

## A EXTENSION TO ARBITRARY DISTRIBUTIONS

**Overall notations.** Let  $X \in \mathcal{R}(\mathbb{R}^d)$  denote a random vector on  $\mathbb{R}^d$  with  $\alpha_X \in \mathcal{P}(\mathbb{R}^d)$  its law (a positive Radon measure with unit mass). By definition, its expectation denoted  $\mathbb{E}(X)$  reads  $\mathbb{E}(X) = \int_{\mathbb{R}^d} x d\alpha_X(x) \in \mathbb{R}^d$ , and for any continuous function  $f : \mathbb{R}^d \rightarrow \mathbb{R}^r$ ,  $\mathbb{E}(f(X)) = \int_{\mathbb{R}^d} f(x) d\alpha_X(x)$ . In the following, two random vectors  $X$  and  $X'$  with same law  $\alpha_X$  are considered indistinguishable, noted  $X' \sim X$ . Letting  $f : \mathbb{R}^d \mapsto \mathbb{R}^r$  denote a function on  $\mathbb{R}^d$ , the push-forward operator by  $f$ , noted  $f_\# : \mathcal{P}(\mathbb{R}^d) \mapsto \mathcal{P}(\mathbb{R}^r)$  is defined as follows, for any  $g$  continuous function from  $\mathbb{R}^d$  to  $\mathbb{R}^r$  ( $g$  in  $\mathcal{C}(\mathbb{R}^d; \mathbb{R}^r)$ ):

$$\forall g \in \mathcal{C}(\mathbb{R}^d; \mathbb{R}^r) \quad \int_{\mathbb{R}^r} g d(f_\# \alpha) \stackrel{\text{def.}}{=} \int_{\mathbb{R}^d} g(f(x)) d\alpha(x)$$

Letting  $\{x_i\}$  be a set of points in  $\mathbb{R}^d$  with  $w_i \geq 0$  such that  $\sum_i w_i = 1$ , the discrete measure  $\alpha_X = \sum_i w_i \delta_{x_i}$  is the sum of the Dirac measures  $\delta_{x_i}$  weighted by  $w_i$ .

**Invariances.** In this paper, we consider functions on probability measures that are *invariant with respect to permutations of coordinates*. Therefore, denoting  $S_d$  the  $d$ -sized permutation group, we consider measures over a symmetrized compact  $\Omega \subset \mathbb{R}^d$  equipped with the following equivalence relation: for  $\alpha, \beta \in \mathcal{P}(\Omega)$ ,  $\alpha \sim \beta \iff \exists \sigma \in S_d, \beta = \sigma_\# \alpha$ , such that a measure and its permuted counterpart are indistinguishable in the corresponding quotient space, denoted alternatively  $\mathcal{P}(\Omega)_{/\sim}$  or  $\mathcal{R}(\Omega)_{/\sim}$ . A function  $\varphi : \Omega^n \rightarrow \mathbb{R}$  is said to be invariant (by permutations of coordinates) iff  $\forall \sigma \in S_d, \varphi(x_1, \dots, x_n) = \varphi(\sigma(x_1), \dots, \sigma(x_n))$  (Definition 1).

**Tensorization.** Letting  $X$  and  $Y$  respectively denote two random vectors on  $\mathcal{R}(\mathbb{R}^d)$  and  $\mathcal{R}(\mathbb{R}^p)$ , the tensor product vector  $X \otimes Y$  is defined as:  $X \otimes Y \stackrel{\text{def.}}{=} (X', Y') \in \mathcal{R}(\mathbb{R}^d \times \mathbb{R}^p)$ , where  $X'$  and  $Y'$  are independent and have the same law as  $X$  and  $Y$ , i.e.  $d(\alpha_{X \otimes Y})(x, y) = d\alpha_X(x) d\alpha_Y(y)$ . In the finite case, for  $\alpha_X = \frac{1}{n} \sum_i \delta_{x_i}$  and  $\alpha_Y = \frac{1}{m} \sum_j \delta_{y_j}$ , then  $\alpha_{X \otimes Y} = \frac{1}{nm} \sum_{i,j} \delta_{x_i, y_j}$ , weighted sum of Dirac measures on all pairs  $(x_i, y_j)$ . The  $k$ -fold tensorization of a random vector  $X \sim \alpha_X$ , with law  $\alpha_X^{\otimes k}$ , generalizes the above construction to the case of  $k$  independent random variables with law  $\alpha_X$ . Tensorization will be used to define the law of datasets, and design universal architectures (Appendix C).

**Invariant layers.** In the general case, a  $G$ -invariant layer  $f_\varphi$  with invariant map  $\varphi : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^r$  such that  $\varphi$  satisfies

$$\forall (x_1, x_2) \in (\mathbb{R}^d)^2, \forall \sigma \in G, \varphi(\sigma(x_1), \sigma(x_2)) = \varphi(x_1, x_2)$$

is defined as

$$f_\varphi : X \in \mathcal{R}(\mathbb{R}^d)_{/\sim} \mapsto \mathbb{E}_{X' \sim X} [\varphi(X, X')] \in \mathcal{R}(\mathbb{R}^r)_{/\sim}$$

where the expectation is taken over  $X' \sim X$ . Note that considering the couple  $(X, X')$  of independent random vectors  $X' \sim X$  amounts to consider the tensorized law  $\alpha_X \otimes \alpha_X$ .

**Remark 9.** Taking as input a discrete distribution  $\alpha_X = \sum_{i=1}^n w_i \delta_{x_i}$ , the invariant layer outputs another discrete distribution  $\alpha_Y = \sum_{i=1}^n w_i \delta_{y_i}$  with  $y_i = \sum_{j=1}^n w_j \varphi(x_i, x_j)$ ; each input point  $x_i$  is mapped onto  $y_i$  summarizing the pairwise interactions with  $x_i$  after  $\varphi$ .

**Remark 10.** (Generalization to arbitrary invariance groups) The definition of invariant  $\varphi$  can be generalized to arbitrary invariance groups operating on  $\mathbb{R}^d$ , in particular sub-groups of the permutation group  $S_d$ . After (Maron et al., 2020) (Thm 5), a simple and only way to design an invariant linear function is to consider  $\varphi(z, z') = \psi(z + z')$  with  $\psi$  being  $G$ -invariant. How to design invariant functions in the general non-linear case is left for further work.

**Remark 11.** Invariant layers can also be generalized to handle higher order interactions functionals, namely  $f_\varphi(X) \stackrel{\text{def.}}{=} \mathbb{E}_{X_2, \dots, X_N \sim X} [\varphi(X, X_2, \dots, X_N)]$ , which amounts to consider, in the discrete case,  $N$ -uple of inputs points  $(x_{j_1}, \dots, x_{j_N})$ .

## B PROOFS ON REGULARITY

**Wasserstein distance.** The regularity of the involved functionals is measured w.r.t. the 1-Wasserstein distance between two probability distributions  $(\alpha, \beta) \in \mathcal{P}(\mathbb{R}^d)$

$$W_1(\alpha, \beta) \stackrel{\text{def.}}{=} \min_{\pi_1=\alpha, \pi_2=\beta} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\| d\pi(x, y) \stackrel{\text{def.}}{=} \min_{X \sim \alpha, Y \sim \beta} \mathbb{E}(\|X - Y\|)$$

where the minimum is taken over measures on  $\mathbb{R}^d \times \mathbb{R}^d$  with marginals  $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$ .  $W_1$  is known to be a norm (Santambrogio, 2015), that can be conveniently computed using

$$W_1(\alpha, \beta) = W_1(\alpha - \beta) = \max_{\text{Lip}(g) \leq 1} \int_{\mathbb{R}^d} g d(\alpha - \beta),$$

where  $\text{Lip}(g)$  is the Lipschitz constant of  $g : \mathbb{R}^d \rightarrow \mathbb{R}$  with respect to the Euclidean norm (unless otherwise stated). For simplicity and by abuse of notations,  $W_1(X, Y)$  is used instead of  $W_1(\alpha, \beta)$  when  $X \sim \alpha$  and  $Y \sim \beta$ . The convergence in law denoted  $\rightarrow$  is equivalent to the convergence in Wasserstein distance in the sense that  $X_k \rightarrow X$  is equivalent to  $W_1(X_k, X) \rightarrow 0$ .

**Permutation-invariant Wasserstein distance.** The Wasserstein distance is quotiented according to the permutation-invariance equivalence classes: for  $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$

$$\overline{W}_1(\alpha, \beta) \stackrel{\text{def.}}{=} \min_{\sigma \in S_d} W_1(\sigma_{\#} \alpha, \beta) = \min_{\sigma \in S_d} \max_{\text{Lip}(g) \leq 1} \int_{\mathbb{R}^d} g \circ \sigma d\alpha - \int_{\mathbb{R}^d} g d\beta$$

such that  $\overline{W}_1(\alpha, \beta) = 0 \iff \alpha \sim \beta$ .  $\overline{W}_1$  defines a norm on  $\mathcal{P}(\mathbb{R}^d)_{/\sim}$ .

**Lipschitz property.** A map  $f : \mathcal{R}(\mathbb{R}^d) \rightarrow \mathcal{R}(\mathbb{R}^r)$  is continuous for the convergence in law (aka the weak\* of measures) if for any sequence  $X_k \rightarrow X$ , then  $f(X_k) \rightarrow f(X)$ . Such a map is furthermore said to be  $C$ -Lipschitz for the permutation invariant 1-Wasserstein distance if

$$\forall (X, Y) \in (\mathcal{R}(\mathbb{R}^d)_{/\sim})^2, \overline{W}_1(f(X), f(Y)) \leq C \overline{W}_1(X, Y). \quad (7)$$

Lipschitz properties enable us to analyze robustness to input perturbations, since it ensures that if the input distributions of random vectors are close in the permutation invariant Wasserstein sense, the corresponding output laws are close, too.

### Proofs of section 3.2.

*Proof.* (Proposition 1). For  $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$ , Proposition 1 from (De Bie et al., 2019) yields  $W_1(f_{\varphi}(\alpha), f_{\varphi}(\beta)) \leq 2r \text{Lip}(\varphi) W_1(\alpha, \beta)$ , hence, for  $\sigma \in G$ ,

$$\begin{aligned} W_1(\sigma_{\#} f_{\varphi}(\alpha), f_{\varphi}(\beta)) &\leq W_1(\sigma_{\#} f_{\varphi}(\alpha), f_{\varphi}(\alpha)) + W_1(f_{\varphi}(\alpha), f_{\varphi}(\beta)) \\ &\leq W_1(\sigma_{\#} f_{\varphi}(\alpha), f_{\varphi}(\alpha)) + 2r \text{Lip}(\varphi) W_1(\alpha, \beta) \end{aligned}$$

hence, taking the infimum over  $\sigma$  yields

$$\begin{aligned} \overline{W}_1(f_{\varphi}(\alpha), f_{\varphi}(\beta)) &\leq \overline{W}_1(f_{\varphi}(\alpha), f_{\varphi}(\alpha)) + 2r \text{Lip}(\varphi) W_1(\alpha, \beta) \\ &\leq 2r \text{Lip}(\varphi) W_1(\alpha, \beta) \end{aligned}$$

Since  $f_{\varphi}$  is invariant, for  $\sigma \in G$ ,  $f_{\varphi}(\mathbf{z}) = f_{\varphi}(\sigma_{\#} \mathbf{z})$ ,

$$\overline{W}_1(f_{\varphi}(\alpha), f_{\varphi}(\beta)) \leq 2r \text{Lip}(\varphi) W_1(\sigma_{\#} \alpha, \beta)$$

Taking the infimum over  $\sigma$  yields the result.  $\square$

*Proof.* (Proposition 2). To upper bound  $\overline{W}_1(\xi_{\#} f_{\varphi}(\tau_{\#} \alpha), f_{\varphi}(\alpha))$  for  $\alpha \in \mathcal{P}(\mathbb{R}^d)$ , we proceed as follows, using proposition 3 from (De Bie et al., 2019) and proposition 1:

$$\begin{aligned} W_1(\xi_{\#} f_{\varphi}(\tau_{\#} \alpha), f_{\varphi}(\alpha)) &\leq W_1(\xi_{\#} f_{\varphi}(\tau_{\#} \alpha), f_{\varphi}(\tau_{\#} \alpha)) + W_1(f_{\varphi}(\tau_{\#} \alpha), f_{\varphi}(\alpha)) \\ &\leq \|\xi - id\|_{L^1(f_{\varphi}(\tau_{\#} \alpha))} + \text{Lip}(f_{\varphi}) W_1(\tau_{\#} \alpha, \alpha) \\ &\leq \sup_{y \in f_{\varphi}(\tau(\Omega))} \|\xi(y) - y\|_2 + 2r \text{Lip}(\varphi) \sup_{x \in \Omega} \|\tau(x) - x\|_2 \end{aligned}$$

For  $\sigma \in G$ , we get

$$W_1(\sigma_{\#}\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), f_{\varphi}(\alpha)) \leq W_1(\sigma_{\#}\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\alpha)) + W_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), f_{\varphi}(\alpha))$$

Taking the infimum over  $\sigma$  yields

$$\begin{aligned} \overline{W}_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), f_{\varphi}(\alpha)) &\leq W_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), f_{\varphi}(\alpha)) \\ &\leq \sup_{y \in f_{\varphi}(\tau(\Omega))} \|\xi(y) - y\|_2 + 2rC(\varphi) \sup_{x \in \Omega} \|\tau(x) - x\|_2 \end{aligned}$$

Similarly, for  $\alpha, \beta \in (\mathcal{P}(\mathbb{R}^d))^2$ ,

$$\begin{aligned} W_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\beta)) &\leq \text{Lip}(\xi) W_1(f_{\varphi}(\tau_{\#}\alpha), f_{\varphi}(\tau_{\#}\beta)) \\ &\leq \text{Lip}(\xi) \text{Lip}(f_{\varphi}) W_1(\tau_{\#}\alpha, \tau_{\#}\beta) \\ &\leq 2r \text{Lip}(\varphi) \text{Lip}(\xi) \text{Lip}(\tau) W_1(\alpha, \beta) \end{aligned}$$

hence, for  $\sigma \in G$ ,

$$\begin{aligned} W_1(\sigma_{\#}\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\beta)) &\leq W_1(\sigma_{\#}\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\alpha)) \\ &\quad + W_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\beta)) \end{aligned}$$

and taking the infimum over  $\sigma$  yields

$$\begin{aligned} \overline{W}_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\beta)) &\leq W_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\beta)) \\ &\leq 2r \text{Lip}(\varphi) \text{Lip}(\xi) \text{Lip}(\tau) W_1(\alpha, \beta) \end{aligned}$$

Since  $\tau$  is equivariant: namely, for  $\alpha \in \mathcal{P}(\mathbb{R}^d)$ ,  $\sigma \in G$ ,  $\tau_{\#}(\sigma_{\#}\alpha) = \sigma_{\#}(\tau_{\#}\alpha)$ , hence, since  $f_{\varphi}$  is invariant,  $f_{\varphi}(\tau_{\#}(\sigma_{\#}\alpha)) = f_{\varphi}(\sigma_{\#}(\tau_{\#}\alpha)) = f_{\varphi}(\tau_{\#}\alpha)$ , hence for  $\sigma \in G$ ,

$$\overline{W}_1(\xi_{\#}f_{\varphi}(\tau_{\#}\alpha), \xi_{\#}f_{\varphi}(\tau_{\#}\beta)) \leq 2r \text{Lip}(\varphi) \text{Lip}(\xi) \text{Lip}(\tau) W_1(\sigma_{\#}\alpha, \beta)$$

Taking the infimum over  $\sigma$  yields the result.  $\square$

## C PROOFS ON UNIVERSALITY

**Detailed proof of Theorem 1.** This paragraph details the result in the case of  $S_d$ -invariance, while the next one focuses on invariances w.r.t. products of permutations. Before providing a proof of Theorem 1 we first state two useful lemmas. Lemma 1 is mentioned for completeness, referring the reader to De Bie et al. (2019), Lemma 1 for a proof.

**Lemma 1.** Let  $(S_j)_{j=1}^N$  be a partition of a domain including  $\Omega$  ( $S_j \subset \mathbb{R}^d$ ) and let  $x_j \in S_j$ . Let  $(\varphi_j)_{j=1}^N$  a set of bounded functions  $\varphi_j : \Omega \rightarrow \mathbb{R}$  supported on  $S_j$ , such that  $\sum_{j=1}^N \varphi_j = 1$  on  $\Omega$ . For  $\alpha \in \mathcal{P}(\Omega)$ , we denote  $\hat{\alpha}_N \stackrel{\text{def}}{=} \sum_{j=1}^N \alpha_j \delta_{x_j}$  with  $\alpha_j \stackrel{\text{def}}{=} \int_{S_j} \varphi_j d\alpha$ . One has, denoting  $\Delta_j \stackrel{\text{def}}{=} \max_{x \in S_j} \|x_j - x\|$ ,

$$W_1(\hat{\alpha}_N, \alpha) \leq \max_{1 \leq j \leq N} \Delta_j.$$

**Lemma 2.** Let  $f : \mathbb{R}^d \rightarrow \mathbb{R}^q$  a  $1/p$ -Hölder continuous function ( $p \geq 1$ ), then there exists a constant  $C > 0$  such that for all  $\alpha, \beta \in \mathcal{P}(\mathbb{R}^d)$ ,  $W_1(f_{\#}\alpha, f_{\#}\beta) \leq C W_1(\alpha, \beta)^{1/p}$ .

*Proof.* For any transport map  $\pi$  with marginals  $\alpha$  and  $\beta$ ,  $1/p$ -Hölderness of  $f$  with constant  $C$  yields  $\int \|f(x) - f(y)\|_2 d\pi(x, y) \leq C \int \|x - y\|_2^{1/p} d\pi(x, y) \leq C \left( \int \|x - y\|_2 d\pi(x, y) \right)^{1/p}$  using Jensen's inequality ( $p \leq 1$ ). Taking the infimum over  $\pi$  yields  $W_1(f_{\#}\alpha, f_{\#}\beta) \leq C W_1(\alpha, \beta)^{1/p}$ .  $\square$

Now we are ready to dive into the proof. Let  $\alpha \in \mathcal{P}(\mathbb{R}^d)$ . We consider:

- $h : x = (x_1, \dots, x_d) \in \mathbb{R}^d \mapsto \left( \sum_{1 \leq j_1 < \dots < j_i \leq d} x_{j_1} \cdot \dots \cdot x_{j_i} \right)_{i=1 \dots d} \in \mathbb{R}^d$  the collection of  $d$  elementary symmetric polynomials;  $h$  does not lead to a loss in information, in the sense that it generates the ring of  $S_d$ -invariant polynomials (see for instance Cox et al. (2007), chapter 7, theorem 3) while preserving the classes (see the proof of Lemma 2, appendix D from Maron et al. (2020));

- $h$  is obviously not injective, so we consider  $\pi : \mathbb{R}^d \rightarrow \mathbb{R}^d/S_d$  the projection onto  $\mathbb{R}^d/S_d$ :  $h = \tilde{h} \circ \pi$  such that  $\tilde{h}$  is bijective from  $\pi(\Omega)$  to its image  $\Omega'$ , compact of  $\mathbb{R}^d$ ;  $\tilde{h}$  and  $\tilde{h}^{-1}$  are continuous;
- Let  $(\varphi_i)_{i=1\dots N}$  the piecewise affine P1 finite element basis, which are hat functions on a discretization  $(S_i)_{i=1\dots N}$  of  $\Omega' \subset \mathbb{R}^d$ , with centers of cells  $(y_i)_{i=1\dots N}$ . We then define  $g : x \in \mathbb{R}^d \mapsto (\varphi_1(x), \dots, \varphi_N(x)) \in \mathbb{R}^N$ ;
- $f : (\alpha_1, \dots, \alpha_N) \in \mathbb{R}^N \mapsto \mathcal{F} \left( \sum_{i=1}^N \alpha_i \delta_{\tilde{h}^{-1}(y_i)} \right) \in \mathbb{R}$ .

We approximate  $\mathcal{F}$  using the following steps:

- Lemma 1 (see Lemma 1 from De Bie et al. (2019)) yields that  $h_{\#}\alpha$  and  $\widehat{h_{\#}\alpha} = \sum_{i=1}^N \alpha_i \delta_{y_i}$  are close:  $W_1(h_{\#}\alpha, \widehat{h_{\#}\alpha}) \leq \sqrt{d}/N^{1/d}$ ;
- The map  $\tilde{h}^{-1}$  is regular enough ( $1/d$ -Hölder) such that according to Lemma 2, there exists a constant  $C > 0$  such that

$$W_1(\tilde{h}_{\#}^{-1}(h_{\#}\alpha), \tilde{h}_{\#}^{-1}\widehat{h_{\#}\alpha}) \leq C W_1(h_{\#}\alpha, \widehat{h_{\#}\alpha})^{1/d} \leq C d^{1/2d}/N^{1/d^2}$$

$$\text{Hence } \overline{W}_1(\alpha, \tilde{h}_{\#}^{-1}\widehat{h_{\#}\alpha}) := \inf_{\sigma \in S_d} W_1(\sigma_{\#}\alpha, \tilde{h}_{\#}^{-1}\widehat{h_{\#}\alpha}) \leq C d^{1/2d}/N^{1/d^2}.$$

Note that  $h$  maps the roots of polynomial  $\prod_{i=1}^d (X - x^{(i)})$  to its coefficients (up to signs). Theorem 1.3.1 from Rahman & Schmeisser (2002) yields continuity and  $1/d$ -Hölderness of the reverse map. Hence  $\tilde{h}^{-1}$  is  $1/d$ -Hölder.

- Since  $\Omega$  is compact, by Banach-Alaoglu theorem, we obtain that  $\mathcal{P}(\Omega)$  is weakly-\* compact, hence  $\mathcal{P}(\Omega)_{/\sim}$  also is. Since  $\mathcal{F}$  is continuous, it is thus uniformly weak-\* continuous: for any  $\varepsilon > 0$ , there exists  $\delta > 0$  such that  $\overline{W}_1(\alpha, \tilde{h}_{\#}^{-1}\widehat{h_{\#}\alpha}) \leq \delta$  implies  $|\mathcal{F}(\alpha) - \mathcal{F}(\tilde{h}_{\#}^{-1}\widehat{h_{\#}\alpha})| < \varepsilon$ . Choosing  $N$  large enough such that  $C d^{1/2d}/N^{1/d^2} \leq \delta$  therefore ensures that  $|\mathcal{F}(\alpha) - \mathcal{F}(\tilde{h}_{\#}^{-1}\widehat{h_{\#}\alpha})| < \varepsilon$ .

### Extension of Theorem 1 to products of permutation groups.

**Corollary 1.** Let  $\mathcal{F} : \mathcal{P}(\Omega)_{/\sim} \rightarrow \mathbb{R}$  a continuous  $S_{d_1} \times \dots \times S_{d_n}$ -invariant map ( $\sum_i d_i = d$ ), where  $\Omega$  is a symmetrized compact over  $\mathbb{R}^d$ . Then  $\forall \varepsilon > 0$ , there exists three continuous maps  $f, g, h$  such that

$$\forall \alpha \in \mathcal{M}_+^1(\Omega)_{/\sim}, |\mathcal{F}(\alpha) - f \circ \mathbb{E} \circ g(h_{\#}\alpha)| < \varepsilon$$

where  $h$  is invariant;  $g, h$  are independent of  $\mathcal{F}$ .

*Proof.* We provide a proof in the case  $G = S_d \times S_p$ , which naturally extends to any product group  $G = S_{d_1} \times \dots \times S_{d_n}$ . We trade  $h$  for the collection of elementary symmetric polynomials in the first  $d$  variables; and in the last  $p$  variables:  $h : (x_1, \dots, x_d, y_1, \dots, y_p) \in \mathbb{R}^{d+p} \mapsto ([\sum_{1 \leq j_1 < \dots < j_d \leq d} x_{j_1} \dots x_{j_d}]_{i=1}^d; [\sum_{1 \leq j_1 < \dots < j_p \leq p} y_{j_1} \dots y_{j_p}]_{i=1}^p) \in \mathbb{R}^{d+p}$  up to normalizing constants (see Lemma 4). Step 1 (in Lemma 3) consists in showing that  $h$  does not lead to a loss of information, in the sense that it generates the ring of  $S_d \times S_p$ -invariant polynomials. In step 2 (in Lemma 4), we show that  $\tilde{h}^{-1}$  is  $1/\max(d, p)$ -Hölder. Combined with the proof of Theorem 1, this amounts to showing that the concatenation of Hölder functions (up to normalizing constants) is Hölder. With these ingredients, the sketch of the previous proof yields the result.  $\square$

**Lemma 3.** Let the collection of symmetric invariant polynomials  $[P_i(X_1, \dots, X_d)]_{i=1}^d \stackrel{\text{def.}}{=} [\sum_{1 \leq j_1 < \dots < j_i \leq d} X_{j_1} \dots X_{j_i}]_{i=1}^d$  and  $[Q_i(Y_1, \dots, Y_p)]_{i=1}^p = [\sum_{1 \leq j_1 < \dots < j_i \leq p} Y_{j_1} \dots Y_{j_i}]_{i=1}^p$ . The  $d+p$ -sized family  $(P_1, \dots, P_d, Q_1, \dots, Q_p)$  generates the ring of  $S_d \times S_p$ -invariant polynomials.

*Proof.* The result comes from the fact the fundamental theorem of symmetric polynomials (see Cox et al. (2007) chapter 7, theorem 3) does not depend on the base field. Every  $S_d \times S_p$ -invariant polynomial  $P(X_1, \dots, X_d, Y_1, \dots, Y_p)$  is also  $S_d \times I_p$ -invariant with coefficients in  $\mathbb{R}[Y_1, \dots, Y_p]$ , hence it can be written  $P = R(Y_1, \dots, Y_p)(P_1, \dots, P_d)$ . It is then also  $S_p$ -invariant with coefficients in  $\mathbb{R}[P_1, \dots, P_d]$ , hence it can be written  $P = S(Q_1, \dots, Q_p)(P_1, \dots, P_d) \in \mathbb{R}[P_1, \dots, P_d, Q_1, \dots, Q_p]$ .  $\square$

**Lemma 4.** Let  $h : (x, y) \in \Omega \subset \mathbb{R}^{d+p} \mapsto (f(x)/C_1, g(y)/C_2) \in \mathbb{R}^{d+p}$  where  $\Omega$  is compact,  $f : \mathbb{R}^d \rightarrow \mathbb{R}^d$  is  $1/d$ -Hölder with constant  $C_1$  and  $g : \mathbb{R}^p \rightarrow \mathbb{R}^p$  is  $1/p$ -Hölder with constant  $C_2$ . Then  $h$  is  $1/\max(d, p)$ -Hölder.

*Proof.* Without loss of generality, we consider  $d > p$  so that  $\max(d, p) = d$ , and  $f, g$  normalized (f.i.  $\forall x, x_0 \in (\mathbb{R}^d)^2, \|f(x) - f(x_0)\|_1 \leq \|x - x_0\|_1^{1/d}$ ). For  $(x, y), (x_0, y_0) \in \Omega^2$ ,  $\|h(x, y) - h(x_0, y_0)\|_1 \leq \|f(x) - f(x_0)\|_1 + \|g(y) - g(y_0)\|_1 \leq \|x - x_0\|_1^{1/d} + \|y - y_0\|_1^{1/p}$  since both  $f, g$  are Hölder. We denote  $D$  the diameter of  $\Omega$ , such that both  $\|x - x_0\|_1/D \leq 1$  and  $\|y - y_0\|_1/D \leq 1$  hold. Therefore  $\|h(x, y) - h(x_0, y_0)\|_1 \leq D^{1/d} \left(\frac{\|x - x_0\|_1}{D}\right)^{1/d} + D^{1/p} \left(\frac{\|y - y_0\|_1}{D}\right)^{1/p} \leq 2^{1-1/d} D^{1/p-1/d} \|(x, y) - (x_0, y_0)\|_1^{1/d}$  using Jensen’s inequality, hence the result.  $\square$

In the next two paragraphs, we focus the case of  $S_d$ -invariant functions for the sake of clarity, without loss of generality. Indeed, the same technique applies to  $G$ -invariant functions as  $h$  in that case has the same structure: its first  $d_X$  components are  $S_{d_X}$ -invariant functions of the first  $d_X$  variables and its last  $d_Y$  components are  $S_{d_Y}$ -invariant functions of the last variables.

#### Extension of Theorem 1 to distributions on spaces of varying dimension.

**Corollary 2.** Let  $I = [0; 1]$  and, for  $k \in [1; d_m]$ ,  $\mathcal{F}_k : \mathcal{P}(I^k) \rightarrow \mathbb{R}$  continuous and  $S_k$ -invariant. Suppose  $(\mathcal{F}_k)_{k=1 \dots d_m-1}$  are restrictions of  $\mathcal{F}_{d_m}$ , namely,  $\forall \alpha_k \in \mathcal{P}(I^k), \mathcal{F}_k(\alpha_k) = \mathcal{F}_{d_m}(\alpha_k \otimes \delta_0^{\otimes d_m-k})$ . Then functions  $f$  and  $g$  from Theorem 1 are uniform: there exists  $f, g$  continuous,  $h_1, \dots, h_{d_m}$  continuous invariant such that

$$\forall k = 1 \dots d_m, \forall \alpha_k \in \mathcal{P}(I^k), |\mathcal{F}_k(\alpha_k) - f \circ \mathbb{E} \circ g(h_{k\#} \alpha_k)| < \varepsilon.$$

*Proof.* Theorem 1 yields continuous  $f, g$  and a continuous invariant  $h_{d_m}$  such that  $\forall \alpha \in \mathcal{P}(I^{d_m}), |\mathcal{F}_{d_m} - f \circ \mathbb{E} \circ g(h_{d_m\#} \alpha)| < \varepsilon$ . For  $k = 1 \dots d_m - 1$ , we denote  $h_k : (x_1, \dots, x_k) \in \mathbb{R}^k \mapsto ((\sum_{1 \leq j_1 < \dots < j_i \leq k} x^{(j_1)} \dots x^{(j_i)})_{i=1 \dots k}, 0 \dots, 0) \in \mathbb{R}^{d_m}$ . With the hypothesis, for  $k = 1 \dots d_m - 1$ ,  $\alpha_k \in \mathcal{P}(I^k)$ , the fact that  $h_{k\#}(\alpha_k) = h_{d_m\#}(\alpha_k \otimes \delta_0^{\otimes d_m-k})$  yields the result.  $\square$

**Approximation by invariant neural networks.** Based on theorem 1,  $\mathcal{F}$  is uniformly close to  $f \circ \mathbb{E} \circ g \circ h$ :

- We approximate  $f$  by a neural network  $f_\theta : x \in \mathbb{R}^N \mapsto C_1 \lambda(A_1 x + b_1) \in \mathbb{R}$ , where  $p_1$  is an integer,  $A_1 \in \mathbb{R}^{p_1 \times N}, C_1 \in \mathbb{R}^{1 \times p_1}$  are weights,  $b_1 \in \mathbb{R}^{p_1}$  is a bias and  $\lambda$  is a non-linearity.
- Since each component  $\varphi_j$  of  $\varphi = g \circ h$  is permutation-invariant, it has the representation  $\varphi_j : x = (x_1, \dots, x_d) \in \mathbb{R}^d \mapsto \rho_j \left( \sum_{i=1}^d u(x_i) \right)$  Zaheer et al. (2017) (which is a special case of our layers with a base function only depending on its first argument, see section 2.2),  $\rho_j : \mathbb{R}^{d+1} \rightarrow \mathbb{R}$ , and  $u : \mathbb{R} \rightarrow \mathbb{R}^{d+1}$  independent of  $j$  (see Zaheer et al. (2017), theorem 7).
- We can approximate  $\rho_j$  and  $u$  by neural networks  $\rho_{j,\theta} : x \in \mathbb{R}^{d+1} \mapsto C_{2,j} \lambda(A_{2,j} x + b_{2,j}) \in \mathbb{R}$  and  $u_\theta : x \in \mathbb{R}^d \mapsto C_3 \lambda(A_3 x + b_3) \in \mathbb{R}^{d+1}$ , where  $p_{2,j}, p_3$  are integers,  $A_{2,j} \in \mathbb{R}^{p_{2,j} \times (d+1)}, C_{2,j} \in \mathbb{R}^{1 \times p_{2,j}}, A_3 \in \mathbb{R}^{p_3 \times 1}, C_3 \in \mathbb{R}^{(d+1) \times p_3}$  are weights and  $b_{2,j} \in \mathbb{R}^{p_{2,j}}, b_3 \in \mathbb{R}^{p_3}$  are biases, and denote  $\varphi_\theta(x) = (\varphi_{j,\theta}(x))_j \stackrel{\text{def}}{=} (\rho_{j,\theta}(\sum_{i=1}^d u_\theta(x_i)))_j$ .

Indeed, we upper-bound the difference of interest  $|\mathcal{F}(\alpha) - f_\theta(\mathbb{E}_{X \sim \alpha}(\varphi_\theta(X)))|$  by triangular inequality by the sum of three terms:

- $|\mathcal{F}(\alpha) - f(\mathbb{E}_{X \sim \alpha}(\varphi(X)))|$
- $|f(\mathbb{E}_{X \sim \alpha}(\varphi(X))) - f_\theta(\mathbb{E}_{X \sim \alpha}(\varphi(X)))|$
- $|f_\theta(\mathbb{E}_{X \sim \alpha}(\varphi(X))) - f_\theta(\mathbb{E}_{X \sim \alpha}(\varphi_\theta(X)))|$

and bound each term by  $\frac{\varepsilon}{3}$ , which yields the result. The bound on the first term directly comes from theorem 1 and yields a constant  $N$  which depends on  $\varepsilon$ . The bound on the second term is a direct application of the universal approximation theorem (UAT) Cybenko (1989); Leshno et al. (1993). Indeed, since  $\alpha$  is a probability measure, input values of  $f$  lie in a compact subset of  $\mathbb{R}^N$ :  $\|\int_{\Omega} g \circ h(x) d\alpha\|_{\infty} \leq \max_{x \in \Omega} \max_i |g_i \circ h(x)|$ , hence the theorem is applicable as long as  $\lambda$  is a nonconstant, bounded and continuous activation function. Let us focus on the third term. Uniform continuity of  $f_{\theta}$  yields the existence of  $\delta > 0$  s.t.  $\|u - v\|_1 < \delta$  implies  $|f_{\theta}(u) - f_{\theta}(v)| < \frac{\varepsilon}{3}$ . Let us apply the UAT: each component  $\varphi_j$  of  $h$  can be approximated by a neural network  $\varphi_{j,\theta}$ . Therefore:

$$\begin{aligned} \|\mathbb{E}_{X \sim \alpha} (\varphi(X) - \varphi_{\theta}(X))\|_1 &\leq \mathbb{E}_{X \sim \alpha} \|\varphi(X) - \varphi_{\theta}(X)\|_1 \leq \sum_{j=1}^N \int_{\Omega} |\varphi_j(x) - \varphi_{j,\theta}(x)| d\alpha(x) \\ &\leq \sum_{j=1}^N \int_{\Omega} |\varphi_j(x) - \rho_{j,\theta}(\sum_{i=1}^d u(x_i))| d\alpha(x) \\ &\quad + \sum_{j=1}^N \int_{\Omega} |\rho_{j,\theta}(\sum_{i=1}^d u(x_i)) - \rho_{j,\theta}(\sum_{i=1}^d u_{\theta}(x_i))| d\alpha(x) \\ &\leq N \frac{\delta}{2N} + N \frac{\delta}{2N} = \delta \end{aligned}$$

using the triangular inequality and the fact that  $\alpha$  is a probability measure. The first term is small by UAT on  $\rho_j$  while the second also is, by UAT on  $u$  and uniform continuity of  $\rho_{j,\theta}$ . Therefore, by uniform continuity of  $f_{\theta}$ , we can conclude.

**Universality of tensorization.** This complementary theorem provides insight into the benefits of tensorization for approximating invariant regression functionals, as long as the test function is invariant.

**Theorem 2.** *The algebra*

$$\mathcal{A}_{\Omega} \stackrel{\text{def}}{=} \left\{ \mathcal{F} : \mathcal{P}(\Omega)_{/\sim} \rightarrow \mathbb{R}, \exists n \in \mathbb{N}, \exists \varphi : \Omega^n \rightarrow \mathbb{R} \text{ invariant}, \forall \alpha, \mathcal{F}(\alpha) = \int_{\Omega^n} \varphi d\alpha^{\otimes n} \right\}$$

where  $\otimes n$  denotes the  $n$ -fold tensor product, is dense in  $\mathcal{C}(\mathcal{M}_+^1(\Omega)_{/\sim})$ .

*Proof.* This result follows from the Stone-Weierstrass theorem. Since  $\Omega$  is compact, by Banach-Alaoglu theorem, we obtain that  $\mathcal{P}(\Omega)$  is weakly-\* compact, hence  $\mathcal{P}(\Omega)_{/\sim}$  also is. In order to apply Stone-Weierstrass, we show that  $\mathcal{A}_{\Omega}$  contains a non-zero constant function and is an algebra that separates points. A (non-zero, constant) 1-valued function is obtained with  $n = 1$  and  $\varphi = 1$ . Stability by scalar is straightforward. For stability by sum: given  $(\mathcal{F}_1, \mathcal{F}_2) \in \mathcal{A}_{\Omega}^2$  (with associated functions  $(\varphi_1, \varphi_2)$  of tensorization degrees  $(n_1, n_2)$ ), we denote  $n \stackrel{\text{def}}{=} \max(n_1, n_2)$  and  $\varphi(x_1, \dots, x_n) \stackrel{\text{def}}{=} \varphi_1(x_1, \dots, x_{n_1}) + \varphi_2(x_1, \dots, x_{n_2})$  which is indeed invariant, hence  $\mathcal{F}_1 + \mathcal{F}_2 = \int_{\Omega^n} \varphi d\alpha^{\otimes n} \in \mathcal{A}_{\Omega}$ . Similarly, for stability by product: denoting this time  $n = n_1 + n_2$ , we introduce the invariant  $\varphi(x_1, \dots, x_n) = \varphi_1(x_1, \dots, x_{n_1}) \times \varphi_2(x_{n_1+1}, \dots, x_n)$ , which shows that  $\mathcal{F} = \mathcal{F}_1 \times \mathcal{F}_2 \in \mathcal{A}_{\Omega}$  using Fubini's theorem. Finally,  $\mathcal{A}_{\Omega}$  separates points: if  $\alpha \neq \nu$ , then there exists a symmetrized domain  $S$  such that  $\alpha(S) \neq \nu(S)$ : indeed, if for all symmetrized domains  $S$ ,  $\alpha(S) = \nu(S)$ , then  $\alpha(\Omega) = \nu(\Omega)$  which is absurd. Taking  $n = 1$  and  $\varphi = 1_S$  (invariant since  $S$  is symmetrized) yields an  $\mathcal{F}$  such that  $\mathcal{F}(\alpha) \neq \mathcal{F}(\nu)$ .  $\square$

## D EXPERIMENTAL VALIDATION, SUPPLEMENTARY MATERIAL

Both DIDA and baselines source code are provided in the last file of the supplementary material.

### D.1 BENCHMARK DETAILS

Three benchmarks are used (Table 3): TOY and UCI, taken from (Jomaa et al., 2019), and OpenML CC-18. TOY includes 10,000 datasets, where instances are distributed along mixtures of Gaussian,

intertwining moons and rings in  $\mathbb{R}^2$ , with 2 to 7 classes. UCI includes 121 datasets from the UCI Irvine repository (Dua & Graff, 2017). **Datasets UCI and OpenML are normalized as follows:** categorical features are one-hot encoded; numerical features are normalized; missing values are imputed with the feature mean (continuous features) or median (for categorical features). Patches are defined as follows. Given an initial dataset, a number  $d_X$  of features and a number  $n$  of examples are uniformly selected in the considered ranges (depending on the benchmark) described in Table 4. A patch is defined by (i) retaining  $n$  examples uniformly selected with replacement in this initial dataset; (ii) retaining  $d_X$  features uniformly selected with replacement among the initial features.

	# datasets	# samples	# features	# labels	test ratio
Toy Dataset	10000	[2048, 8192]	2	[2, 7]	0.3
UCI	121	[10, 130064]	[3, 262]	[2, 100]	0.3
OpenML CC-18	71	[500, 100000]	[5, 3073]	[2, 46]	0.5

Table 3: Benchmarks characteristics

	Patch Identification		Performance Modeling
Dataset	TOY	UCI	OpenML
# Features	2	[2, 15]	[3, 11]
# Examples	200	[200, 500]	[700, 900]

Table 4: Patch Size

## D.2 DETAILED EXPERIMENTAL PROCEDURE: PATCH IDENTIFICATION

The following Algorithm 2 details the learning procedure used to train DIDA, DSS or DATASET2VEC on the patch identification task (Section 4.1, Table 1). Note that function *generate\_patches()* is extracted from the DATASET2VEC source code.

---

### Algorithm 2 Batch Identification

---

- 1:  $\mathcal{F}_\zeta \leftarrow$  meta-feature extractor (DIDA Deep Sets, DSS, or Hand-crafted)
  - 2: **for** iteration=1, 2, ... **do**
  - 3:    $\mathbf{z}_1, \mathbf{z}_2, y \leftarrow$  generate\_patches()    $\triangleright y \leftarrow 1$  if  $\mathbf{z}_1$  and  $\mathbf{z}_2$  are from the same dataset else 0
  - 4:    $m_{f_1} \leftarrow \mathcal{F}_\zeta(\mathbf{z}_1)$
  - 5:    $m_{f_2} \leftarrow \mathcal{F}_\zeta(\mathbf{z}_2)$
  - 6:   Backpropagate  $\text{logloss}(\exp(-\|m_{f_1} - m_{f_2}\|_2), y)$
  - 7: **end for**
- 

## D.3 BASELINE DETAILS

**DATASET2VEC details.** The publicly available implementation of DATASET2VEC<sup>3</sup> does not allow for a random uniform subsampling of all features, hence we have included as baselines: (i) the reported accuracy from (Jomaa et al., 2019); (ii) the computed accuracy from our own implementation of DATASET2VEC, based on a uniform sampling of the features. As said, this implementation only aims at solely making up for the feature sampling procedure. The architecture is the same as reported in (Jomaa et al., 2019), Eq. 4, namely

$$D : \mathbf{z} \in \mathbb{Z}_n(\mathbb{R}^d) \mapsto h \left( \frac{1}{d_X d_Y} \sum_{m=1}^{d_X} \sum_{t=1}^{d_Y} g \left( \frac{1}{n} \sum_{i=1}^n f(x_i[m], y_i[t]) \right) \right) \quad (8)$$

where functions  $f, g, h$  characterizing the architecture are chosen as depicted in the publicly available file *config.py*<sup>4</sup>. More precisely,  $f, g$  are FC(128)-ReLU-ResFC(128, 128, 128)-FC(128) and

<sup>3</sup>See <https://github.com/hadijomaa/dataset2vec>

<sup>4</sup>See <https://github.com/hadijomaa/dataset2vec/blob/master/config.py>

$h$  is FC(128)-ReLU-FC(128)-ReLU where ResFC is a sequence of fully connected layer with skip connection. We provide our implementation of DATASET2VEC in the supplementary material.

**DSS layer details.** We built our own implementation of invariant DSS layers, as follows. Linear invariant DSS layers (see (Maron et al., 2020), Theorem 5, 3.) are of the form

$$L_{inv} : X \in \mathbb{R}^{n \times d} \mapsto L^H\left(\sum_{j=1}^n x_j\right) \in \mathbb{R}^K \quad (9)$$

where  $L^H : \mathbb{R}^d \rightarrow \mathbb{R}^K$  is a linear  $H$ -invariant function. Our applicative setting requires that the implementation accommodates to varying input dimensions  $d$  as well as permutation invariance, hence we consider the Deep Sets representation (see (Zaheer et al., 2017), Theorem 7)

$$L^H : x = (x_1, \dots, x_d) \in \mathbb{R}^d \mapsto \rho\left(\sum_{i=1}^d \varphi(x_i)\right) \in \mathbb{R}^K \quad (10)$$

where  $\varphi : \mathbb{R} \rightarrow \mathbb{R}^{d+1}$  and  $\rho : \mathbb{R}^{d+1} \rightarrow \mathbb{R}^K$  are modelled as (i) purely linear functions; (ii) FC networks, which extends the initial linear setting (9). In our case,  $H = S_{d_x} \times S_{d_y}$ , hence, two invariant layers of the form (9-10) are combined to suit both feature- and label-invariance requirements. Both outputs are concatenated and followed by an FC network to form the DSS meta-features. The last experiments use DSS equivariant layers (see (Maron et al., 2020), Theorem 1), which take the form

$$L_{eq} : X \in \mathbb{R}^{n \times d} \mapsto \left( L_{eq}^1(x_i) + L_{eq}^2\left(\sum_{j \neq i} x_j\right) \right)_{i \in [n]} \in \mathbb{R}^{n \times d} \quad (11)$$

where  $L_{eq}^1$  and  $L_{eq}^2$  are linear  $H$ -equivariant layers. Similarly, both feature- and label-equivariance requirements are handled via the Deep Sets representation of equivariant functions (see (Zaheer et al., 2017), Lemma 3) and concatenated to be followed by an invariant layer, forming the DSS meta-features. All methods are allocated the same number of parameters to ensure fair comparison. We provide our implementation of the DSS layers in the supplementary material.

**NO-FINV-DSS baseline (no invariance in feature permutation).** This baseline aims at showcasing the empirical relevance of the invariance requirement in feature and label permutations, while retaining invariance in permutation with respect to the datasets. To this end, aggregation with respect to the examples is performed as exemplified in (Zaheer et al., 2017), Theorem 2, namely

$$L : \mathbf{z} = (\mathbf{z}_1, \dots, \mathbf{z}_n) \in Z(\mathbb{R}^d) \mapsto \frac{1}{n} \sum_{i=1}^n g(\mathbf{z}_i) \in \mathbb{R}^K \quad (12)$$

where  $g : \mathbb{R}^d \rightarrow \mathbb{R}^K$  is an MLP with FC(128)-ReLU-FC(64)-ReLU-FC(32)-ReLU layers. To ensure label information is captured, the output is concatenated to the mean of labels  $\bar{y} \stackrel{\text{def.}}{=} \frac{1}{n} \sum_{i=1}^n y_i$  and followed by an MLP with FC(1024)-ReLU-FC(700)-ReLU-FC(512) layers. The so-called NO-FINV-DSS baseline defined as such, can be summed up as follows

$$\mathbf{z} \in Z(\mathbb{R}^d) \mapsto \text{MLP}([L(\mathbf{z}); \bar{y}]) \quad (13)$$

**Hand-crafted meta-features.** For the sake of reproducibility, the list of meta-features used in Section 4 is given in Table 5. Note that meta-features related to missing values and categorical features are omitted, as being irrelevant for the considered benchmarks. Hand-crafted meta-features are extracted using `BYU metalearn` library. In total, we extracted 43 meta-features.

#### D.4 PERFORMANCE PREDICTION

**Experimental setting.** Table 6 details all hyper-parameter configurations  $\Theta$  considered in Section 4.2. As said, the learnt meta-features  $\mathcal{F}_\zeta(\mathbf{z})$  can be used in a regression setting, predicting the performance of various ML algorithms on a dataset  $\mathbf{z}$ . Several performance models have been considered



<b>Meta-features</b>	<b>Mean</b>	<b>Min</b>	<b>Max</b>
Quartile2ClassProbability	0.500	0.75	0.25
MinorityClassSize	487.423	426.000	500.000
Quartile3CardinalityOfNumericFeatures	224.354	0.000	976.000
RatioOfCategoricalFeatures	0.347	0.000	1.000
MeanCardinalityOfCategoricalFeatures	0.907	0.000	2.000
SkewCardinalityOfNumericFeatures	0.148	-2.475	3.684
RatioOfMissingValues	0.001	0.000	0.250
MaxCardinalityOfNumericFeatures	282.461	0.000	977.000
Quartile2CardinalityOfNumericFeatures	185.555	0.000	976.000
KurtosisClassProbability	-2.025	-3.000	-2.000
NumberOfNumericFeatures	3.330	0.000	30.000
NumberOfInstancesWithMissingValues	2.800	0.000	1000.000
MaxCardinalityOfCategoricalFeatures	0.917	0.000	2.000
Quartile1CardinalityOfCategoricalFeatures	0.907	0.000	2.000
MajorityClassSize	512.577	500.000	574.000
MinCardinalityOfCategoricalFeatures	0.879	0.000	2.000
Quartile2CardinalityOfCategoricalFeatures	0.915	0.000	2.000
NumberOfCategoricalFeatures	1.854	0.000	27.000
NumberOfFeatures	5.184	4.000	30.000
Dimensionality	0.005	0.004	0.030
SkewCardinalityOfCategoricalFeatures	-0.050	-4.800	0.707
KurtosisCardinalityOfCategoricalFeatures	-1.244	-3.000	21.040
StdevCardinalityOfNumericFeatures	68.127	0.000	678.823
StdevClassProbability	0.018	0.000	0.105
KurtosisCardinalityOfNumericFeatures	-1.060	-3.000	12.988
NumberOfInstances	1000.000	1000.000	1000.000
Quartile3CardinalityOfCategoricalFeatures	0.916	0.000	2.000
NumberOfMissingValues	2.800	0.000	1000.000
Quartile1ClassProbability	0.494	0.463	0.500
StdevCardinalityOfCategoricalFeatures	0.018	0.000	0.707
MeanClassProbability	0.500	0.500	0.500
NumberOfFeaturesWithMissingValues	0.003	0.000	1.000
MaxClassProbability	0.513	0.500	0.574
NumberOfClasses	2.000	2.000	2.000
MeanCardinalityOfNumericFeatures	197.845	0.000	976.000
SkewClassProbability	0.000	-0.000	0.000
Quartile3ClassProbability	0.506	0.500	0.537
MinCardinalityOfNumericFeatures	138.520	0.000	976.000
MinClassProbability	0.487	0.426	0.500
RatioOfInstancesWithMissingValues	0.003	0.000	1.000
Quartile1CardinalityOfNumericFeatures	160.748	0.000	976.000
RatioOfNumericFeatures	0.653	0.000	1.000
RatioOfFeaturesWithMissingValues	0.001	0.000	0.250

Table 5: Hand-crafted meta-features

on top of the meta-features learnt in Section 4.2, for instance (i) a BOHAMIANN network (Springenberg et al., 2016); (ii) Random Forest models, trained under a Mean Squared Error loss between predicted and true performances.

**Results.** Table 7 reports the Mean Squared Error on the test set with performance model BOHAMIANN (Springenberg et al., 2016), comparatively to DSS and hand-crafted ones. Replacing the surrogate model with Random Forest concludes to the same ranking as in Table 7. Figure 3 complements Table 7 in assessing the learnt DIDA meta-features for performance model learning.

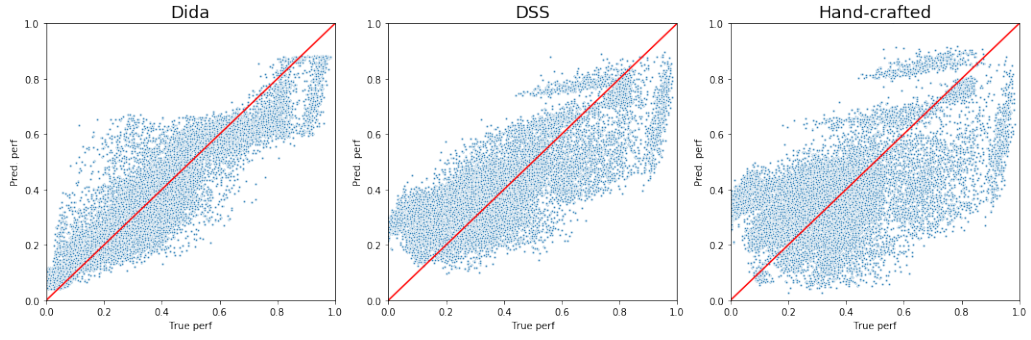
	Parameter	Parameter values	Scale
<b>LR</b>	warm start	True, False	
	fit intercept	True, False	
	tol	[0.00001, 0.0001]	
	C	[1e-4, 1e4]	log
	solver	newton-cg, lbfgs, liblinear, sag, saga	
	max_iter	[5, 1000]	
<b>SVM</b>	kernel	linear, rbf, poly, sigmoid	
	C	[0.0001, 10000]	log
	shrinking	True, False	
	degree	[1, 5]	
	coef0	[0, 10]	
	gamma	[0.0001, 8]	
	max_iter	[5, 1000]	
<b>KNN</b>	n_neighbors	[1, 100]	log
	p	[1, 2]	
	weights	uniform, distance	
<b>SGD</b>	alpha	[0.1, 0.0001]	log
	average	True, False	
	fit_intercept	True, False	
	learning_rate	optimal, invscaling, constant	
	loss	hinge, log, modified_huber, squared_hinge, perceptron	
	penalty	l1, l2, elasticnet	
	tol	[1e-05, 0.1]	log
	eta0	[1e-7, 0.1]	log
	power_t	[1e-05, 0.1]	log
	epsilon	[1e-05, 0.1]	log
	l1_ratio	[1e-05, 0.1]	log

Table 6: Hyper-parameter configurations considered in Section 4.2.

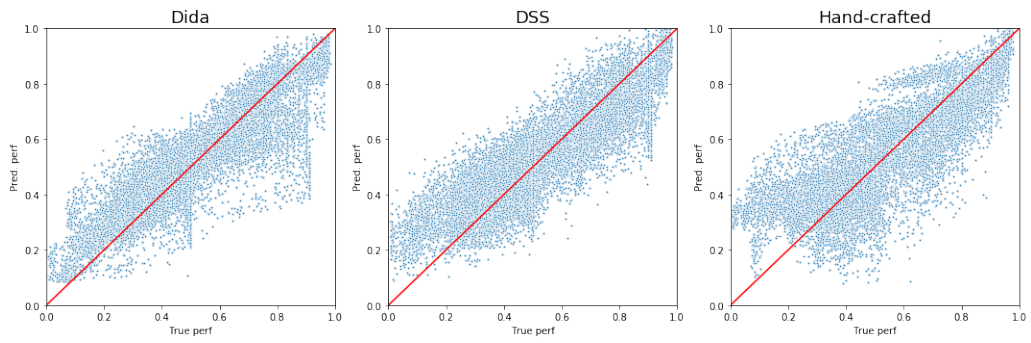
Method	SGD	SVM	LR	KNN
Hand-crafted	0.016 $\pm$ 0.001	0.021 $\pm$ 0.001	0.018 $\pm$ 0.002	0.034 $\pm$ 0.001
DSS (Linear aggregation)	0.015 $\pm$ 0.007	0.020 $\pm$ 0.002	0.019 $\pm$ 0.001	0.025 $\pm$ 0.010
DSS (Equivariant+Invariant)	0.014 $\pm$ 0.002	0.017 $\pm$ 0.003	0.015 $\pm$ 0.003	0.028 $\pm$ 0.003
DSS (Non-linear aggregation)	0.015 $\pm$ 0.009	0.016 $\pm$ 0.003	0.014 $\pm$ 0.001	0.020 $\pm$ 0.005
DIDA	<b>0.012 <math>\pm</math> 0.001</b>	<b>0.015 <math>\pm</math> 0.001</b>	<b>0.010 <math>\pm</math> 0.001</b>	<b>0.009 <math>\pm</math> 0.000</b>

Table 7: Performance modelling, comparative results of DIDA, DSS and Hand-crafted (HC) meta-features: Mean Squared Error (average over 5 runs) on test set, between the true performance and the performance predicted by the trained BOHAMIANN surrogate model, for ML algorithms SVM, LR, kNN, SGD (see text).

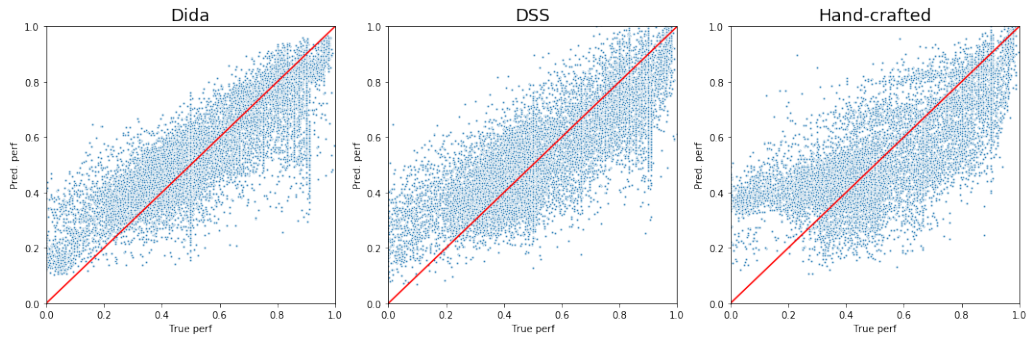
It shows DIDA’s ability to capture more expressive meta-features than both DSS and hand-crafted ones, for all ML algorithms considered.



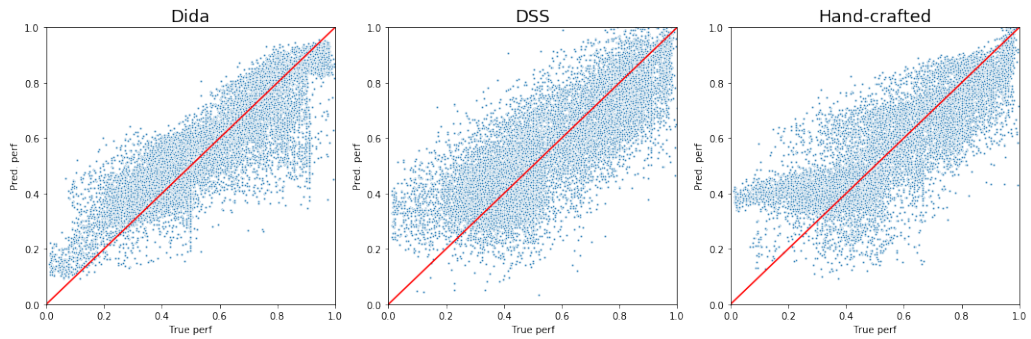
((a)) KNN



((b)) Logistic Regression



((c)) SVM



((d)) SGD

Figure 3: Comparison between the true performance and the performance predicted by the trained surrogate model on DIDA, DSS or Hand-crafted meta-features, for various ML algorithms.

### D.5 STABILITY OF META-FEATURES WITH RESPECT TO SAMPLE AND FEATURE SAMPLING

The robustness of the learned meta-features is investigated along three settings (below). The robustness performance indicators are the average and standard deviation of the distance between the meta-feature vectors and a reference vector. The comparative performances of DIDA and the baseline NO-FINV-DSS (Section D.3) are reported in Fig. 4. Both DIDA and NO-FINV-DSS are trained on Task 1.

Specifically, the three settings aim to measure the robustness w.r.t. (A) the uniform selection of the samples only; (B) the uniform selection of the samples and the permutation of features; (C) the uniform selection of the samples and the features:

- A Considering a fixed set of features, 128 patches are extracted from a dataset  $\mathbf{u}$ . For each patch  $\mathbf{z}$ , DIDA computes a meta-feature vector  $\mathcal{F}_\zeta(\mathbf{z})$  in  $\mathbb{R}^{64}$ . The reference vector is the average of these meta-feature vectors. Fig. 4.A reports the mean and standard deviation of the distance between the meta-feature vectors and their mean (Fig. 4.A).
- B Same as in A, except that for each patch, the features are permuted. The reference vector is the same as in [A]. The mean and standard deviation of the distances between these meta-feature vectors and the reference vector thus reflect the impact of the permutation of features (Fig. 4.B);
- C 128 Patches are uniformly selected (subset of samples, subset of features drawn with replacement), and a meta-feature vector is computed for each patch. The reference vector here is the average of these meta-feature vectors. The mean and standard deviation of the distances between these meta-feature vectors and the reference vector thus reflect the impact of sampling both examples and features (Fig. 4.C).

Fig. 4 shows that for DIDA, similar results are obtained for settings [A] and [B] (the distributions of the meta-feature vectors around the reference vector are similar), while a slightly higher mean and standard deviations are observed for [C]. Quite the contrary, for the baseline NO-FINV-DSS, similar results are obtained for [B] and [C], suggesting that the baseline makes no difference between permuting features and sampling new features.

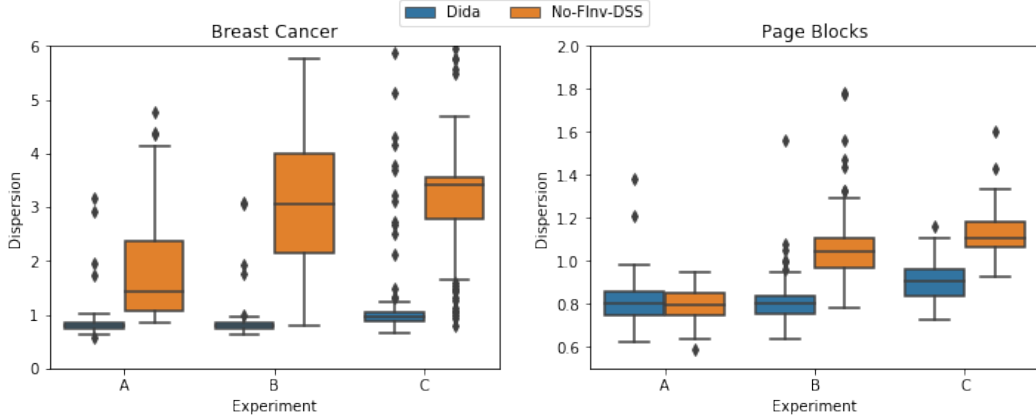


Figure 4: Robustness of meta-features: average and standard deviation of the distance between the meta-feature vectors and their reference vector along the A, B, and C settings (please see text). Left: Breast Cancer dataset. Right: Page Blocks dataset.