches in kernel ridge regression. In *International Conference on Artificial Intelligence and Statistics*, pages 2953–2961. PMLR.

Cherfaoui, F., Kadri, H., and Ralaivola, L. (2022). Scalable ridge leverage score sampling for the nyström method. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4163–4167.

Ciliberto, C., Rosasco, L., and Rudi, A. (2020). A general framework for consistent structured prediction with implicit loss embeddings. *J. Mach. Learn. Res.*, 21(98):1–67.

Cortes, C., Mohri, M., and Weston, J. (2005). A general regression technique for learning transductions. In *International Conference on Machine Learning (ICML)*, pages 153–160.

References v

Edwards, C., Zhai, C., and Ji, H. (2021). Text2Mol: Cross-Modal Molecule Retrieval with Natural Language Queries. In Moens, M.-F., Huang, X., Specia, L., and Yih, S. W.-t., editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 595–607, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

El Ahmad, T., Brogat-Motte, L., Laforgue, P., and d'Alché Buc, F. (2024). Sketch in, sketch out: Accelerating both learning and inference for structured prediction with kernels. In Dasgupta, S., Mandt, S., and Li, Y., editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 109–117. PMLR.

El Ahmad, T., Laforgue, P., and d'Alché Buc, F. (2023). Fast kernel methods for generic lipschitz losses via p-sparsified sketches. *Transactions on Machine Learning Research*.

References vi

El Ahmad, T., Yang, J., Laforgue, P., and d'Alché Buc, F. (2024). Deep sketched output kernel regression for structured prediction.

Geurts, P., Wehenkel, L., and d'Alché Buc, F. (2006). Kernelizing the output of tree-based methods. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 345–352, New York, NY, USA. Association for Computing Machinery.

Harrison Jr, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of environmental economics and management*, 5(1):81–102.

Jaeger, S., Fulle, S., and Turk, S. (2018). Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition. *Journal of Chemical Information and Modeling*, 58(1):27–35.

References vii

Johnson, W. B. and Lindenstrauss, J. (1984). Extensions of lipschitz mappings into a hilbert space 26. *Contemporary mathematics*, 26:28.

Kadri, H., Ghavamzadeh, M., and Preux, P. (2013). A generalized kernel approach to structured output learning. In Dasgupta, S. and McAllester, D., editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 471–479, Atlanta, Georgia, USA. PMLR.

Kadri, H., Rakotomamonjy, A., Preux, P., and Bach, F. (2012). Multiple operator-valued kernel learning. *Advances in Neural Information Processing Systems*, 25.

References viii

Katakis, I., Tsoumakas, G., and Vlahavas, I. (2008). Multilabel text classification for automated tag suggestion. In *Proceedings of the ECML/PKDD*, volume 18, page 5. Citeseer.

Kimeldorf, G. and Wahba, G. (1971). Some results on tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1):82–95.

Korba, A., Garcia, A., and d'Alché-Buc, F. (2018). A structured prediction approach for label ranking. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Lacotte, J. and Pilanci, M. (2022). Adaptive and oblivious randomized subspace methods for high-dimensional optimization: Sharp analysis and lower bounds. *IEEE Transactions on Information Theory*, 68(5):3281–3303.

References ix

Laforgue, P., Clémençon, S., and d'Alché-Buc, F. (2019). Autoencoding any data through kernel autoencoders. In Chaudhuri, K. and Sugiyama, M., editors, *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 1061–1069. PMLR.

Laforgue, P., Lambert, A., Brogat-Motte, L., and d'Alché Buc, F. (2020). Duality in rkhss with infinite dimensional outputs: Application to robust losses. In *International Conference on Machine Learning*, pages 5598–5607. PMLR.

Li, Z., Ton, J.-F., Oglic, D., and Sejdinovic, D. (2021). Towards a unified analysis of random fourier features. *Journal of Machine Learning Research*, 22(108):1–51.

References x

Meanti, G., Chatalic, A., Kostic, V. R., Novelli, P., massimiliano pontil, and Rosasco, L. (2023). Estimating koopman operators with sketching to provably learn large scale dynamical systems. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Moen, E., Handegard, N. O., Allken, V., Albert, O. T., Harbitz, A., and Malde, K. (2018). Automatic interpretation of otoliths using deep learning. *PLoS One*, 13(12):e0204713.

Nikolentzos, G., Meladianos, P., Limnios, S., and Vazirgiannis, M. (2018). A Degeneracy Framework for Graph Similarity. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 2595–2601. International Joint Conferences on Artificial Intelligence Organization.

References xi

Ordoñez, A., Eikvil, L., Salberg, A.-B., Harbitz, A., Murray, S. M., and Kampffmeyer, M. C. (2020). Explaining decisions of deep neural networks used for fish age prediction. *PloS one*, 15(6):e0235013.

Rahimi, A. and Recht, B. (2007). Random features for large scale kernel machines. *NIPS*, 20:1177–1184.

Ramakrishnan, R., Dral, P. O., Rupp, M., and von Lilienfeld, O. A. (2014). Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1. Publisher: Nature Publishing Group.

Ruddigkeit, L., van Deursen, R., Blum, L. C., and Reymond, J.-L. (2012). Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *Journal of Chemical Information and Modeling*, 52(11):2864–2875. Publisher: American Chemical Society.

References xii

Rudi, A., Calandriello, D., Carratino, L., and Rosasco, L. (2018). On fast leverage score sampling and optimal learning. In *NeurIPS*.

Rudi, A., Camoriano, R., and Rosasco, L. (2015). Less is more: Nyström computational regularization. *Advances in Neural Information Processing Systems*, 28.

Schmidt-Hieber, J. (2017). Nonparametric regression using deep neural networks with relu activation function. *Annals of Statistics*, 48.

Schymanski, E., Ruttkies, C., and Krauss, M. e. a. (2017). Critical assessment of small molecule identification 2016: automated methods. *J Cheminform*, 9:22.

References xiii

Sriperumbudur, B. K. and Szabó, Z. (2015). Optimal rates for random fourier features. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 1144–1152, Cambridge, MA, USA. MIT Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is All you Need. In *Advances in Neural Information Processing Systems (NIPS)*, pages 5998–6008.

Weston, J., Chapelle, O., Vapnik, V., Elisseeff, A., and Schölkopf, B. (2003). Kernel dependency estimation. In Becker, S., Thrun, S., and Obermayer, K., editors, *Advances in Neural Information Processing Systems 15*, pages 897–904. MIT Press.

References xiv

Williams, C. and Seeger, M. (2001). Using the nyström method to speed up kernel machines. In Leen, T., Dietterich, T., and Tresp, V., editors, *Advances in Neural Information Processing Systems*, volume 13, pages 682–688. MIT Press.

Yang, T., Li, Y.-f., Mahdavi, M., Jin, R., and Zhou, Z.-H. (2012). Nyström method vs random fourier features: A theoretical and empirical comparison. In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.

Yang, Y., Pilanci, M., Wainwright, M. J., et al. (2017). Randomized sketches for kernels: Fast and optimal nonparametric regression. *The Annals of Statistics*, 45(3):991–1023.

Sub-sampling is random projection

Let $n = 5$, $X = \{x_1, \ldots, x_5\}$, $k_X^x = (k_{\mathcal{X}}(x, x_1), \ldots, k_{\mathcal{X}}(x, x_5))$, $m_{\mathcal{X}} = 2$ and
$R_{\mathcal{X}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$

$$K_{m_{\mathcal{X}} n} = \begin{pmatrix} k_X^{x_1} \\ k_X^{x_4} \end{pmatrix} = R_{\mathcal{X}} K \quad \text{and} \quad K_{m_{\mathcal{X}} m_{\mathcal{X}}} = \begin{pmatrix} k_{\mathcal{X}}(x_1, x_1) & k_{\mathcal{X}}(x_1, x_4) \\ k_{\mathcal{X}}(x_4, x_1) & k_{\mathcal{X}}(x_4, x_4) \end{pmatrix} = R_{\mathcal{X}} K_X R_{\mathcal{X}}^\top$$

$\tilde{f} = \sum_{i=1}^{m_{\mathcal{X}}} k_{\mathcal{X}}(\cdot, \tilde{x}_i) \tilde{\gamma}_j = \sum_{j=1}^{n} k_{\mathcal{X}}(\cdot, \tilde{x}_i) [R_{\mathcal{X}}^\top \tilde{\gamma}]_i$, where

$$\tilde{\gamma} = \operatorname*{arg\,min}_{\gamma \in \mathbb{R}^{\mathcal{X}}} \frac{1}{n} \sum_{i=1}^{n} \Delta\left(\left[K_X R_{\mathcal{X}}^\top \gamma \right]_{i:}^\top, y_i \right) + \frac{\lambda}{2} \gamma_{\mathcal{X}}^\top K_X R_{\mathcal{X}}^\top \gamma \,.$$

Could we use other random matrix distributions?

Which property should sketching distributions satisfy?

- $K_X/n = UDU^\top$
- $D = \mathbf{diag}\left(\sigma_1(K_X), \ldots, \sigma_n(K_X)\right)$ where $\sigma_1(K_X) \geq \ldots \geq \sigma_n(K_X)$
- δ_n^2 the lowest value s. t. $\psi(\delta_n) = \left(\frac{1}{n}\sum_{i=1}^n \mathbf{min}(\delta_n^2, \sigma_i(K_X))\right)^{1/2} \leq \delta_n^2$ (Bartlett et al., 2005)
- $d_n = \mathbf{min}\ \left\{j \in \{1, \ldots, n\} \colon \sigma_j(K_X) \leq \delta_n^2\right\}$

Definition (K_X-satisfiability (Yang et al., 2017))

Let $c > 0$ independent of n. Let $U_1 \in \mathbb{R}^{n \times d_n}$ and $U_2 \in \mathbb{R}^{n \times (n - d_n)}$ be the left and right blocks of matrix U previously defined, and $D_2 = \mathbf{diag}\left(\sigma_{d_n+1}(K_X), \ldots, \sigma_n(K_X)\right)$. A sketch matrix $R_{\mathcal{X}}$ is said to be K_X-satisfiable for c if $R_{\mathcal{X}}$ is such that

$$\left\|\left(R_{\mathcal{X}} U_1\right)^\top R_{\mathcal{X}} U_1 - I_{d_n}\right\|_{\mathbf{op}} \leq 1/2\,, \qquad \text{and} \qquad \left\|R_{\mathcal{X}} U_2 D_2^{1/2}\right\|_{\mathbf{op}} \leq c\delta_n\,.$$

Intuition: $R_{\mathcal{X}}$ is K_X-satisfiable $\implies$ isometry on the largest eigenvectors of K_X/n and small operator norm on the smallest eigenvectors

Previous work

Settings in Yang et al. (2017):

- $d = 1 \implies$ **scalar regression only**
- $\Delta(y, y') = (y - y')^2 \implies$ **KRR only**
- Focus on the squared $L^2(\mathbb{P}_n)$ error, i.e., $\left\|\tilde{f} - f^\star\right\|_n^2 = \frac{1}{n}\sum_{i=1}^n \left(\tilde{f}(x_i) - f^\star(x_i)\right)^2 \implies$ **not excess risk in expectation**

Yang et al. (2017, Theorem 2): If $f^* \in \mathcal{H}$, then for any $\lambda \geq 2\delta_n^2$, with a probability greater than $1 - c_1 e^{-c_2 n \delta_n^2}$

$$\left\|\tilde{f} - f^*\right\|_n^2 \leq c_u \left(\lambda + \delta_n^2\right) \,, \tag{1}$$

where c_u only depends on $\|f^*\|_{\mathcal{H}}$.

Assumptions

A. 1: Expected risk is minimized over $\mathcal{H}$ at $f_{\mathcal{H}} = \operatorname{arg\,inf}_{f \in \mathcal{H}} \; \mathbb{E}\left[\Delta\left(f\left(X\right), Y\right)\right]$.

A. 2: The hypothesis set considered is the unit ball $\mathcal{B}\left(\mathcal{H}\right)$ of $\mathcal{H}$.

A. 3: $\forall\, y \in \mathbb{R}^d$, $z \mapsto \Delta(z, y)$ is L-Lipschitz over $\mathcal{H}(\mathcal{X}) = \{f(x) : f \in \mathcal{H},\, x \in \mathcal{X}\}$.

A. 4: $\exists\, \kappa_{\mathcal{X}} > 0$ s. t. $k_{\mathcal{X}}(x, x) \leq \kappa_{\mathcal{X}}, \forall\, x \in \mathcal{X}$ and M is non-singular.

A. 5: The sketching matrix $R_{\mathcal{X}}$ is K_X-satisfiable for a $c > 0$ independent of n.

Excess risk bound

Theorem

*Under **Asm. 1, 2, 3, 4 and 5**, let $C = 1 + \sqrt{6}c$, for any $\delta \in (0, 1)$, then with probability at least $1 - \delta$,*

$$\mathbb{E}\left[\Delta_{\tilde{f}}\right] \leq \mathbb{E}\left[\Delta_{f_{\mathcal{H}}}\right] + LC\sqrt{\lambda + \|M\|_{\mathrm{op}}\,\delta_n^2} + \frac{\lambda}{2} + 8L\sqrt{\frac{\kappa_{\mathcal{X}}\,\mathrm{Tr}\,(M)}{n}} + 2\sqrt{\frac{8\log(4/\delta)}{n}}\,.$$

If $\Delta(z, y) = \|z - y\|_2^2 / 2$ and $\mathcal{Y} \subset \mathcal{B}\left(\mathbb{R}^d\right)$, then with probability at least $1 - \delta$,

$$\mathbb{E}\left[\Delta_{\tilde{f}}\right] \leq \mathbb{E}\left[\Delta_{f_{\mathcal{H}}}\right] + \left(C^2 + \frac{1}{2}\right)\lambda + C^2\|M\|_{\mathrm{op}}\,\delta_n^2 + 8\,\mathrm{Tr}\,(M)^{1/2}\,\frac{\kappa_{\mathcal{X}}\,\|M\|_{\mathrm{op}}^{1/2} + \kappa_{\mathcal{X}}^{1/2}}{\sqrt{n}} + 2\sqrt{\frac{8\log(4/\delta)}{n}}\,.$$

Sketch of proof: error decomposition

$$\begin{aligned}\mathbb{E}[\Delta_{\tilde{f}}] - \mathbb{E}[\Delta_{f_{\mathcal{H}}}] &= \mathbb{E}_{(X,Y)\sim\rho}[\Delta(\tilde{f}(X), Y)] - \frac{1}{n}\sum_{i=1}^{n}\Delta(\tilde{f}(x_i), y_i) \leftarrow \text{gen. error}\\ &+ \frac{1}{n}\sum_{i=1}^{n}\Delta(\tilde{f}(x_i), y_i) - \frac{1}{n}\sum_{i=1}^{n}\Delta(f_{\mathcal{H}}(x_i), y_i) \leftarrow \text{approx. error}\\ &+ \frac{1}{n}\sum_{i=1}^{n}\Delta(f_{\mathcal{H}}(x_i), y_i) - \mathbb{E}_{(X,Y)\sim\rho}[\Delta(f_{\mathcal{H}}(X), Y)] \leftarrow \text{gen. error}\end{aligned}$$

Sketch of proof: approximation error

Let $\mathcal{H}_{R_{\mathcal{X}}} = \left\{ f = \sum_{i=1}^{n} k_{\mathcal{X}}(\cdot, x_i) M \left[R_{\mathcal{X}}^{\top} \tilde{\Gamma} \right]_i \mid \gamma \in \mathbb{R}^{m_{\mathcal{X}} \times d} \right\}$

$$\begin{aligned}
\frac{1}{n} \sum_{i=1}^{n} \Delta(\tilde{f}(x_i), y_i) - \frac{1}{n} \sum_{i=1}^{n} \Delta(f_{\mathcal{H}}(x_i), y_i) \\
&\leq \inf_{\substack{f \in \mathcal{H}_{R_{\mathcal{X}}} \\ \|f\|_{\mathcal{H}} \leq 1}} \frac{L}{n} \sum_{i=1}^{n} \| f(x_i) - f_{\mathcal{H}}(x_i) \|_2 + \frac{\lambda}{2} \leftarrow \text{A. 2} \\
&\leq L \inf_{\substack{f \in \mathcal{H}_{R_{\mathcal{X}}} \\ \|f\|_{\mathcal{H}} \leq 1}} \sqrt{\frac{1}{n} \sum_{i=1}^{n} \| f(x_i) - f_{\mathcal{H}}(x_i) \|_2^2} + \frac{\lambda}{2} \leftarrow \text{Jensen}
\end{aligned}$$

K_X-satisfiability of p-sparsified sketches

Theorem (El Ahmad et al., 2023)

Let $R_{\mathcal{X}}$ be a p-sparsified sketch. Then, there are some universal constants $C_0, C_1 > 0$ and a constant $c(p)$, increasing with p, such that for $m_{\mathcal{X}} \geq \max\left(C_0 d_n / p^2, \delta_n^2 n\right)$ and with a probability at least $1 - C_1 e^{-m_{\mathcal{X}} c(p)}$, the sketch $R_{\mathcal{X}}$ is K_X-satisfiable for $c = \frac{2}{\sqrt{p}}\left(1 + \sqrt{\log(5)}\right) + 1$.

Intuitive behavior of p:

- $p = 1$: we recover Yang et al. (2017)'s result for Gaussian sketching
- the larger it is, the denser S is, and the more likely $R_{\mathcal{X}}$ is K_X-satisfiable
- the smaller it is, the larger $m_{\mathcal{X}}$ is needed

Joint quantile regression on real data

- Boston dataset (Harrison Jr and Rubinfeld, 1978): house price prediction, $n = 506$
- Otoliths dataset (Moen et al., 2018; Ordoñez et al., 2020): fish age prediction, $n = 3\,780$

Quantile levels to predict: $(0.1, 0.3, 0.5, 0.7, 0.9)$

Table 3: Empirical test pinball and crossing loss and training times (in sec) without sketching and with sketching ($m_{\mathcal{X}} = 50$).

Dataset	Metrics	w/o Sketch	$20/n$-SR	$20/n$-SG	Acc. $m = 20$
Boston	Pinball loss	**51.28 ± 0.67**	54.75 ± 0.74	54.78 ± 0.72	54.73 ± 0.75
	Crossing loss	0.34 ± 0.13	0.26 ± 0.08	**0.11 ± 0.07**	0.15 ± 0.07
	Training time	6.97 ± 0.25	1.43 ± 0.07	**1.38 ± 0.08**	1.48 ± 0.05
otoliths	Pinball loss	2.78	2.66 ± 0.02	**2.64 ± 0.02**	2.67 ± 0.03
	Crossing loss	**5.18**	5.46 ± 0.06	5.43 ± 0.05	5.46 ± 0.06
	Training time	606.8	20.4 ± 0.5	**20.0 ± 0.3**	22.1 ± 0.4

Complexity of IOKR and SISOKR for various types of sketching

Table 4: Time and space complexities at training and inference for the IOKR and SISOKR algorithms with sub-sampling, p-sparsified ($p \in (0, 1]$) or Gaussian sketching, for a test set of size n_{te} and a candidate set of size n_c, such that $n_{te} \leq m_{\mathcal{X}}, m_{\mathcal{Y}} < n \leq n_c$. For the sake of simplicity, we omit the $\mathcal{O}(\cdot)$ in the following.

	Training		Inference	
Method	Time	Space	Time	Space
IOKR	n^3	n^2	$n_{te} n n_c$	$n n_c$
SISOKR (sub-sampling)	$\max(m_{\mathcal{X}}, m_{\mathcal{Y}})n$	$\max(m_{\mathcal{X}}, m_{\mathcal{Y}})n$	$n_{te} m_{\mathcal{Y}} n_c$	$m_{\mathcal{Y}} n_c$
SISOKR (p-sparsified)	$\max(m_{\mathcal{X}}, m_{\mathcal{Y}})^2 pn$	$\max(m_{\mathcal{X}}, m_{\mathcal{Y}})pn$	$\max(n_{te}, n m_{\mathcal{Y}} p) m_{\mathcal{Y}} n_c$	$npm_{\mathcal{Y}} n_c$
SISOKR (Gaussian)	$\max(m_{\mathcal{X}}, m_{\mathcal{Y}})n^2$	n^2	$n m_{\mathcal{Y}} n_c$	$n n_c$

Sketching sizes selection strategy

Goal: set the minimal value of $m_{\mathcal{Z}}$ s.t. it captures the information contained in the empirical covariance operator $\widehat{C}_Z = \frac{1}{n} \sum_{i=1}^{n} \psi_{\mathcal{Z}}(z_i) \otimes \psi_{\mathcal{Z}}(z_i)$

However: computing the SVD of $\widehat{C}_Z$ is costing, i.e. $\mathcal{O}(n^3)$ in time.

1. Approximate leverage scores of $\widehat{C}_X$ and $\widehat{C}_Y$

2. Empirical approach: given training/inference budgets of time $T_{\text{tr}}/T_{\text{inf}}$, set low $m_{\mathcal{X}}$ and $m_{\mathcal{Y}}$ and evaluate the performance of $\tilde{f}$ until reaching one of the following condition:

- convergence of the performance of $\tilde{f}$
- training time attains T_{tr} or inference time attains T_{te}

Selection of $m_{\mathcal{X}}$

$\tilde{h}^{\text{SIOKR}}(x) = \sum_{i=1}^{n} \tilde{\alpha}_i^{\text{SIOKR}}(x)\, \psi_{\mathcal{Y}}(y_i)$ where

$$\tilde{\alpha}^{\text{SIOKR}}(x) = K_X R_{\mathcal{X}}^\top (R_{\mathcal{X}} K_X^2 R_{\mathcal{X}}^\top + n\lambda R_{\mathcal{X}} K_X R_{\mathcal{X}}^\top)^\dagger$$

Set the optimal $m_{\mathcal{X}}$ according to a training budget of time $T_{\mathbf{tr}}$ and the performance of $\tilde{h}^{\text{SIOKR}}$ in terms of surrogate regression error on the validation set, i.e. minimizing

$$\sum_{i=1}^{n_{\text{val}}} \left\| \tilde{h}^{\text{SIOKR}}(x_i^{\text{val}}) - \psi_{\mathcal{Y}}(y_i^{\text{val}}) \right\|_{\mathcal{H}_{\mathcal{Y}}}^2$$

$$= \sum_{i=1}^{n_{\text{val}}} \tilde{\alpha}^{\text{SIOKR}}\left(x_i^{\text{val}}\right)^\top K_Y \tilde{\alpha}^{\text{SIOKR}}\left(x_i^{\text{val}}\right) - 2\tilde{\alpha}^{\text{SIOKR}}\left(x_i^{\text{val}}\right)^\top k_Y^{y_i^{\text{val}}} + k_{\mathcal{Y}}(y_i^{\text{val}}, y_i^{\text{val}})$$

$\Longrightarrow$ allows to cope with the inference phase

Selection of $m_{\mathcal{Y}}$

Set the optimal $m_{\mathcal{Y}}$ according to an inference budget of time T_{inf} and the performance of the *perfect* h estimator on the validation set, i.e.

$$h : (x, y) \mapsto \widetilde{P}_Y \psi_{\mathcal{Y}}(y)$$

$$f(x_i^{\mathsf{val}}) = y_j^{\mathsf{c}} \quad \text{where} \quad j = \operatorname*{arg\,max}_{1 \leq j \leq n_{\mathsf{c}}} [K_Y^{\mathsf{val,tr}} R_{\mathcal{Y}}^\top \widetilde{K}_Y^\dagger R_{\mathcal{Y}} K_Y^{\mathsf{tr,c}}]_{ij}$$

$\implies$ allows to cope with the training phase

Theory: previous works and differences

Rudi et al. (2015):

1. **scalar** kernel Ridge regression
2. sketching **only** applied in the **input** feature space
3. **Nyström** approximation with **uniform** or **approximate leverage scores** sampling

Ciliberto et al. (2020):

1. **vector-valued** kernel Ridge regression, with possibly infinite-dimensional outputs
2. **no approximation** considered

This work (El Ahmad et al., 2024):

1. **vector-valued** kernel Ridge regression, with possibly infinite-dimensional outputs
2. sketching applied in **both** the **input and output** feature space
3. generic **sub-Gaussian** sketches

Related recent works on Koopman operators: (Meanti et al., 2023; Caldarelli et al., 2024)

SISOKR excess risk bound

Theorem (El Ahmad et al., 2024)

Let $\delta \in [0,1]$, $n \in \mathbb{N}$ sufficiently large such that $\lambda = n^{-1/(1+\gamma_{\mathcal{X}})} \geq \frac{9\kappa_{\mathcal{X}}^2}{n} \log(\frac{n}{\delta})$. Under Asm. 1, 2, 3 and 4, the following holds with probability at least $1-\delta$

$$\mathbb{E}[\|\tilde{h}(x) - h^*(x)\|_{\mathcal{H}_{\mathcal{Y}}}^2]^{\frac{1}{2}} \leq S(n) + c_2 A_{\rho_x}^{\psi_{\mathcal{X}}}(\widetilde{P}_X) + A_{\rho_y}^{\psi_{\mathcal{Y}}}(\widetilde{P}_Y)$$

where

$S(n) = c_1 \log(4/\delta) n^{-\frac{1}{2(1+\gamma_{\mathcal{X}})}}$ (regression error)

$A_{\rho_z}^{\psi_{\mathcal{Z}}}(\widetilde{P}_Z) = \mathbb{E}_z[\|(\widetilde{P}_Z - I_{\mathcal{H}_{\mathcal{Z}}})\psi_{\mathcal{Z}}(z)\|_{\mathcal{H}_{\mathcal{Z}}}^2]^{\frac{1}{2}}$ (sketching reconstruction error)

and $c_1, c_2 > 0$ are constants independent of n and δ defined in the proofs.

Sub-Gaussian sketching reconstruction error

Theorem (El Ahmad et al., 2024)

Under Asm. 1, 2, 3 and 4, for $\delta \in (0, 1/e]$, $n \in \mathbb{N}$ sufficiently large such that $\frac{9}{n}\log(n/\delta) \leq n^{-\frac{1}{1+\gamma_{\mathcal{Z}}}} \leq \|C_{\mathcal{Z}}\|_{\mathrm{op}}/2$, then if

$$m_{\mathcal{Z}} \geq c_4 \max\left(\nu_{\mathcal{Z}}^2 n^{\frac{\gamma_{\mathcal{Z}}+\mu_{\mathcal{Z}}}{1+\gamma_{\mathcal{Z}}}}, \nu_{\mathcal{Z}}^4 \log\left(1/\delta\right)\right),$$

then with probability $1-\delta$

$$\mathbb{E}_z[\|(\widetilde{P}_Z - I_{\mathcal{H}_{\mathcal{Z}}})\psi_{\mathcal{Z}}(z)\|_{\mathcal{H}_{\mathcal{Z}}}^2] \leq c_3 n^{-\frac{1-\gamma_{\mathcal{Z}}}{(1+\gamma_{\mathcal{Z}})}}$$

where $c_3, c_4 > 0$ are constants independents of $n, m_{\mathcal{Z}}, \delta$ defined in the proofs.

Synthetic least squares regression

1) $n = 10\,000$, $\mathcal{X} = \mathcal{Y} = \mathbb{R}^d$, $d = 300$, $k_{\mathcal{X}}$ and $k_{\mathcal{Y}}$ linear kernels $\implies$ $\mathcal{H}_{\mathcal{X}} = \mathcal{H}_{\mathcal{Y}} = \mathbb{R}^d$

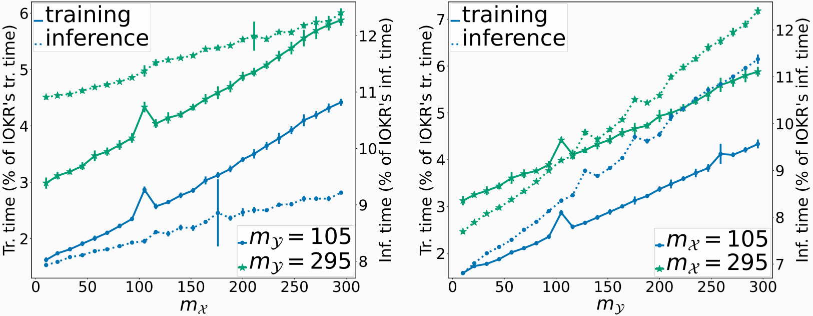
2) Construct covariance matrices $C_{\mathcal{X}}$ and E such that $\sigma_k(C_{\mathcal{X}}) = k^{-3/2}$ and $\sigma_k(E) = 0.2k^{-1/10}$

3) Draw $H_0 \sim \mathcal{N}(0, I_d)$, and for $i \leq n$, $x_i \sim \mathcal{N}(0, C_{\mathcal{X}})$, $\epsilon_i \sim \mathcal{N}(0, E)$,

$$y_i = C_{\mathcal{X}} H_0 x_i + \epsilon_i$$

4) $20/n$-SR input and output sketches

Synthetic least squares regression



(a) Training and inference time w.r.t. $m_\mathcal{X}$ for $m_\mathcal{Y} \in \{105, 295\}$

(b) Training and inference time w.r.t. $m_\mathcal{Y}$ for $m_\mathcal{X} \in \{105, 295\}$

Synthetic least squares regression

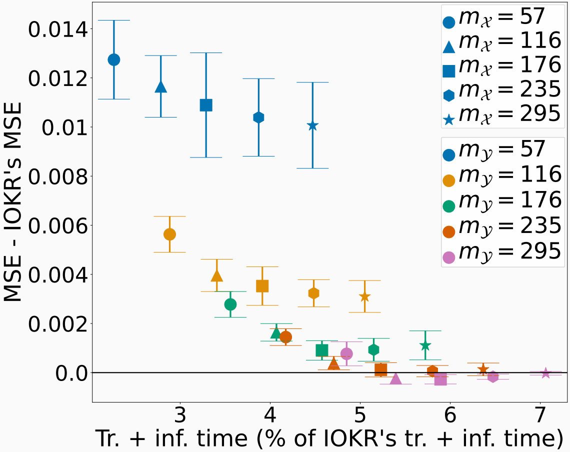


Figure 4: MSE w.r.t. learning time for different values of $m_{\mathcal{X}}$ and $m_{\mathcal{Y}}$

Multi-label classification

Bibtex and **Bookmarks** (Katakis et al., 2008): tag recommendation problems
Mediamill: detection of semantic concepts in a video

Table 5: Multi-label data sets description.

Data set	n	n_{te}	n_{features}	n_{labels}
Bibtex	4 880	2 515	1 836	159
Bookmarks	60 000	27 856	2 150	298
Mediamill	30 993	12 914	120	101

Multi-label classification: statistical performance

Table 6: F_1 scores on tag prediction from text data.

Method	Bibtex	Bookmarks	Mediamill
LR	37.2	30.7	NA
SPEN	42.2	34.4	NA
PRLR	44.2	34.9	NA
DVN	44.7	37.1	NA
SISOKR	44.1 ± 0.07	**39.3** ± 0.61	57.26 ± 0.04
ISOKR	44.8 ± 0.01	NA	58.02 ± 0.01
SIOKR	44.7 ± 0.09	39.1 ± 0.04	57.33 ± 0.04
IOKR	**44.9**	NA	**58.17**

Multi-label classification: computational performance

Table 7: Training/inference times (in seconds).

Method	Bibtex	Bookmarks	Mediamill
SISOKR	**1.41 ± 0.03 / 0.46 ± 0.01**	**118 ± 1.5 / 20 ± 0.2**	**66 ± 0.1 / 4 ± 0.01**
ISOKR	2.51 ± 0.06 / 0.58 ± 0.01	NA	636 ± 3.7 9 ± 0.2
SIOKR	1.99 ± 0.07 / 1.22 ± 0.03	354 ± 2.1 / 297 ± 2.1	199 ± 0.1 / 121 ± 0.02
IOKR	2.54 / 1.18	NA	621 / 204

Metabolite identification

Inputs: tandem mass spectra of metabolites

Outputs: molecular structures, i.e. fingerprints, encoded by binary vectors of length $d = 7593 \rightarrow$ **probability product kernel**

$n = 5\,579$ and **each molecule is associated with a specific candidate set**: **median size = 292** and **largest = 36 918** fingerprints $\rightarrow$ **Gaussian-Tanimoto kernel**

Method	kernel loss	Top-1 | 5 | 10 accuracies	training	inference
SPEN	0.537 ± 0.008	25.9% | 54.1% | 64.3%	NA	NA
SISOKR	0.566 ± 0.007	25.1% | 54.2% | 64.7%	4.05 ± 0.05	$\mathbf{1112 \pm 29}$
ISOKR	0.509 ± 0.009	28.0% | 58.9% | 68.9%	6.25 ± 50.31	1133 ± 32
SIOKR	0.492 ± 0.008	29.5% | 61.3% | 70.9%	$\mathbf{1.25 \pm 0.02}$	1179 ± 37
IOKR	$\mathbf{0.486 \pm 0.008}$	**29.6% | 61.6% | 71.4%**	3.54 ± 0.15	1191 ± 38

OKR with the basis approach: beyond the square loss

Let $\Delta : (y, y') \mapsto c\left(\|\psi_{\mathcal{Y}}(y) - \psi_{\mathcal{Y}}(y')\|^2_{\mathcal{H}_{\mathcal{Y}}}\right)$ with $c : \mathbb{R} \to \mathbb{R}$ non-decreasing and at least sub-differentiable, then for $l(W; x, y) = \|g_E \circ g_W(x) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}}$

$$\frac{\partial}{\partial W}\, c(l(W; x, y)) = c'\,(l(W; x, y)) \left(\frac{\partial}{\partial W}\, \|g_W(x)\|_2^2 - 2\frac{\partial}{\partial W}\, \tilde{\psi}_{\mathcal{Y}}(y)^\top g_W(x)\right)$$

For IOKR: let $k_{\mathcal{X}} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and $g_W : x \mapsto \hat{W}^\top k_X^x$ where

$$\hat{W} = \underset{W \in \mathbb{R}^{n \times p_X}}{\arg\min}\ \frac{1}{n} \sum_{i=1}^{n} c\left({k_X^{x_i}}^\top W W^\top k_X^{x_i} - 2{k_X^{x_i}}^\top W \tilde{\psi}_{\mathcal{Y}}(y) + k_{\mathcal{Y}}(y, y)\right) + \lambda\, \mathrm{Tr}(K_X W W^\top)$$

DSOKR Inference: Ensemble Approach

Let $T > 1$, and for $1 \leq t \leq T$, let $R_{\mathcal{Y}_t}$ be a randomly drawn sketching matrix, $h_{\hat{\theta}_t} = g_{\tilde{E}_t} \circ g_{\hat{W}_t}$ denotes the trained DSOKR neural network based on $R_{\mathcal{Y}_t}$

$$f_{\hat{\theta}}^{\text{mean}}(x) = \underset{y \in \mathcal{Y}_c}{\arg\max} \sum_{t=1}^{T} \omega_t \, g_{\hat{W}_t}(x)^\top \tilde{\psi}_{\mathcal{Y}_t}(y) \quad \text{with} \quad \sum_{t=1}^{T} \omega_t = 1$$

or*

$$f_{\hat{\theta}}^{\max}(x) = \underset{y \in \mathcal{Y}_c}{\arg\max} \; \arg\max_{1 \leq t \leq T} \; g_{\hat{W}_t}(x)^\top \tilde{\psi}_{\mathcal{Y}_t}(y)$$

Sketching size selection strategy

Goal: set the minimal value of $m_{\mathcal{Y}}$ s.t. it captures the information contained in the empirical covariance operator $\widehat{C}_Y = \frac{1}{n}\sum_{i=1}^{n} \psi_{\mathcal{Y}}(y_i) \otimes \psi_{\mathcal{Y}}(y_i)$

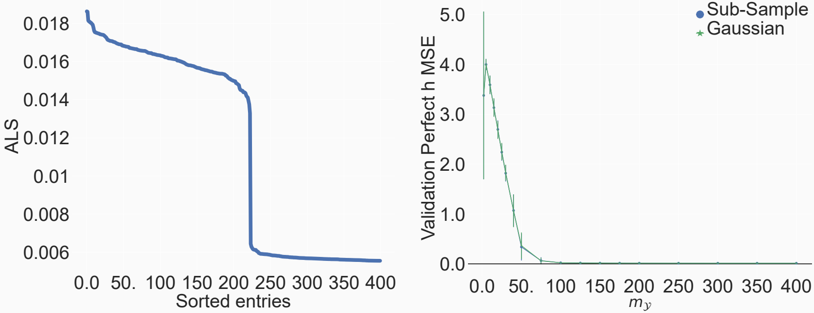
However: computing the SVD of $\widehat{C}_Y$ is costing, i.e. $\mathcal{O}(n^3)$ in time.

1. Approximate leverage scores of $\widehat{C}_Y$
2. Set the optimal $m_{\mathcal{Y}}$ according to the performance of the *perfect h* estimator on the validation set, i.e.

$$h : (x, y) \mapsto \sum_{j=1}^{p_Y} \langle \tilde{e}_j^Y, \psi_{\mathcal{Y}}(y) \rangle_{\mathcal{H}_{\mathcal{Y}}} \, \tilde{e}_j^Y = \sum_{j=1}^{p_Y} \tilde{\psi}_{\mathcal{Y}}(y)_j \, \tilde{e}_j^Y \, . \qquad (2)$$

$\implies$ **allows to cope with the neural net training phase!**

Synthetic least squares regression: sketching size selection



(a) Sorted 400 highest ALS.

(b) Validation MSE of *Perfect h* w.r.t. $m_{\mathcal{Y}}$.

Smiles to molecule

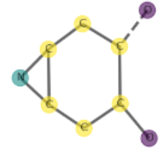
QM9 molecule dataset (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014)

$n = n_{\mathbf{c}} = 131\,382$, $n_{\mathbf{te}} = 2\,000$

Inputs: strings (smiles)

Outputs: graphs (molecules)

O=C1CC2NC2CC1O $\xrightarrow{f \ = \ ?}$



Input neural network: transformer (Vaswani et al., 2017)

Output kernel: core Weisfeiler-Lehman subtree kernel (CORE-WL) (Nikolentzos et al., 2018)

Input/output sketching: Sub-sample, $m_{\mathcal{Y}} = 3\,200$

Smiles to molecule: Perfect *h* strategy

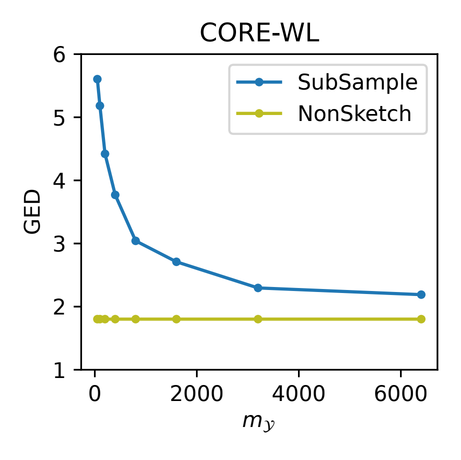
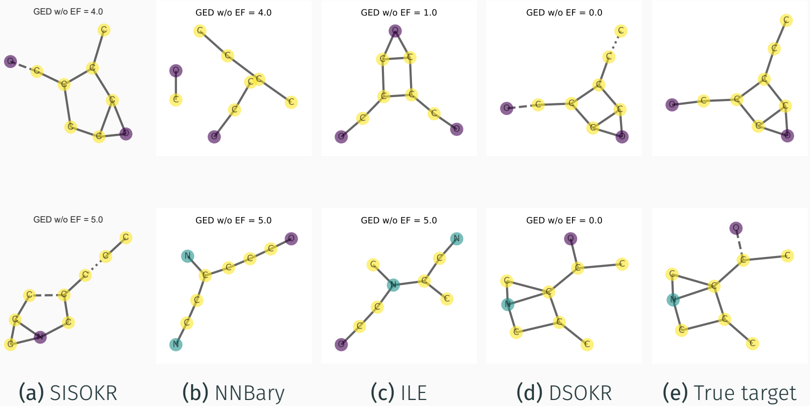


Figure 6: The GED w/ edge feature w.r.t. thes ketching size $m_{\mathcal{Y}}$ for *Perfect h* for the CORE-WL output kernel on SMI2Mol ($m_{\mathcal{Y}} > 6400$ is too costly computationally).

Smiles to molecule: results

	GED w/o edge feature ↓	GED w/ edge feature ↓
NNBary-FGW	5.115 ± 0.129	-
Sketched ILE-FGW	2.998 ± 0.253	-
IOKR	NA	NA
SIOKR	NA	NA
ISOKR	NA	NA
SISOKR	3.330 ± 0.080	4.192 ± 0.109
DSOKR	$\mathbf{1.951 \pm 0.074}$	$\mathbf{2.960 \pm 0.079}$

Smiles to Molecule: some nice figures



(a) SISOKR (b) NNBary (c) ILE (d) DSOKR (e) True target

Figure 7: Predicted molecules on the SMI2Mol dataset.

Text to molecule: Perfect h strategy

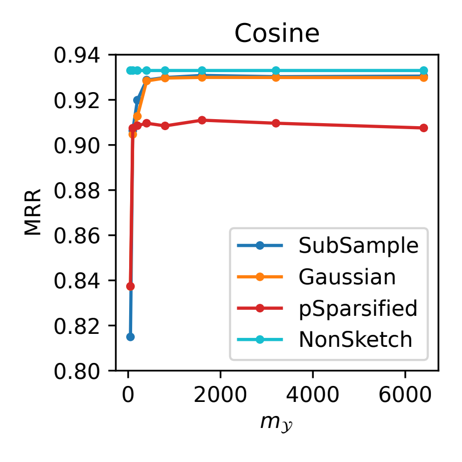


Figure 8: The MRR scores on ChEBI-20 validation set w.r.t. $m_\mathcal{Y}$ for *Perfect h* when the output kernel is Cosine on the ChEBI-20 dataset.