# Orthogonal Contrastive Learning for Multi-Representation fMRI Analysis

**Supplementary Material**

**Tony Muhammad Yousefnezhad**[1,2,*]
[1]Learning By Machine
[2]Information Management, National Bank of Canada
Edmonton AB Canada
tony@learningbymachine.com

## A Proofs

**Lemma 1** *Let each neural responses matrix admit a reduced QR factorization* $\mathbf{X}_s = \mathbf{Q}_s\,\mathbf{R}_s$ *with* $\mathbf{Q}_s^\top \mathbf{Q}_s = \mathbf{I}$. *Define* $\boldsymbol{\phi}_s = \operatorname{vec}(\mathbf{R}_s) = \operatorname{vec}\big(\mathbf{Q}_s^\top\,\mathbf{X}_s\big)$. *If*

$$\big\|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2\big\| < \big\|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_3\big\| \to \Pr\big[\ell_1 = \ell_2\big] \ > \ \Pr\big[\ell_1 = \ell_3\big].$$

*Proof.* We use the LSH scheme $\ell_s = \big\lfloor(\mathbf{a}^\top \phi_s + b)/w\big\rfloor$ with $\mathbf{a}$ is drawn from a $p$-stable distribution and $b \sim U[0, w]$. In this work, we employ a 2-stable (Gaussian) distribution for our LSH projection vectors. Define the projected difference $\Delta_{1s} = \mathbf{a}^\top\big(\boldsymbol{\phi}_1 - \boldsymbol{\phi}_s\big)$, which by the 2-stable property of Gaussians satisfies $\Delta_{1s} \sim \mathcal{N}\big(0, \|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_s\|^2\big)$. The collision event $\ell_1 = \ell_s$ is equivalent to $\lfloor(\mathbf{a}^\top\boldsymbol{\phi}_1 + b)/w\rfloor = \lfloor(\mathbf{a}^\top\boldsymbol{\phi}_s + b)/w\rfloor$, which occurs exactly when $|\Delta_{1s}| < w - r$ for some fractional offset $r$. Since $r$ is uniform on $[0, w]$,

$$\Pr\big[\ell_1 = \ell_s\big] = \frac{1}{w}\int_0^w \Pr\big(|\Delta_{1s}| < w - r\big)\,dr.$$

For $\Delta \sim \mathcal{N}(0, \sigma^2)$, $\Pr(|\Delta| < t) = \int_{-t}^{t} \frac{1}{\sqrt{2\pi}\,\sigma} e^{-u^2/(2\sigma^2)} du$, which is strictly decreasing in $\sigma$ for any fixed $t > 0$ [1]. Therefore, if $\|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2\| < \|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_3\|$, then the smaller variance $\|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2\|^2$ yields

$$\Pr\big[\ell_1 = \ell_2\big] > \ \Pr\big[\ell_1 = \ell_3\big].$$

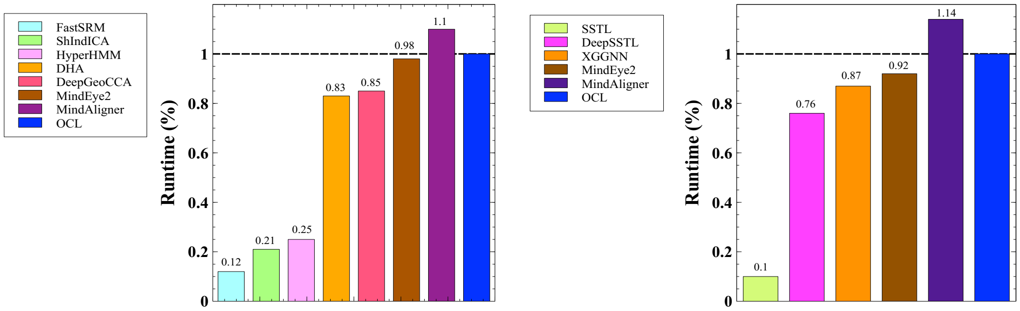
**Remark.** In OCL, we truncate to $d$ components with $d \le \min_{s=1,\dots,S} \operatorname{rank}(\mathbf{X}_s)$, so that $\mathbf{Q}_s^{\text{truncate}} \in \mathbb{R}^{T_s \times d}$ and $\mathbf{R}_s \in \mathbb{R}^{d \times V}$. We then set $\boldsymbol{\phi}_s = \operatorname{vec}(\mathbf{R}_s)$. Since the full-rank factorization obeys $\mathbf{R}_s^{\text{full}} = \begin{bmatrix}\mathbf{R}_s \\ *_s\end{bmatrix}$, one has

$$\|\boldsymbol{\phi}_1^{\text{full}} - \boldsymbol{\phi}_2^{\text{full}}\|^2 = \|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2\|^2 + \|\, *_1 - *_2 \,\|^2 \ \ge\ \|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2\|^2,$$

and similarly for views 1 and 3. Therefore if $\|\boldsymbol{\phi}_1^{\text{full}} - \boldsymbol{\phi}_2^{\text{full}}\| < \|\boldsymbol{\phi}_1^{\text{full}} - \boldsymbol{\phi}_3^{\text{full}}\|$, then also $\|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_2\| < \|\boldsymbol{\phi}_1 - \boldsymbol{\phi}_3\|$. The above 2-stable argument then applies unchanged to the truncated signatures $\boldsymbol{\phi}_s$, preserving the collision-probability ordering. □

[*] This research study is conducted independently and is not connected to the author's role at the *National Bank of Canada.*

39th Conference on Neural Information Processing Systems (NeurIPS 2025).



(a) CMU (b) E ⇌ F

Figure S1: Runtime Analysis

**Lemma 2** *Let each synthetic view be generated by an orthonormal rotation of the base data,* $\mathbf{X}_s = \mathbf{M}\,\mathbf{U}_s, \quad \mathbf{U}_s^\top \mathbf{U}_s = \mathbf{I}$. *Write the corresponding QR factor* $\mathbf{R}_s$ *and its flattened signature* $\boldsymbol{\phi}_s = \mathrm{vec}(\mathbf{R}_s)$. *Then for any three views* $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ *generated from* $\mathbf{M}$*,*

$$\Pr\big[\ell_1 = \ell_2\big] \;=\; \Pr\big[\ell_1 = \ell_3\big].$$

*In other words, the collision probability of the LSH is identical across all random rotations generated from* $\mathbf{M}$*.*

*Proof.* Since each view is generated by an orthonormal rotation of the same base matrix,

$$\mathbf{X}_s = \mathbf{M}\,\mathbf{U}_s, \quad \mathbf{U}_s^\top \mathbf{U}_s = \mathbf{I},$$

the reduced QR decomposition gives

$$\mathbf{X}_s = \mathbf{Q}_s\,\mathbf{R}_s, \qquad \mathbf{Q}_s^\top \mathbf{Q}_s = \mathbf{I}.$$

Hence

$$\mathbf{R}_s = \mathbf{Q}_s^\top\,\mathbf{X}_s = \mathbf{Q}_s^\top\,\mathbf{M}\,\mathbf{U}_s.$$

so that

$$\boldsymbol{\phi}_s = \mathrm{vec}(\mathbf{R}_s) = (\mathbf{U}_s^\top \otimes \mathbf{Q}_s^\top)\,\mathrm{vec}(\mathbf{M}).$$

Denote $\mathbf{A}_s = \mathbf{U}_s^\top \otimes \mathbf{Q}_s^\top$ (which is orthonormal since both $\mathbf{U}_s$ and $\mathbf{Q}_s$ are), and $\mathbf{v} = \mathrm{vec}(\mathbf{M})$. By considering the Gaussian properties of $\mathbf{a}$ [2, 3], and hash function

$$\ell = \left\lfloor \frac{\mathbf{a}^\top \mathbf{v} + b}{w} \right\rfloor.$$

For any view $s$,

$$\mathbf{a}^\top \boldsymbol{\phi}_s = \mathbf{a}^\top \mathbf{A}_s\,\mathbf{v} = (\mathbf{A}_s^\top \mathbf{a})^\top \mathbf{v} \;\overset{d}{=}\; \mathbf{a}^\top \mathbf{v},$$

by the rotational invariance of multivariate Gaussians. Hence the distribution of $\ell_s$ depends only on $\mathbf{v} = \mathrm{vec}(\mathbf{M})$, not on $\mathbf{U}_s$. It follows that for any three views $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ generated from $\mathbf{M}$,

$$\Pr\big[\ell_1 = \ell_2\big] \;=\; \Pr\big[\ell_1 = \ell_3\big].$$

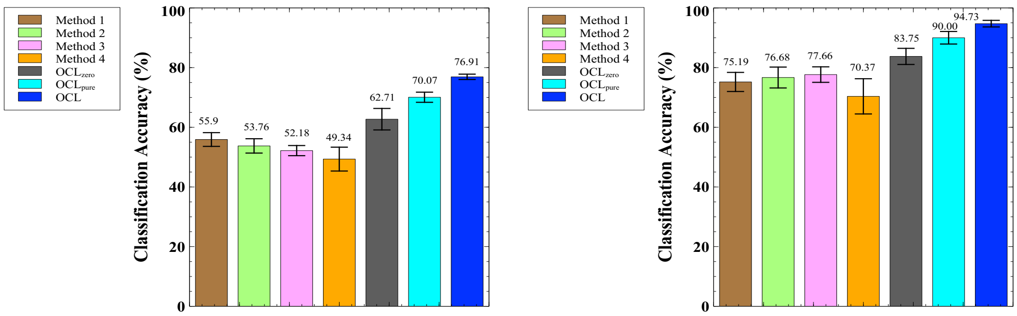
□

## B Evaluating Feature Representation Quality of Pretrained $\mathrm{OCL}_{\mathrm{zero}}$

This section evaluates the feature-dependency before and after alignment in the pretrained network $\mathrm{OCL}_{\mathrm{zero}}$. OCL is designed to map multi-subject or multi-site fMRI responses into a latent space that pulls together embeddings from the same stimulus and pushes apart those from different stimuli, thereby improving downstream classification. While Section 4 measures representational quality via classification accuracy, here we directly quantify how $\mathrm{OCL}_{\mathrm{zero}}$ transforms synthetic data by computing (i) the Pearson correlation $\rho$ for linear dependence and (ii) the mutual information $I$ for nonlinear dependence.

On two million raw synthetic samples, we observe

$$\rho_{\mathrm{raw}}^{\mathrm{within}} = 0.763 \pm 0.162, \quad \rho_{\mathrm{raw}}^{\mathrm{between}} = 0.204 \pm 0.134,$$



(a) CMU (b) DS232

Figure S2: OCL Component Ablation Study

which after alignment become

$$\rho_{\text{aligned}}^{\text{within}} = 0.915 \pm 0.042, \quad \rho_{\text{aligned}}^{\text{between}} = -0.075 \pm 0.050.$$

Similarly, the mean mutual information on raw data is

$$I_{\text{raw}}^{\text{within}} = 0.444 \pm 0.021, \quad I_{\text{raw}}^{\text{between}} = 0.022 \pm 0.011,$$

improving to

$$I_{\text{aligned}}^{\text{within}} = 0.900 \pm 0.011, \quad I_{\text{aligned}}^{\text{between}} = 0.003 \pm 0.0001.$$

These results demonstrate that $\text{OCL}_{\text{zero}}$ substantially increases both linear and nonlinear dependencies among same-class embeddings, while effectively decorrelating different-class features.

## C Runtime Analysis

We evaluate the runtime of each alignment method on (i) the CMU dataset for multi-subject fMRI analysis and (ii) the cross-site pair E ⇌ F for multi-site evaluation. All timings were collected on a PC with the hardware and software configuration detailed in Footnote [2]. In Figure S1, we plot each method's total fine-tuning and inference time normalized by the runtime of OCL. To ensure a fair comparison, we exclude the pretraining phase for all self-supervised approaches (MindEye2, MindAligner, and OCL), measuring only their downstream fine-tuning and evaluation costs. Relative to linear methods—*i.e.*, FastSRM, ShIndICA, HyperHMM, and SSTL — deep neural network–based approaches incur higher runtimes due to their more complex model and optimization, but they consistently achieve superior classification accuracies. We would like to note that during inference on typical fMRI datasets, OCL requires about 14 GB of VRAM on a single GPU — suggesting that its computational resource demands are moderate and attainable by many neuroimaging labs equipped with modern hardware.

## D OCL Component Ablation Study

To quantify the contribution of each OCL component, we perform an ablation study on the two simple cognitive-task datasets in Table 1, *viz.*, CMU and DS232. We compare four simplified variants against three baselines: (i) the pretrained $\text{OCL}_{\text{zero}}$ (ii) the OCL model fine-tuned using the pretrained $\text{OCL}_{\text{zero}}$, and (iii) $\text{OCL}_{\text{pure}}$ that is the complete QR+LSH+PositionalEncoding+Transformer pipeline but is trained from random initialization (no pretraining). Our ablation architectures include:

- **Method 1:** Skip both QR and LSH. The raw input $\mathbf{X}_s$ is passed directly through positional encoding and then the Transformer encoder.
- **Method 2:** Skip LSH. Apply thin QR decomposition to obtain $(\mathbf{Q}_s, \mathbf{R}_s)$, then concatenate $\mathbf{Q}_s$ and $\mathbf{R}_s$, followed by positional encoding and the Transformer.
- **Method 3:** Skip Positional Encoding. Compute QR and then LSH to get signature $\phi_s$; concatenate $\mathbf{Q}_s$ with the MLP-embedded $\mathbf{s}_s$, and feed directly into the Transformer.

[2] OS: Fedora 42, Python: 3.11.9, PyTorch: 2.6, CUDA: 12.6; CPU: AMD EPYC $7551P$ (64 cores), RAM: 256G GPU: 2×NVIDIA 4060Ti 16G.

- **Method 4:** Skip Encoder Transformer. Use the full QR+LSH+PositionalEncoding pipeline, but replace the Transformer encoder with a bidirectional LSTM of comparable capacity.

Figure S2 reports classification accuracy for each variant. We observe that: (1) all ablated methods (Methods 1–4) perform worse than even the pretrained $\mathrm{OCL}_{\mathrm{zero}}$ model. (2) $\mathrm{OCL}_{\mathrm{zero}}$ pretraining consistently boosts downstream fine-tuning accuracy over random initialization. (3) the full QR+LSH+PositionalEncoding+Transformer pipeline is essential: removing any component incurs a significant drop in performance. Note that we observe the same performance trends across all other datasets evaluated in our experiments.

We further highlight that the theoretical analysis presented in Lemma 1 (please see Section A) demonstrates that the LSH collision probability decreases smoothly as the Euclidean distance between orthogonal codes increases, offering intrinsic resilience to noise and minor perturbations. To confirm this property, we performed a series of empirical experiments by injecting Gaussian noise into the synthetic pretraining data at SNR levels of $10$ dB, $5$ dB, and $0$ dB. In all cases, the classification accuracy declined by less than $1\%$, confirming the strong robustness of the method.

## E Notations and Algorithms

Table S1 summarizes the notations used throughout this paper. Algorithms S1 and S2 present the detailed pseudocode for the proposed orthogonal contrastive learning applied to multi-subject and multi-site fMRI analyses, respectively.

Table S1: Notations

| **Symbol** | **Description** |
|---|---|
| $S$ | Number of subjects in the training set. |
| $S_b$ | Number of subjects for site $b$. |
| $T_s$ | Number of time points for subject $s, s = 1, \ldots, S$. |
| $T \ \geq \ \max_{s=1,\ldots,S} T_s$ | Maximum time-series length, *i.e.*, content-window size |
| $V$ | Number of voxels in the region of interest. |
| $N$ | Number of OCL layers. |
| $r_s$ | Rank of $\mathbf{X}_s$. |
| $d$ | Embedding dimension (number of features). |
| $\bar{d}$ | Number of nonzeros in the upper-triangular $\mathbf{R}_s$. |
| $\mathbb{R}$ | The set of real numbers. |
| $p$ | Stability exponent for the $p$-stable distribution ($p = 2$ in this paper). |
| $w$ | Quantization granularity (LSH bin width). |
| $b \sim U[0, w]$ | Random offset for LSH. |
| $\tau$ | Temperature hyperparameter in contrastive loss $\mathcal{L}_{\text{OCL}}$. |
| $\mu$ | Margin hyperparameter in between-class loss $\mathcal{L}_{\text{OCL}}$. |
| $\lambda$ | Weight for the between-class term in loss $\mathcal{L}_{\text{OCL}}$. |
| $\eta$ | Learning rate. |
| $\psi$ | Total number of iterations. |
| $k$ | Number of classes in synthetic data. |
| $B$ | Number of distinct training sites in multi-site OCL. |
| $f_n$ | Transformation implemented by layer $n$. |
| $\theta$ | Learnable parameters of the OCL online network. |
| $\tilde{\theta}$ | Parameters of the target network. |
| $\tilde{\theta}_b$ | Target-encoder parameters for site $b$. |
| $\tilde{\theta}_{\text{sites}} = \frac{1}{B}\sum_b \tilde{\theta}_b$ | Aggregated multi-site encoder parameters. |
| $\mathbf{I}$ | Identity matrix. |
| $\mathbf{X}_s \in \mathbb{R}^{T_s \times V}$ | Preprocessed neural response matrix for subject $s$. |
| $\mathbf{X}_s^{(b)}$ | Preprocessed neural response matrix for subject $s$ in site $b$. |
| $\mathbf{H}_{0,s} = \mathbf{X}_s$ | Input to the first layer ($f_1$) for subject $s$. |
| $\mathbf{H}_{n,s} = f_n(\mathbf{H}_{n-1,s})$ | Output of layer $n$ for subject $s$. |
| $\mathbf{Z}_s = \mathbf{H}_{N,s} \in \mathbb{R}^{T_s \times d}$ | Final online-encoder representation for subject $s$. |
| $\mathbf{Q}_s^{\text{full}} \in \mathbb{R}^{T_s \times r_s}$ | Orthonormal factor from full QR of $\mathbf{X}_s$. |
| $\mathbf{Q}_s^{\text{truncate}} \in \mathbb{R}^{T_s \times d}$ | First $d$ columns of $\mathbf{Q}_s^{\text{full}}$. |
| $\mathbf{Q}_s \in \mathbb{R}^{T \times d}$ | Zero-padded $\mathbf{Q}_s^{\text{truncate}}$ to length $T$. |
| $\mathbf{R}_s^{\text{full}} \in \mathbb{R}^{r_s \times V}$ | Upper-triangular factor from full QR of $\mathbf{X}_s$. |
| $\mathbf{R}_s \in \mathbb{R}^{d \times V}$ | First $d$ rows of $\mathbf{R}_s^{\text{full}}$. |
| $\mathbf{m}_s \in \{0, 1\}^T$ | Binary padding mask (1 =real time point, 0 =pad). |
| $\boldsymbol{\phi}_s \in \mathbb{R}^{\bar{d}}$ | vec($\mathbf{R}_s$), vectorized nonzero entries. |
| $\mathbf{a} \in \mathbb{R}^{\bar{d}}$ | Random vector from a $p$-stable distribution ($p = 2$ in this paper). |
| $\ell_s = \text{lsh}(\phi_s) = \lfloor (\langle \mathbf{a}, \phi_s \rangle + b)/w \rfloor$ | LSH hash index for subject $s$. |
| $\mathbf{s}_s \in \mathbb{R}^d$ | Subject-specific signature embedding from $\ell_s$. |
| $\mathbf{C}_s \in \mathbb{R}^{T \times 2d}$ | Concatenation of each row of $\mathbf{Q}_s$ and $\mathbf{s}_s$. |
| $\mathbf{E} \in \mathbb{R}^{T \times 2d}$ | Sinusoidal positional-encoding matrix. |
| $\mathbf{P}_s = \mathbf{C}_s + \mathbf{E}$ | Positional-encoded features. |
| $\mathbf{N}_s = \text{Norm}(\mathbf{P}_s)$ | Layer-wise normalization of $\mathbf{P}_s$. |
| $\mathbf{y}_s = [y_{s,1}, \ldots, y_{s,T_s}]^\top$ | Stimulus-label sequence for subject $s$. |
| $\mathbf{y}_s^{(b)}$ | Stimulus-label sequence for subject $s$ in site $b$. |
| $\mathbf{M} \in \mathbb{R}^{T \times V}$ | Base matrix in synthetic pretraining. |
| $\mathbf{U}_s \in \mathbb{R}^{V \times V}$ | Random orthonormal rotation for view $s$ in synthetic pretraining. |
| $\mathcal{L}_{\text{OCL}}(\mathbf{Z}_s, \mathbf{y}_s)$ | OCL contrastive loss for subject $s$. |
| $\Pr[x]$ | Probability of $x$ |
| $\pi$ | A Classification Model ($\nu$-SVM in our paper) |

**Algorithm S1** Orthogonal contrastive learning for multi-subject fMRI analysis

```
Require: Training data $\{(\mathbf{X}_s, \mathbf{y}_s)\}_{s=1}^{S}$, Testing data $\{\tilde{\mathbf{X}}_s\}_{s=1}^{\tilde{S}}$, Number of network layers $N$, Iterations $\psi$, Learning rate $\eta$, Positional encoding $\mathbf{E}$, other internal parameters $(T, \tau, \mu, \ldots)$
# Training Phase
Initialize online parameters $\theta$, target parameters $\tilde{\theta} \leftarrow \theta$ (randomly or from pretrained model)
for $i = 1$ to $\psi$ do
    for $s = 1$ to $S$ do
        $\mathbf{H}_{0,s} \leftarrow \mathbf{X}_s$
        for $n = 1$ to $N$ do
            $(\mathbf{Q}_s^{(n)}, \mathbf{R}_s^{(n)}, \mathbf{m}_s^{(n)}) \leftarrow$ QR decomposition and Truncate on $\mathbf{H}_{n-1,s}$
            $\phi_s^{(n)} \leftarrow \mathrm{vec}(\mathbf{R}_s^{(n)})$
            $\ell_s^{(n)} \leftarrow \mathrm{lsh}(\phi_s^{(n)})$
            $\mathbf{s}_s^{(n)} \leftarrow \mathrm{MLP}(\ell_s^{(n)})$
            $\mathbf{C}_s^{(n)} \leftarrow \mathrm{concat}(\mathbf{Q}_s^{(n)}, \mathbf{s}_s^{(n)})$
            $\mathbf{P}_s^{(n)} \leftarrow \mathbf{C}_s^{(n)} + \mathbf{E}$
            $\mathbf{N}_s^{(n)} \leftarrow \mathrm{Norm}(\mathbf{P}_s^{(n)})$
            $\mathbf{H}_{n,s} \leftarrow \mathrm{TransformerEncoder}(\mathbf{N}_s^{(n)}, \mathbf{m}_s^{(n)})$
        end for
        $\mathbf{Z}_s \leftarrow \mathbf{H}_{N,s}$
        Calculating contrastive loss $\mathcal{L}_{OCL}(\mathbf{Z}_s, \mathbf{y}_s)$
        $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{OCL}$
    end for
    $\tilde{\theta} = \frac{1}{\psi}\theta + \left(1 - \frac{1}{\psi}\right)\tilde{\theta}$
end for
Train classifier $\pi$ on $\{(\mathbf{Z}_s, \mathbf{y}_s)\}_{s=1}^{S}$
# Testing Phase
for $s = 1$ to $\tilde{S}$ do
    Compute $\tilde{\mathbf{Z}}_s$ via same OCL layers using parameters $\tilde{\theta}$
    Predicting $\tilde{\mathbf{y}}_s^* \leftarrow \pi(\tilde{\mathbf{Z}}_s)$
end for
return $\{\tilde{\mathbf{y}}_s^*\}_{s=1}^{\tilde{S}}$
```

**Algorithm S2** Orthogonal contrastive learning for multi-site fMRI analysis

```
Require: Training sites $\{(\mathbf{X}^{(b)}, \mathbf{y}^{(b)})\}_{b=1}^{B}$, Testing sites $\{\tilde{\mathbf{X}}^{(b)}\}_{b=1}^{\tilde{B}}$
# Training Phase
for $b = 1$ to $B$ do
    Learning the target network with $\tilde{\theta}_b$ parameters using Algorithm S1
end for
$\tilde{\theta}_{\text{sites}} \leftarrow \frac{1}{B}\sum_{b=1}^{B}\tilde{\theta}_b$
$\{\mathbf{Z}^{(b)}\}_{b=1}^{B} \leftarrow \{\mathbf{X}^{(b)}\}_{b=1}^{B}$ mappings by using the target network with $\tilde{\theta}_{\text{sites}}$ parameters
Train global classifier $\pi$ on features $\{\mathbf{Z}^{(b)}\}_{b=1}^{B}$
# Testing Phase
for $b = 1$ to $\tilde{B}$ do
    $\tilde{\mathbf{Z}}^{(b)} \leftarrow \tilde{\mathbf{X}}^{(b)}$ mappings by using the target network with $\tilde{\theta}_{\text{sites}}$ parameters
    $\tilde{\mathbf{y}}^{(b)} \leftarrow \pi(\tilde{\mathbf{Z}}^{(b)})$
end for
return Predictions $\{\tilde{\mathbf{y}}^{(b)}\}_{b=1}^{\tilde{B}}$
```

## References

[1] Martin Aumüller, Tobias Christiani, Rasmus Pagh, and Francesco Silvestri. Distance-sensitive hashing. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 89–104, 2018.

[2] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of Symposium on Computational Geometry (SoCG)*, pages 253–262, 2004.

[3] Hongkai Nguyen. Lecture notes: Sketching and $p$-stable distributions. CS261, Stanford University, 2019. Definition 2.1.