Deep Sketched Output Kernel Regression for Structured Prediction

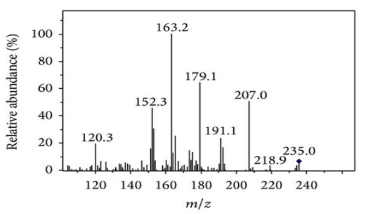
Tamim El Ahmad[⋆1], Junjie Yang[⋆1], Pierre Laforgue[2], Florence d'Alché-Buc[1]
⋆ Equal contribution
1 LTCI, Télécom Paris, Institut Polytechnique de Paris
2 Università degli Studi di Milano

June 7, 2024

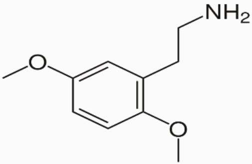
Structured Prediction

Goal: learn a mapping $f : \mathcal{X} \longrightarrow \mathcal{Y}$ with $\mathcal{Y}$ a space of structured objects (graphs, rankings, sequences, binary vectors, etc.).

MS/MS spectra



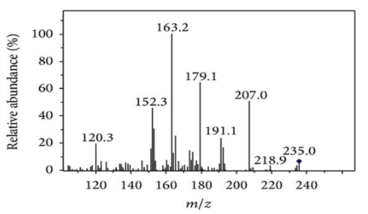
$f \; = \; ?$ $\longrightarrow$



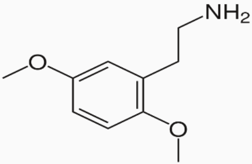
Structured Prediction

Goal: learn a mapping $f : \mathcal{X} \longrightarrow \mathcal{Y}$ with $\mathcal{Y}$ a space of structured objects (graphs, rankings, sequences, binary vectors, etc.).

MS/MS spectra



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



Existing works: Energy-based models (Lafferty et al., 2001; Taskar et al., 2003; Tsochantaridis et al., 2004; LeCun et al., 2007; Belanger and McCallum, 2016):

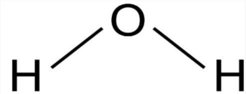
$$f(x) = \arg\min_{y \in \mathcal{Y}} E(x, y) \tag{1}$$

Structured Prediction with complex inputs

Goal of this work: solve structured prediction tasks with **complex inputs** such as texts

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

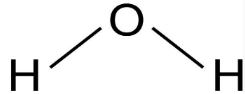
$$\xrightarrow{f \;=\; ?}$$



Structured Prediction with complex inputs

Goal of this work: solve structured prediction tasks with **complex inputs** such as texts

Water is an oxygen hydride consisting of an oxygen atom that is covalently bonded to two hydrogen atoms. $\xrightarrow{f\ =\ ?}$



$\implies$ need of **expressive** models such as **deep neural networks**

Motivation

Build a versatile and expressive estimator able to tackle a wide variety of structured prediction tasks and learn representations from complex inputs.

Table of contents

Output Kernel Regression

Output Kernel Regression

Given a p.d. kernel $k : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ **defining a relevant similarity measure** and $\psi : y \in \mathcal{Y} \mapsto k(\cdot, y) \in \mathcal{H}$,

we define $\Delta(y, y') = \|\psi(y) - \psi(y')\|^2_{\mathcal{H}} = k(y, y) - 2k(y, y') + k(y', y')$ (Weston et al., 2003; Cortes et al., 2005), and solve

$$\min_{\theta \in \Theta} \ \mathbb{E}_{(X,Y)\sim\rho}[\|\psi(f_\theta(X)) - \psi(Y)\|^2_{\mathcal{H}}] \tag{2}$$

Output Kernel Regression

Given a p.d. kernel $k : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ **defining a relevant similarity measure** and $\psi : y \in \mathcal{Y} \mapsto k(\cdot, y) \in \mathcal{H}$,

we define $\Delta(y, y') = \|\psi(y) - \psi(y')\|_{\mathcal{H}}^2 = k(y, y) - 2k(y, y') + k(y', y')$ (Weston et al., 2003; Cortes et al., 2005), and solve

$$\min_{\theta \in \Theta} \mathbb{E}_{(X,Y)\sim\rho}[\|\psi(f_\theta(X)) - \psi(Y)\|_{\mathcal{H}}^2] \tag{3}$$

How to learn f_θ through ψ?

Output Kernel Regression

$$\mathcal{X} \xrightarrow[f_{\hat{\theta}} \approx \underset{f_\theta,\, \theta \in \Theta}{\arg\inf}\ \mathbb{E}_{x,y}\left[\|\psi(f_\theta(x)) - \psi(y)\|^2_{\mathcal{H}}\right]]{} \mathcal{Y}$$

Output Kernel Regression

$$
\begin{tikzcd}
 & \mathcal{H} \arrow[d, bend left=60, "d"] \\
\mathcal{X} \arrow[ur, "h_{\widehat{\theta}} \approx \underset{h_\theta,\, \theta\in\Theta}{\arg\inf}\ \mathbb{E}_{x,y}\left[\|h_\theta(x) - \psi(y)\|_{\mathcal{H}}^2\right]"] \arrow[r, "f_{\widehat{\theta}} = d \circ h_{\widehat{\theta}} = \underset{y\in\mathcal{Y}}{\arg\min}\ \underbrace{\|h_{\widehat{\theta}}(\,\cdot\,) - \psi(y)\|_{\mathcal{H}}^2}_{E(\cdot,\, y)}"'] & \mathcal{Y} \arrow[u, "\psi"']
\end{tikzcd}
$$

Output Kernel Regression

$$\begin{tikzcd}[column sep=huge, row sep=huge]
 & \mathcal{H} \arrow[d, bend left=60, "d"] \\
\mathcal{X} \arrow[ur, "h_{\widehat{\theta}} \approx \underset{h_\theta,\, \theta \in \Theta}{\arg\inf}\ \mathbb{E}_{x,y}\left[\|h_\theta(x) - \psi(y)\|^2_{\mathcal{H}}\right]"] \arrow[r, "f_{\widehat{\theta}} = d \circ h_{\widehat{\theta}} = \underset{y \in \mathcal{Y}}{\arg\min}\ \underbrace{\|h_{\widehat{\theta}}(\,\cdot\,) - \psi(y)\|^2_{\mathcal{H}}}_{E(\cdot,\, y)}"'] & \mathcal{Y} \arrow[u, "\psi"]
\end{tikzcd}$$

How to deal with implicit or infinite-dimensional output feature maps while using an input neural network?

Output Kernel Regression with Deep Learning: a basis approach

$$
\begin{tikzcd}[column sep=huge, row sep=huge]
 & \mathrm{span}\Big((e_i)_{i=1}^{p}\Big) \subsetneq \mathcal{H} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur, "{h_{\widehat{\theta}}(x) = g_E \circ g_{\widehat{W}}(x) = \sum_{i=1}^{p} g_{\widehat{W}}(x)\, e_i}"] \arrow[r, "{f_{\widehat{\theta}} = d \circ h_{\widehat{\theta}} = \underset{y \in \mathcal{Y}}{\arg\min}\, \underbrace{\|h_{\widehat{\theta}}(\,\cdot\,) - \psi(y)\|_{\mathcal{H}}^{2}}_{E(\cdot,\, y)}}"'] & \mathcal{Y} \arrow[u, "\psi"']
\end{tikzcd}
$$

$$
\begin{tikzcd}
& \mathrm{span}\left((e_i)_{i=1}^{p}\right) \subsetneq \mathcal{H} \arrow[d, bend left, "d"] \\
\mathcal{X} \arrow[ur, "h_{\widehat{\theta}}(x) = g_E \circ g_{\widehat{W}}(x) = \sum_{i=1}^{p} g_{\widehat{W}}(x)\, e_i"] \arrow[r, "f_{\widehat{\theta}} = d \circ h_{\widehat{\theta}} = \arg\min_{y \in \mathcal{Y}} \underbrace{\|h_{\widehat{\theta}}(\cdot) - \psi(y)\|_{\mathcal{H}}^{2}}_{E(\cdot, y)}"'] & \mathcal{Y} \arrow[u, "\psi"']
\end{tikzcd}
$$

How to build this base $\mathrm{span}\left((e_i)_{i=1}^{p}\right)$?

Deep Sketched Output Kernel Regression

Sketching: random linear projections

Sketching: random linear projections

Let $m \ll n$, $R \in \mathbb{R}^{m \times n}$ be a random matrix and n data $(y_i)_{i=1}^n \in \mathcal{Y}$

Sketching

Sketching: random linear projections

Let $m \ll n$, $R \in \mathbb{R}^{m \times n}$ be a random matrix and n data $(y_i)_{i=1}^n \in \mathcal{Y}$

Basic idea: The linear subspace of $\mathcal{H}$ is obtained by

$$\operatorname{span}\left(\left(\sum_{j=1}^{n} R_{ij}\psi(y_j)\right)_{i=1}^{m}\right) \tag{4}$$

Sketching

Sketching: random linear projections

Let $m \ll n$, $R \in \mathbb{R}^{m \times n}$ be a random matrix and n data $(y_i)_{i=1}^n \in \mathcal{Y}$

Basic idea: The linear subspace of $\mathcal{H}$ is obtained by

$$\operatorname{span}\left(\left(\sum_{j=1}^{n} R_{ij}\psi(y_j)\right)_{i=1}^{m}\right) \tag{4}$$

What is its orthonormal basis?

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n} \sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n}\sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n}\sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li}\psi(y_i)\right) \otimes \left(\sum_{j=1}^{n} R_{lj}\psi(y_j)\right) \in \mathcal{H}^{\mathcal{H}}$

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n} \sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n} \sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li} \psi(y_i)\right) \otimes \left(\sum_{j=1}^{n} R_{lj} \psi(y_j)\right) \in \mathcal{H}^{\mathcal{H}}$
- $K = (k(y_i, y_j))_{1 \le i,j \le n} \in \mathbb{R}^{n \times n}$

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n}\sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n}\sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li}\psi(y_i)\right) \otimes \left(\sum_{j=1}^{n} R_{lj}\psi(y_j)\right) \in \mathcal{H}^{\mathcal{H}}$
- $K = (k(y_i, y_j))_{1 \leq i,j \leq n} \in \mathbb{R}^{n \times n}$
- $\widetilde{K} = RKR^\top \in \mathbb{R}^{m \times m}$, and $\left\{\left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i\right), i \in [m]\right\}$ its eigenpairs

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n} \sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n} \sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li} \psi(y_i) \right) \otimes \left(\sum_{j=1}^{n} R_{lj} \psi(y_j) \right) \in \mathcal{H}^{\mathcal{H}}$
- $K = (k(y_i, y_j))_{1 \le i,j \le n} \in \mathbb{R}^{n \times n}$
- $\widetilde{K} = RKR^\top \in \mathbb{R}^{m \times m}$, and $\left\{ \left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i \right), i \in [m] \right\}$ its eigenpairs
- $p = \operatorname{rank}\left(\widetilde{K}\right)$, and for all $1 \le i \le p$,
  $\tilde{e}_i = \sqrt{\frac{n}{\sigma_i(\widetilde{K})}} \sum_{j=1}^{n} [R^\top \tilde{\mathbf{u}}_i]_j \psi(y_j) \in \mathcal{H}$

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n}\sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n}\sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li}\psi(y_i)\right) \otimes \left(\sum_{j=1}^{n} R_{lj}\psi(y_j)\right) \in \mathcal{H}^{\mathcal{H}}$
- $K = (k(y_i, y_j))_{1 \le i,j \le n} \in \mathbb{R}^{n \times n}$
- $\widetilde{K} = RKR^\top \in \mathbb{R}^{m \times m}$, and $\left\{\left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i\right), i \in [m]\right\}$ its eigenpairs
- $p = \text{rank}\left(\widetilde{K}\right)$, and for all $1 \le i \le p$,
  $\tilde{e}_i = \sqrt{\frac{n}{\sigma_i(\widetilde{K})}} \sum_{j=1}^{n} [R^\top \tilde{\mathbf{u}}_i]_j \psi(y_j) \in \mathcal{H}$

Proposition

The $\tilde{e}_i$s are the eigenfunctions, associated to the eigenvalues $\sigma_i(\widetilde{K})/n$, of $\widetilde{C}$.

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n}\sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n}\sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li}\psi(y_i)\right) \otimes \left(\sum_{j=1}^{n} R_{lj}\psi(y_j)\right) \in \mathcal{H}^{\mathcal{H}}$
- $K = (k(y_i, y_j))_{1\leq i,j\leq n} \in \mathbb{R}^{n\times n}$
- $\widetilde{K} = RKR^\top \in \mathbb{R}^{m\times m}$, and $\left\{\left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i\right), i \in [m]\right\}$ its eigenpairs
- $p = \text{rank}\left(\widetilde{K}\right)$, and for all $1 \leq i \leq p$,
  $\tilde{e}_i = \sqrt{\frac{n}{\sigma_i(\widetilde{K})}}\sum_{j=1}^{n}[R^\top \tilde{\mathbf{u}}_i]_j\psi(y_j) \in \mathcal{H}$

Proposition

The $\tilde{e}_i$s are the eigenfunctions, associated to the eigenvalues $\sigma_i(\widetilde{K})/n$, of $\widetilde{C}$.

Then, $\widetilde{E} = (\tilde{e}_1, \ldots, \tilde{e}_p)$ is an orthonormal basis of
$\text{span}\left(\left(\sum_{j=1}^{n} R_{ij}\psi(y_j)\right)_{i=1}^{m}\right)$.

Construction of the orthonormal basis

- $\widehat{C} = \frac{1}{n}\sum_{i=1}^{n} \psi(y_i) \otimes \psi(y_i) \in \mathcal{H}^{\mathcal{H}}$
- $\widetilde{C}_Z = \frac{1}{n}\sum_{l=1}^{m} \left(\sum_{i=1}^{n} R_{li}\psi(y_i)\right) \otimes \left(\sum_{j=1}^{n} R_{lj}\psi(y_j)\right) \in \mathcal{H}^{\mathcal{H}}$
- $K = (k(y_i, y_j))_{1\leq i,j\leq n} \in \mathbb{R}^{n\times n}$
- $\widetilde{K} = RKR^\top \in \mathbb{R}^{m\times m}$, and $\left\{\left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i\right), i \in [m]\right\}$ its eigenpairs
- $p = \mathrm{rank}\left(\widetilde{K}\right)$, and for all $1 \leq i \leq p$,
  $\tilde{e}_i = \sqrt{\frac{n}{\sigma_i(\widetilde{K})}} \sum_{j=1}^{n} [R^\top \tilde{\mathbf{u}}_i]_j \psi(y_j) \in \mathcal{H}$

Proposition

The $\tilde{e}_i$s are the eigenfunctions, associated to the eigenvalues $\sigma_i(\widetilde{K})/n$, of $\widetilde{C}$.

Then, $\widetilde{E} = (\tilde{e}_1, \ldots, \tilde{e}_p)$ is an orthonormal basis of $\mathrm{span}\left(\left(\sum_{j=1}^{n} R_{ij}\psi(y_j)\right)_{i=1}^{m}\right)$.

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

Solving the surrogate problem

$$\min_{W \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^{n} \| g_{\widetilde{E}} \circ g_W(x_i) - \psi(y_i) \|_{\mathcal{H}}^2 \tag{5}$$

Solving the surrogate problem

$$\min_{W \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^{n} \|g_{\widetilde{E}} \circ g_W(x_i) - \psi(y_i)\|_{\mathcal{H}}^2 \tag{5}$$

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

Solving the surrogate problem

$$\min_{W\in\mathcal{W}} \frac{1}{n}\sum_{i=1}^{n} \|g_{\widetilde{E}} \circ g_W(x_i) - \psi(y_i)\|_{\mathcal{H}}^2 \tag{5}$$

$$\begin{aligned}\left\|g_{\widetilde{E}} \circ g_W(x) - \psi(y)\right\|_{\mathcal{H}}^2 &= \left\|\sum_{i=1}^{p} g_W(x)_j \tilde{e}_j - \psi(y)\right\|_{\mathcal{H}}^2 \\ &= \sum_{i,j=1}^{p} g_W(x)_i g_W(x)_j \langle \tilde{e}_i, \tilde{e}_j\rangle_{\mathcal{H}} - 2\sum_{j=1}^{p} g_W(x)_j \langle \tilde{e}_j, \psi(y)\rangle_{\mathcal{H}} + k(y,y)\end{aligned}$$

Solving the surrogate problem

$$\min_{W \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^{n} \| g_{\widetilde{E}} \circ g_W(x_i) - \psi(y_i) \|_{\mathcal{H}}^2 \tag{5}$$

$$\begin{aligned}
\left\| g_{\widetilde{E}} \circ g_W(x) - \psi(y) \right\|_{\mathcal{H}}^2 &= \left\| \sum_{i=1}^{p} g_W(x)_j \tilde{e}_j - \psi(y) \right\|_{\mathcal{H}}^2 \\
&= \sum_{i,j=1}^{p} g_W(x)_i g_W(x)_j \langle \tilde{e}_i, \tilde{e}_j \rangle_{\mathcal{H}} - 2 \sum_{j=1}^{p} g_W(x)_j \langle \tilde{e}_j, \psi(y) \rangle_{\mathcal{H}} + k(y,y) \\
&= \| g_W(x) \|_2^2 - 2 g_W(x)^\top \tilde{\psi}(y) + k(y,y) \,,
\end{aligned}$$

where $\tilde{\psi}(y) = (\langle \tilde{e}_1, \psi(y) \rangle_{\mathcal{H}}, \dots, \langle \tilde{e}_p, \psi(y) \rangle_{\mathcal{H}})^\top = \widetilde{D}_p^{-1/2} \widetilde{U}_p^\top R k^Y \in \mathbb{R}^p$, $\widetilde{U}_p = (\tilde{\mathbf{u}}_1, \dots, \tilde{\mathbf{u}}_p)$, $\widetilde{D}_p = \operatorname{diag}(\sigma_1(\widetilde{K}), \dots, \sigma_p(\widetilde{K}))$, and $k^Y = (k(y, y_1), \dots, k(y, y_n))$.

Solving the surrogate problem

$$\min_{W\in\mathcal{W}} \frac{1}{n}\sum_{i=1}^{n} \|g_{\widetilde{E}} \circ g_W(x_i) - \psi(y_i)\|_{\mathcal{H}}^2 \tag{5}$$

$$\begin{aligned}
\left\|g_{\widetilde{E}} \circ g_W(x) - \psi(y)\right\|_{\mathcal{H}}^2 &= \left\|\sum_{i=1}^{p} g_W(x)_j \tilde{e}_j - \psi(y)\right\|_{\mathcal{H}}^2 \\
&= \sum_{i,j=1}^{p} g_W(x)_i g_W(x)_j \langle \tilde{e}_i, \tilde{e}_j\rangle_{\mathcal{H}} - 2\sum_{j=1}^{p} g_W(x)_j \langle \tilde{e}_j, \psi(y)\rangle_{\mathcal{H}} + k(y,y) \\
&= \|g_W(x)\|_2^2 - 2 g_W(x)^\top \tilde{\psi}(y) + k(y,y) \\
&= \left\|g_W(x) - \tilde{\psi}(y)\right\|_2^2 - \left\|\tilde{\psi}(y)\right\|_2^2 + k(y,y)\,,
\end{aligned}$$

where $\tilde{\psi}(y) = (\langle \tilde{e}_1, \psi(y)\rangle_{\mathcal{H}}, \dots, \langle \tilde{e}_p, \psi(y)\rangle_{\mathcal{H}})^\top = \widetilde{D}_p^{-1/2}\widetilde{U}_p^\top R k^y \in \mathbb{R}^p$, $\widetilde{U}_p = (\tilde{\mathbf{u}}_1, \dots, \tilde{\mathbf{u}}_p)$, $\widetilde{D}_p = \operatorname{diag}(\sigma_1(\widetilde{K}), \dots, \sigma_p(\widetilde{K}))$, and $k^y = (k(y,y_1), \dots, k(y,y_n))$.

Deep Sketched Output Kernel Regression: Inference

$$f_{\hat{\theta}}(x) = d \circ h_{th\hat{e}ta}(x) = \underset{y \in \mathcal{Y}}{\arg\min} \; \left\| h_{\hat{\theta}}(x) - \psi(y) \right\|_{\mathcal{H}}^2 =$$

$$\underset{y \in \mathcal{Y}}{\arg\min} \; k(y, y) - 2 g_{\hat{W}}(x)^\top \tilde{\psi}(y) = \underset{y \in \mathcal{Y}}{\arg\max} \; g_{\hat{W}}(x)^\top \tilde{\psi}(y)$$

Deep Sketched Output Kernel Regression: Inference

$$f_{\hat{\theta}}(x) = d \circ h_{th\hat{e}ta}(x) = \underset{y \in \mathcal{Y}}{\arg\min} \; \left\| h_{\hat{\theta}}(x) - \psi(y) \right\|_{\mathcal{H}}^2 =$$
$$\underset{y \in \mathcal{Y}}{\arg\min} \; k(y, y) - 2 g_{\hat{W}}(x)^\top \tilde{\psi}(y) = \underset{y \in \mathcal{Y}}{\arg\max} \; g_{\hat{W}}(x)^\top \tilde{\psi}(y)$$

- **Test set:** X_{te} of size n_{te}
- **Candidate set:** $\mathcal{Y}_c \subseteq \mathcal{Y}$ of size n_c

$$\underbrace{g_{\hat{W}}(X^{te})}_{n_{te} \times p} \underbrace{\tilde{\psi}(Y^c)^\top}_{p \times n_c} \tag{6}$$

DSOKR Inference: Ensemble Approach

Let $T > 1$, and for $1 \leq t \leq T$, let R_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_t

DSOKR Inference: Ensemble Approach

Let $T > 1$, and for $1 \leq t \leq T$, let R_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_t

$$f_{\hat{\theta}}^{\text{mean}}(x) = \underset{y \in \mathcal{Y}_c}{\arg\max} \sum_{t=1}^{T} \omega_t \, g_{\hat{W}_t}(x)^\top \tilde{\psi}_t(y) \quad \text{with} \quad \sum_{t=1}^{T} \omega_t = 1 \tag{7}$$

or

$$f_{\hat{\theta}}^{\max}(x) = \underset{y \in \mathcal{Y}_c}{\arg\max} \ \underset{1 \leq t \leq T}{\arg\max} \ g_{\hat{W}_t}(x)^\top \tilde{\psi}_t(y) \tag{8}$$

DSOKR: summary

1. Training. a. Computations for the basis $\widetilde{E}$.
   - SVD of $\widetilde{K} = RKR^\top \to \left\{\left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i\right), i \in [m]\right\}$
   - $\widetilde{\Omega} = \widetilde{D}_p^{-1/2}\widetilde{U}_p^\top \in \mathbb{R}^{p\times m}$, where $\widetilde{U}_p = (\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_p)$, $\widetilde{D}_p = \mathrm{diag}(\sigma_1(\widetilde{K}), \ldots, \sigma_p(\widetilde{K}))$

DSOKR: summary

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

- SVD of $\widetilde{K} = RKR^\top \to \left\{ \left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i \right), i \in [m] \right\}$
- $\widetilde{\Omega} = \widetilde{D}_p^{-1/2} \widetilde{U}_p^\top \in \mathbb{R}^{p \times m}$, where $\widetilde{U}_p = (\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_p)$, $\widetilde{D}_p = \mathrm{diag}(\sigma_1(\widetilde{K}), \ldots, \sigma_p(\widetilde{K}))$

1. Training. b. Solving the surrogate problem.

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

DSOKR: summary

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

- SVD of $\widetilde{K} = RKR^\top \to \left\{ \left(\sigma_i(\widetilde{K}), \tilde{\mathbf{u}}_i \right), i \in [m] \right\}$
- $\widetilde{\Omega} = \widetilde{D}_p^{-1/2} \widetilde{U}_p^\top \in \mathbb{R}^{p \times m}$, where $\widetilde{U}_p = (\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_p)$, $\widetilde{D}_p = \mathrm{diag}(\sigma_1(\widetilde{K}), \ldots, \sigma_p(\widetilde{K}))$

1. Training. b. Solving the surrogate problem.

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

2. Inference.

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

Experiments

Sketching size selection strategy

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

Sketching size selection strategy

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

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

Sketching size selection strategy

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

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

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

Sketching size selection strategy

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

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

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

2. Set the optimal m according to the performance of the *perfect h* estimator on the validation set, i.e.

$$h : (x, y) \mapsto \sum_{j=1}^{p} \langle \tilde{e}_j, \psi(y) \rangle_{\mathcal{H}} \, \tilde{e}_j = \sum_{j=1}^{p} \tilde{\psi}(y)_j \, \tilde{e}_j \,. \tag{9}$$

$\Longrightarrow$ allows to cope with the neural net training phase

Synthetic Least Squares Regression

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

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 linear kernel $\implies$ $\mathcal{H} = \mathcal{Y} = \mathbb{R}^{1,000}$

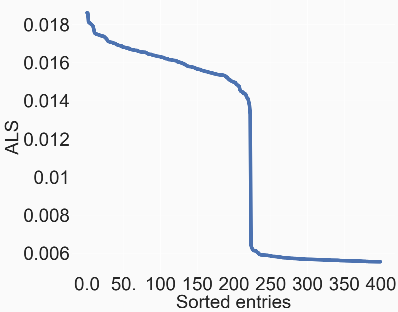
Goal: build this dataset such that the outputs lie in a subspace of $\mathcal{Y}$ of dimension $d = 50 < 1,000$

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. $H_{ij} \sim \mathcal{N}(0,1)$, $x_i \sim \mathcal{N}(0, C)$, where $(\sigma_j(C) = j^{-1/2})_{j=1}^{2,000}$, $\varepsilon_i \sim \mathcal{N}(0, \sigma^2 I_{1,000})$ with $\sigma^2 = 0.01$,

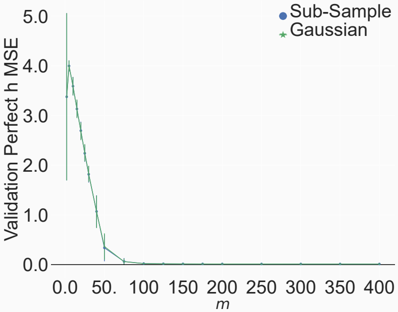
$$y_i = UHx_i + \varepsilon_i\,, \tag{10}$$

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: Sketching Size Selection



(a) Sorted 400 highest ALS.



(b) Validation MSE of *Perfect h* w.r.t. *m*.

Synthetic Least Squares Regression

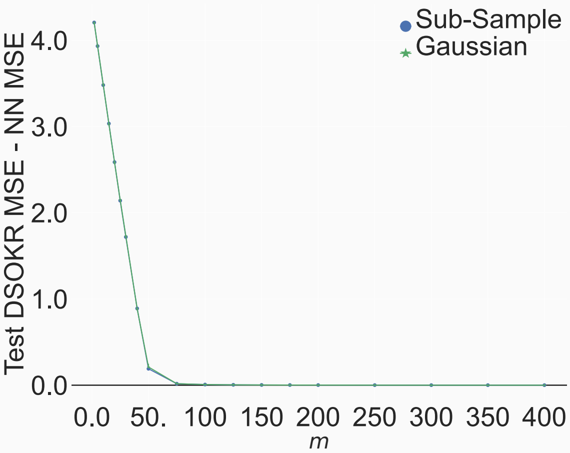


Figure 2: Difference between test MSE of DSOKR and NN w.r.t. m.

Smiles to Molecule

QM9 molecule dataset (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014), containing around 130,000 small organic molecules

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

Smiles to Molecule

QM9 molecule dataset (Ruddigkeit et al., 2012; Ramakrishnan et al., 2014), containing around 130,000 small organic 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)

Sketching: Sub-Sample

Smiles to Molecule: Results

Table 1: Edit distance of different methods on SMI2Mol test set

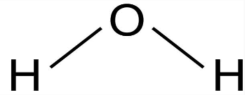
	GED w/o edge feature ↓	GED w/ edge feature ↓
SISOKR	3.330 ± 0.080	4.192 ± 0.109
NNBary-FGW	5.115 ± 0.129	-
Sketched ILE-FGW	2.998 ± 0.253	-
DSOKR	$\mathbf{1.951} \pm \mathbf{0.074}$	$\mathbf{2.960} \pm \mathbf{0.079}$

Text to Molecule

ChEBI-20 dataset (Edwards et al., 2021), containing 33,010 pairs of compounds and descriptions, compounds from PubChem (Kim et al., 2016, 2019) and their descriptions from the Chemical Entities of Biological Interest (ChEBI) database (Hastings et al., 2016). 80% for training, 10% for validation, and 10% for testing

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

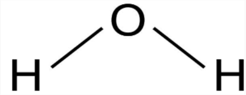
$$\xrightarrow{\quad f \;=\; ? \quad}$$



Text to Molecule

ChEBI-20 dataset (Edwards et al., 2021), containing 33,010 pairs of compounds and descriptions, compounds from PubChem (Kim et al., 2016, 2019) and their descriptions from the Chemical Entities of Biological Interest (ChEBI) database (Hastings et al., 2016). 80% for training, 10% for validation, and 10% for testing

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: Mol2vec (Jaeger et al., 2018)

Sketching: Sub-Sample and Gaussian

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

Take-home messages

- *Deep Sketched Output Kernel Regression* is a family of **deep neural architectures** whose last layer predicts a **data-dependent finite-dimensional representation of the outputs**, that lies in the possibly infinite-dimensional feature space deriving from the kernel-induced loss

Take-home messages

- *Deep Sketched Output Kernel Regression* is a family of **deep neural architectures** whose last layer predicts a **data-dependent finite-dimensional representation of the outputs**, that lies in the possibly infinite-dimensional feature space deriving from the kernel-induced loss
- This last layer is computed **beforehand**, and is the **eigenbasis of the sketched empirical covariance operator** $\implies$ we can use **gradient-based techniques** to learn the weights of the previous layers **for any neural architecture**

Take-home messages

- *Deep Sketched Output Kernel Regression* is a family of **deep neural architectures** whose last layer predicts a **data-dependent finite-dimensional representation of the outputs**, that lies in the possibly infinite-dimensional feature space deriving from the kernel-induced loss
- This last layer is computed **beforehand**, and is the **eigenbasis of the sketched empirical covariance operator** $\implies$ we can use **gradient-based techniques** to learn the weights of the previous layers **for any neural architecture**
- We provide a **strategy to select the sketching size**

Take-home messages

- *Deep Sketched Output Kernel Regression* is a family of **deep neural architectures** whose last layer predicts a **data-dependent finite-dimensional representation of the outputs**, that lies in the possibly infinite-dimensional feature space deriving from the kernel-induced loss
- This last layer is computed **beforehand**, and is the **eigenbasis of the sketched empirical covariance operator** $\implies$ we can use **gradient-based techniques** to learn the weights of the previous layers **for any neural architecture**
- We provide a **strategy to select the sketching size**
- We show that DSOKR performs well on **two text-to-molecule datasets**

Future work

- Excess risk bound for DSOKR
- End-to-end version of DSOKR
- Extension to the auto-encoder architecture

References

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, pages 337–404.

Bach, F. (2013). Sharp analysis of low-rank kernel matrix approximations. In *Proc. of the 26th annual Conference on Learning Theory*, pages 185–209. PMLR.

Belanger, D. and McCallum, A. (2016). Structured prediction energy networks. In *International Conference on Machine Learning*, pages 983–992.

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.

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.

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.

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.

Drineas, P., Mahoney, M. W., and Cristianini, N. (2005). On the nyström method for approximating a gram matrix for improved kernel-based learning. *JMLR*, 6(12).

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.

Gazagnadou, N., Ibrahim, M., and Gower, R. M. (2021). *RidgeSketch*: A fast sketching based solver for large scale ridge regression.

Hastings, J., Owen, G., Dekker, A., Ennis, M., Kale, N., Muthukrishnan, V., Turner, S., Swainston, N., Mendes, P., and Steinbeck, C. (2016). ChEBI in 2016: Improved services and an expanding collection of metabolites. *Nucleic Acids Research*, 44(D1):D1214–D1219.

References v

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.

Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B. A., Thiessen, P. A., Yu, B., Zaslavsky, L., Zhang, J., and Bolton, E. E. (2019). PubChem 2019 update: improved access to chemical data. *Nucleic Acids Research*, 47(D1):D1102–D1109.

Kim, S., Thiessen, P. A., Bolton, E. E., Chen, J., Fu, G., Gindulyte, A., Han, L., He, J., He, S., Shoemaker, B. A., Wang, J., Yu, B., Zhang, J., and Bryant, S. H. (2016). PubChem Substance and Compound databases. *Nucleic Acids Research*, 44(D1):D1202–D1213.

Kpotufe, S. and Sriperumbudur, B. K. (2020). Gaussian sketching yields a J-L lemma in RKHS. In Chiappa, S. and Calandra, R., editors, *AISTATS 2020*, volume 108 of *Proceedings of Machine Learning Research*, pages 3928–3937. PMLR.

Lacotte, J. and Pilanci, M. (2020). Adaptive and oblivious randomized subspace methods for high-dimensional optimization: Sharp analysis and lower bounds. *arXiv preprint arXiv:2012.07054*.

Lacotte, J., Pilanci, M., and Pavone, M. (2019). High-dimensional optimization in adaptive random subspaces. In *Proc. of the 33rd International Conference on Neural Information Processing Systems*, pages 10847–10857.

Lafferty, J. D., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*.

LeCun, Y., Chopra, S., Ranzato, M., and Huang, F.-J. (2007). Energy-based models in document recognition and computer vision. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 1, pages 337–341. IEEE.

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.

Mahoney, M. W. et al. (2011). Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning*, 3(2):123–224.

Meanti, G., Carratino, L., Rosasco, L., and Rudi, A. (2020). Kernel methods through the roof: Handling billions of points efficiently. *Advances in Neural Information Processing Systems (NeurIPS)*, 33.

Nikolentzos, G., Meladianos, P., Limnios, S., and Vazirgiannis, M. (2018). A Degeneracy Framework for Graph Similarity. In *IJCAI*.

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.

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).

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

Rudi, A., Carratino, L., and Rosasco, L. (2017). Falkon: an optimal large scale kernel method. In *Proceedings of the 31st International Conference on Advances on Neural Information Processing Systems (NeurIPS)*, pages 3891–3901.

Rudi, A. and Rosasco, L. (2017). Generalization properties of learning with random features. In *Advances on Neural Information Processing Systems (NeurIPS)*, pages 3215–3225.

Sriperumbudur, B. K. and Szabó, Z. (2015). Optimal rates for random fourier features. In *NIPS*.

Taskar, B., Guestrin, C., and Koller, D. (2003). Max-margin markov networks. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems*, volume 16. MIT Press.

Tsochantaridis, I., Hofmann, T., Joachims, T., and Altun, Y. (2004). Support vector machine learning for interdependent and structured output spaces. *Machine Learning*.

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 *NeurIPS*.

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.

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.

Woodruff, D. P. (2014). Sketching as a tool for numerical linear algebra. *Found. Trends Theor. Comput. Sci.*, 10(1-2):1–157.

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.

Reminder: positive definite kernels and Reproducing Kernel Hilbert Space

Positive definite kernel: $k_{\mathcal{Z}} : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ such that

- for all $(z, z') \in \mathcal{Z}^2$, $k_{\mathcal{Z}}(z, z') = k_{\mathcal{Z}}(z', z)^\top$
- for all $n \in \mathbb{N}$ and any $(z_i, \alpha_i)_{i=1}^n \in (\mathcal{Z} \times \mathbb{R})^n$, $\sum_{i,j=1}^n \alpha_i \alpha_j k_{\mathcal{Z}}\left(z_i, z_j\right) \geqslant 0$

RKHS (Aronszajn, 1950): Hilbert space $\mathcal{H}_{\mathcal{Z}}$ of functions $f : \mathcal{Z} \to \mathbb{R}$ s. t. for all $f \in \mathcal{H}_{\mathcal{Z}}$ and $z \in \mathcal{Z}$

1. $z' \mapsto k_{\mathcal{Z}}(z, z') \in \mathcal{H}_{\mathcal{Z}}$,
2. $\langle f, k_{\mathcal{Z}}(\cdot, z) \rangle_{\mathcal{H}_{\mathcal{Z}}} = f(z)$ **(reproducing property)**.

Fisher consistency and excess risk bound

Lemma 1 and Theorem 3 from Ciliberto et al. (2020). Let $\mathcal{Y}$ be compact, $k_{\mathcal{Y}} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be a p.d. kernel and $\psi_{\mathcal{Y}} : y \mapsto k_{\mathcal{Y}}(\cdot, y)$ s.t. $\|\psi_{\mathcal{Y}}(y)\|_{\mathcal{H}_{\mathcal{Y}}} = 1, \forall y \in \mathcal{Y}$, and

$$f^* = \operatorname*{arg\,inf}_{f:\mathcal{X}\to\mathcal{Y}} \mathcal{R}(f) = \operatorname*{arg\,inf}_{f:\mathcal{X}\to\mathcal{Y}} \mathbb{E}_{(x,y)\sim\rho}[\|\psi_{\mathcal{Y}}(f(x)) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}}] .$$

Then,

$$f^*(x) = \operatorname*{arg\,min}_{y\in\mathcal{Y}} \|h^*(x) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}} = d \circ h^*(x) , \quad h^*(x) = \mathbb{E}_y[\psi_{\mathcal{Y}}(y)|x] ,$$

almost surely with respect to $\rho_{\mathcal{X}}$.

Fisher consistency and excess risk bound

Lemma 1 and Theorem 3 from Ciliberto et al. (2020). Let $\mathcal{Y}$ be compact, $k_{\mathcal{Y}} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be a p.d. kernel and $\psi_{\mathcal{Y}} : y \mapsto k_{\mathcal{Y}}(\cdot, y)$ s.t. $\|\psi_{\mathcal{Y}}(y)\|_{\mathcal{H}_{\mathcal{Y}}} = 1, \forall y \in \mathcal{Y}$, and

$$f^* = \underset{f:\mathcal{X}\to\mathcal{Y}}{\arg\inf}\ \mathcal{R}(f) = \underset{f:\mathcal{X}\to\mathcal{Y}}{\arg\inf}\ \mathbb{E}_{(x,y)\sim\rho}[\|\psi_{\mathcal{Y}}(f(x)) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}}]\,.$$

Then,

$$f^*(x) = \underset{y\in\mathcal{Y}}{\arg\min}\ \|h^*(x) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}} = d \circ h^*(x)\,, \quad h^*(x) = \mathbb{E}_y[\psi_{\mathcal{Y}}(y)|x]\,,$$

almost surely with respect to $\rho_{\mathcal{X}}$.

Moreover, let $h : \mathcal{X} \to \mathcal{H}_{\mathcal{Y}}$ be measurable and $f : \mathcal{X} \to \mathcal{Y}$ such that, for any $x \in \mathcal{X}$,

$$f(x) = \underset{y\in\mathcal{Y}}{\arg\min}\ \|h(x) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}} = d \circ h(x)\,.$$

Then,

$$\mathcal{R}(f) - \mathcal{R}(f^*) \leq 12\sqrt{\mathcal{E}(h) - \mathcal{E}(h^*)}\,,$$

where $\mathcal{E}(h) = \mathbb{E}_{(x,y)\sim\rho}[\|h(x) - \psi_{\mathcal{Y}}(y)\|^2_{\mathcal{H}_{\mathcal{Y}}}]$.

Background: Scalability to large datasets

1) Random Fourier Features (Rahimi and Recht, 2007; Rudi and Rosasco, 2017; Sriperumbudur and Szabó, 2015; Brault et al., 2016; Li et al., 2021)

Background: Scalability to large datasets

1) **Random Fourier Features** (Rahimi and Recht, 2007; Rudi and Rosasco, 2017; Sriperumbudur and Szabó, 2015; Brault et al., 2016; Li et al., 2021)

2) **Sketching (Mahoney et al., 2011; Woodruff, 2014):** dimension reduction approach based on random linear projections

- Nyström approximation ($\iff$ sub-sampling sketch) (Williams and Seeger, 2001; Drineas et al., 2005; Bach, 2013; Rudi et al., 2017; Meanti et al., 2020)
- Gaussian, Randomized Orthogonal Systems, sparse sketches etc. (Yang et al., 2017; Lacotte et al., 2019; Kpotufe and Sriperumbudur, 2020; Lacotte and Pilanci, 2020; Chen and Yang, 2021; Gazagnadou et al., 2021)

Example: Sketching for scalar Kernel Ridge Regression ($\mathcal{Y} = \mathbb{R}$)

Representer theorem: $\hat{f} = \sum_{i=1}^{n} \hat{\alpha}_i k_x(\cdot, x_i)$, where

$$\hat{\alpha} = (\hat{\alpha}_1, \ldots, \hat{\alpha}_n)^\top = \underset{\alpha \in \mathbb{R}^n}{\arg\min}\ \alpha^\top \left(K_X^2 + n\lambda K_X\right) \alpha - 2Y^\top K_X \alpha$$

$$= \underbrace{(K_X + n\lambda I_n)}_{n \times n}{}^{-1} Y$$

Example: Sketching for scalar Kernel Ridge Regression ($\mathcal{Y} = \mathbb{R}$)

Representer theorem: $\hat{f} = \sum_{i=1}^{n} \hat{\alpha}_i k_x(\cdot, x_i)$, where

$$\hat{\alpha} = (\hat{\alpha}_1, \ldots, \hat{\alpha}_n)^\top = \underset{\alpha \in \mathbb{R}^n}{\arg\min}\ \alpha^\top \left(K_X^2 + n\lambda K_X\right) \alpha - 2Y^\top K_X \alpha$$
$$= \underbrace{(K_X + n\lambda I_n)}_{n \times n}{}^{-1} Y$$

Let $m \ll n$, $R \in \mathbb{R}^{m \times n}$ be a random matrix: $\alpha \leftarrow R^\top \gamma$

Example: Sketching for scalar Kernel Ridge Regression ($\mathcal{Y} = \mathbb{R}$)

Representer theorem: $\hat{f} = \sum_{i=1}^{n} \hat{\alpha}_i k_x(\cdot, x_i)$, where

$$\hat{\alpha} = (\hat{\alpha}_1, \dots, \hat{\alpha}_n)^\top = \underset{\alpha \in \mathbb{R}^n}{\arg\min}\ \alpha^\top \left(K_X^2 + n\lambda K_X\right) \alpha - 2Y^\top K_X \alpha$$
$$= \underbrace{(K_X + n\lambda I_n)}_{n \times n}{}^{-1} Y$$

Let $m \ll n$, $R \in \mathbb{R}^{m \times n}$ be a random matrix: $\alpha \leftarrow R^\top \gamma$

$\hat{f} \leftarrow \tilde{f} = \sum_{i=1}^{n} [R^\top \tilde{\gamma}]_i k_x(\cdot, x_i)$, where

$$\tilde{\gamma} = (\tilde{\gamma}_1, \dots, \tilde{\gamma}_m)^\top = \underset{\gamma \in \mathbb{R}^m}{\arg\min}\ \gamma^\top \left(RK_X^2R^\top + n\lambda RK_XR^\top\right) \gamma - 2Y^\top K_X R^\top \gamma$$
$$= \underbrace{(RK_X^2R^\top + n\lambda RK_XR^\top)}_{m \times m}{}^{\dagger} RK_X Y$$

Smiles to Molecule: some nice figures

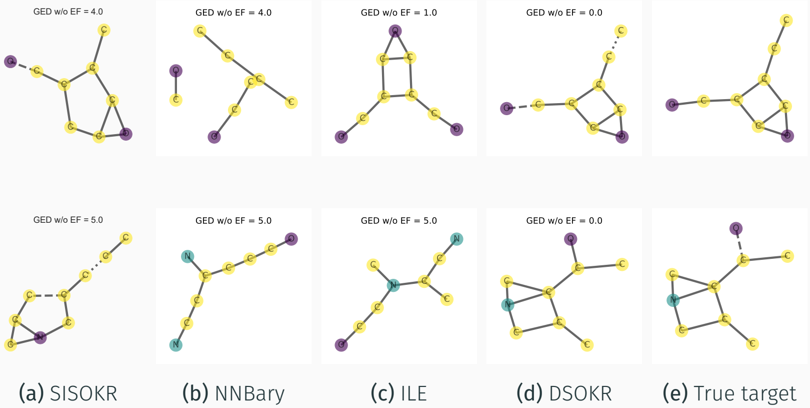


Figure 3: Predicted molecules on the SMI2Mol dataset.