

Machine learning over spaces of measures : invariant deep networks and quantile regression

Gwendoline de Bie

▶ **To cite this version:**

Gwendoline de Bie. Machine learning over spaces of measures : invariant deep networks and quantile regression. General Mathematics [math.GM]. Université Paris sciences et lettres, 2020. English. NNT : 2020UPSLE072 . tel-03659503

HAL Id: tel-03659503
https://theses.hal.science/tel-03659503

Submitted on 5 May 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PSL
UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à l'Ecole Normale Supérieure

Machine learning over spaces of measures: invariant deep networks and quantile regression.

Soutenue par

Gwendoline De Bie

Le 30 novembre 2020

Ecole doctorale n° 386

Sciences Mathématiques de Paris Centre

Spécialité

Mathématiques Appliquées

Composition du jury :

Jérémie BIGOT Université de Bordeaux	*Président*
Yaron LIPMAN Weizmann Institute of Science	*Rapporteur*
Jean-Michel LOUBES Université Toulouse Paul Sabatier	*Rapporteur*
Rémi FLAMARY CMAP, Ecole Polytechnique	*Examinateur*
Michèle SEBAG Laboratoire de Recherche en Informatique	*Examinateur*
Guillaume CARLIER Université Paris-Dauphine	*Examinateur*
Gabriel PEYRE Ecole Normale Supérieure	*Directeur de thèse*
Marco CUTURI ENSAE, Google Brain	*Co-directeur de thèse*

ENS | PSL

Summary

Modeling data using probability distributions is a simple yet powerful way to address countless problems in statistics and learning. Typical applicative topics encountered in this thesis encompass modeling population dynamics in biology, summarizing complex datasets for automated machine learning, and public policy evaluation thanks to quantile regression. This thesis develops numerical schemes with provable performance guarantees to perform machine learning over the space of probability distributions. Manipulating such probability distributions requires new type of computational methods, which can cope with the discretization of distributions using point clouds and can integrate additional invariances of the problems. This raises both computational challenges (providing scalable and problem-independent numerical schemes) and theoretical questions (ensuring smoothness and expressiveness of the models for the topology of the convergence in law) which are addressed in this thesis. Optimal transport (OT), which offers a geometrical toolbox to compare probability distributions, is the cornerstone of this work. More precisely, we leverage the entropic regularization approach to OT, to enable scalable models which can be trained by gradient descent methods. In Chapter 1, we introduce a new class of neural network architectures processing probability measures in their Lagrangian form (obtained by sampling) as both inputs and outputs. The formulation is versatile enough to adapt to desired tasks from classification, regression to training of generative networks, and is characterized by robustness and universal approximation properties. In Chapter 2, we show that this framework can be adapted to perform regression with customized invariance requirements on probability measure inputs, in a way that also preserves its robustness and approximation capabilities. This method is proven to be of interest to design expressive, adaptable summaries of datasets referred to as "meta-features", in the context of automated machine learning. Finally, we consider probabilities as objects of interest for inference in Chapter 3: we demonstrate that the resort to entropy eases the computation of conditional multivariate quantiles. We introduce the regularized vector quantile regression problem, provide a scalable algorithm to compute multivariate quantiles and show that it benefits from desirable asymptotic properties.

Résumé

Modéliser des données à l'aide de distributions de probabilité est un moyen simple mais puissant de résoudre d'innombrables problèmes en statistiques et en apprentissage. Les sujets applicatifs typiques rencontrés dans cette thèse comprennent la modélisation des dynamiques de populations en biologie, la synthèse de bases de données complexes pour l'apprentissage automatique et l'évaluation des politiques publiques par la régression de quantile. Manipuler de telles distributions nécessite un nouveau type de méthodes computationnelles, adaptées à la discrétisation des distributions par des nuages de points et pouvant incorporer des invariances supplémentaires. Cela soulève à la fois des défis de calcul (fournir des schémas numériques efficaces et indépendants du problème) et des questions théoriques (assurer la régularité et l'expressivité des modèles pour la topologie de la convergence en loi) qui sont abordés dans cette thèse. Le transport optimal (TO) est la pierre angulaire de ce travail, qui propose une boîte à outils géométrique pour comparer des distributions de probabilité. Plus précisément, nous exploitons l'approche de régularisation entropique du TO, construisant des modèles efficaces qui peuvent être appris par des méthodes de descente de gradient. Dans le chapitre 1, nous introduisons une nouvelle classe d'architectures neuronales qui gère des mesures de probabilité sous leur forme lagrangienne (obtenue par échantillonnage) en tant qu'entrées et sorties. La formulation est suffisamment polyvalente pour s'adapter à la variété des tâches souhaitées, de la classification et de la régression aux réseaux génératifs, et se caractérise par sa robustesse et ses propriétés d'approximation universelle. Dans le chapitre 2, nous montrons que ce cadre peut être adapté pour effectuer des tâches de régression avec invariances additionnelles dont les entrées sont des mesures de probabilité, en préservant sa robustesse et ses capacités d'approximation. Cette méthode est utilisée pour concevoir des résumés expressifs et adaptables de bases de données, appelés "meta-features", dans le contexte de l'apprentissage automatisé. Enfin, nous considérons les probabilités comme des objets d'intérêt pour l'inférence au chapitre 3: nous montrons que le recours à l'entropie facilite le calcul des quantiles conditionnels multivariés. Nous introduisons le problème de régression de quantile vectoriel régularisé, fournissons un algorithme efficace pour calculer les quantiles multivariés et montrons qu'il bénéficie de propriétés asymptotiques souhaitables.

Contents

1 Remerciements

2 Introduction **1**
1 Background 1
2 Summary of Contributions 7

3 Notations **19**

4 Chapter 1: Stochastic Deep Networks **21**
1 Introduction 22
2 Stochastic Deep Architectures 25
2.1 Notion of Elementary Block 25
2.2 Building Stochastic Deep Architectures 27
3 Theoretical Guarantees 28
3.1 Convergence in Law Topology 28
3.2 Regularity of Building blocks 29
3.3 Approximation Theorems 33
4 Applications 40
4.1 Classification tasks 40
4.2 Generative networks 42
4.3 Dynamics Prediction 42

5 Chapter 2: Distribution-Based Invariant Deep Networks for Automated Machine Learning **47**
1 Introduction 48
2 Distribution-Based Invariant Networks for Meta-Feature Learning 51
2.1 Invariant Functions of Discrete Distributions 51
2.2 Distribution-Based Invariant Layers 52
2.3 Learning Dataset Meta-features from Distributions . . 54
3 Theoretical Analysis 55
3.1 Optimal Transport Comparison of Datasets 55
3.2 Regularity of Distribution-Based Invariant Layers . . . 56
3.3 Universality of Invariant Layers 58
4 Learning Meta-Features: Proof of Concept 65
4.1 Distribution Identification 66
4.2 Performance Model Learning 68
5 Appendix 70

5.1 Benchmark details 70
5.2 Detailed experimental procedure: Patch Identification 71
5.3 Baseline Details 72
5.4 Performance Prediction 74
5.5 Stability of meta-features with respect to sample and feature sampling 74

6 Chapter 3: Regularized Vector Quantile Regression 79
1 Introduction 80
2 Several Characterizations of Quantiles 83
2.1 Quantiles 83
2.2 Conditional Quantiles 84
3 Quantile Regression 86
3.1 Specified Quantile Regression 86
3.2 Quasi-Specified Quantile Regression 87
3.3 Quantile Regression without specification 90
4 Vector Quantile Regression 92
4.1 Brenier's map as a Vector Quantile 92
4.2 Conditional Vector Quantiles 94
4.3 Vector Quantile Regression 94
5 Numerical Vector Quantile Regression 96
5.1 Regularized Vector Quantile Regression 96
5.2 Numerical Resolution 99
5.3 Numerical results 99
6 Statistical Analysis 101
6.1 Regularity of Dual Potentials 101
6.2 Law of Large Numbers 106
6.3 Central Limit Theorem 107

7 Conclusion 108
1 Summary of salient features 108
2 Perspectives for Future Work 109

8 Bibliography 112

Remerciements

Je tiens à remercier mes directeurs de thèse, Gabriel Peyré et Marco Cuturi, de m'avoir accompagnée et soutenue durant ces trois années, ainsi que mes co-auteurs Michèle Sebag, Guillaume Carlier et Alfred Galichon, qui m'ont aidée à cheminer avec patience tout en élargissant mes centres d'intérêt.

Je suis très reconnaissante aux rapporteurs, Jean-Michel Loubes et Yaron Lipman, ainsi qu'aux membres du jury, Jérémie Bigot et Rémi Flamary, de m'avoir fait l'honneur de s'intéresser à mon travail.

Merci à tous ceux dont la présence a adouci pour moi ces trois années, que ce soit à l'ENS, à l'INRIA, au CEREMADE, au CREST ou au LRI, en particulier Aude, Guillaume, Léa, Mélanie, Théo, Simon, Maxime, Luca, Andrea, Irène, Arnaud, Laurent, Giulia, Badr, Hicham et Heri.

Merci à mes proches dont le soutien a été déterminant, en particulier mes amis, mes parents, Corine et Karim, Ghyslaine, Pierre, et en particulier Claire, ma plus grande supportrice.

Introduction

Dealing with probability measures is a crucial challenge in machine learning, whether it be in supervised and unsupervised settings. While learning with underlying probability measures has been considered in different fields, deep learning architectures do not offer obvious tools to address learning from distributions. Probability distributions have been however the object of numerous innovations in the field of optimal transport, from which we take inspiration to propose new computational methods dealing specifically with probability measures. We show that learning from probability distributions can be eased thanks to dedicated models compatible with entropy-regularized optimal transport. In settings ranging from neural networks to statistical applications, we show that representing objects of interest as probability distributions comes with several computational and theoretical advantages. We introduce a general pipeline to support measures in neural architectures, that is able to cope with desired invariance properties, and propose to ease the computation of quantile regression in the multivariate case using the entropy.

1. Background

Statistics and Machine learning over the space of distributions. The application of machine learning techniques to a wide variety of tasks and settings has been shedding light upon its predictive power as well as its limitations. In *supervised* learning, the input dataset is composed of labelled examples $(\boldsymbol{X_i}, \boldsymbol{Y_i})_{i=1}^n$, where the observation $\boldsymbol{X_i}$ belongs to a feature space $\mathcal{X}$ (for instance, an image with varying pixel intensities), and the label $\boldsymbol{Y_i}$ encodes a target value (eg, whether it represents a dog). Within supervised learning, classification intends to learn from categorical labels, a classification rule f such that $\boldsymbol{Y_i} \approx f(\boldsymbol{X_i})$ in a certain sense, so that the class of a new input $\boldsymbol{X}$ can be predicted and the rule generalizes well. Similarly, regression tasks aim at predicting data characteristics $\boldsymbol{Y_i}$ using continuous labels. Its most common instance lies in linear regression, where f is affine. While the latter consists in assessing the conditional mean of a response variable Y to a set of predictors X, quantile regression goes beyond that by allowing analysis of the response at any quantile of its distribution. On the other hand, learning from unlabelled data $(\boldsymbol{X_i})_{i=1}^n$ in a data space $\mathcal{X}$ is called *unsupervised* learning. Procedures of interest include density fitting, which

corresponds to parameterizing the underlying, unknown distribution of the data with a parametric distribution. The field has sparked interest recently notably through generative models [Goodfellow et al., 2014, Kingma and Welling, 2014], that enable to generate other examples resembling the input data through dimensionality reduction.

The success of such procedures relies heavily on the nature of the instance space $\mathcal{X}$, as well as the metrics used to perform learning. For instance, it is worth noting that a lot of data types can be represented as discrete probability distributions, namely of the form $\boldsymbol{X} = \sum_{i=1}^{p} \mu_i \delta_{x_i} \in \mathcal{P}(\Omega)$, where x_i belongs to another space Ω, the weights $\mu_i \geqslant 0$ are such that $\sum_{i=1}^{p} \mu_i = 1$ and $\mathcal{P}(\Omega)$ stands for the space of probability measures with ground space Ω. This representation is naturally invariant in the ordering of the ground instances $(x_i)_{i=1}^{p}$. Such objects can also be seen as random vectors X distributed according to $\mu = (\mu_i)_{i=1}^{p}$, which is written $X \sim \mu$. Consequently, we alternatively denote them as objects belonging to $\mathcal{P}(\Omega)$ or to $\mathcal{R}(\Omega)$, the space of random vectors with ground space Ω. As such, we consider two random vectors having the same distribution as equivalent and indistinguishable. Their introduction to model for instance bags-of-images [Rubner et al., 2000] or color transfer [Pitié et al., 2007] in computer vision, shape registration in computer graphics [Solomon et al., 2015], bags-of-words in natural language processing [Kusner et al., 2015], to scan variations in neuroimaging [Gramfort et al., 2015], among other fields, still stimulates interest to this day. In fact, datasets themselves can be considered as input instances $\boldsymbol{X} \in \mathcal{X}$. In this setting, the goal of uncovering the best-performing algorithm for a task at hand has fuelled research for more than four decades [Rice, 1976], in the name of automated machine learning, referred to as auto-ML [Hutter et al., 2018]. Task-dependence as well as computational challenges linked to high dimensionality are intended to be alleviated notably by the design of expressive summaries of datasets called meta-features [Brazdil et al., 2008]. All in all, while these recent works have seen a more persistant resort to probability measures within their frameworks, the lack of unifying pipeline composed of adapted operations on raw probabilities is still a major bottleneck to the wide spread of this class of methods. In this thesis, we tackle this problem by proposing a general framework to process raw probability measures in neural networks, as both inputs and outputs. Section 2 below highlights these contributions, which are detailed in Chapter 1 – Section 4.

Invariant architectures. Best-performing models are expected to take into account the structure of the instance space Ω, its regularity as well as

the desirably recovered properties. Among them, symmetries and invariances play a major role in coping with input variabilities linked to their high dimensionality. Namely, a function f from an instance space Ω is said to be invariant under the action of a group G if

$$\forall x \in \Omega, g \in G, f(g \cdot x) = f(x)$$

An $f : \Omega \to \Omega$ is said to be G-equivariant (or G-covariant) if

$$\forall x \in \Omega, g \in G, f(g \cdot x) = g \cdot f(x)$$

Neural networks have long been designed to satisfy such invariance properties [Shawe-Taylor, 1993], such as original convolutional networks [LeCun et al., 1989, Krizhevsky et al., 2012] or wavelet scattering networks [Bruna and Mallat, 2013] for images. More recently, the necessity to deal with broader input types such as point clouds [Zaheer et al., 2017, Qi et al., 2017a, Hartford et al., 2018] or sequences [Vaswani et al., 2017, Lee et al., 2019, Murphy et al., 2019] spurred renewed interest on invariant and equivariant architectures. Initially designed to extend classical convolutional networks [Scarselli et al., 2009, Bruna et al., 2014, Defferrard et al., 2016], graph neural networks now also support invariance or equivariance properties with respect to the whole permutation group [Kondor and Trivedi, 2018, Maron et al., 2019a, Keriven and Peyré, 2019]. General treatment of symmetries in the case of finite subgroups of the symmetric group have been investigated [Ravanbakhsh et al., 2017] as well as in the infinite case [Kondor et al., 2018, Cohen and Welling, 2016, Weiler et al., 2018]). Quantifying their expressive power through universal approximation is to this day an active field of research [Maron et al., 2019a, Xu et al., 2019, Keriven and Peyré, 2019]. Despite these recent advances largely focused on point sets and graphs, the issue of dealing with invariant architectures processing probability measure inputs is still a major bottleneck in the field. In this thesis, we introduce a framework that performs regression on probability measure inputs, with customized invariance requirements, and illustrate its applicative relevance in the context of automated machine learning. Section 2 below highlights these contributions, which are detailed in Chapter 2 – Section 5.

Optimal transport methods in learning. Learning from probability distributions requires adapted metrics expressing meaningful notions of proximities. Among them, φ-divergences [Csiszar, 1975] have been widely used thanks to their computational simplicity, but suffer from the drawback of

not metrizing weak convergence. Therefore, other metrics such as Maximum Mean Discrepencies [Gretton et al., 2007], Optimal Transport (OT) [Kantorovich, 1942, Villani, 2008] or related Sinkhorn divergences [Genevay et al., 2018, Feydy et al., 2019] have been put in the spotlight. The in-depth study of transport maps [Santambrogio, 2015, Villani, 2008], such as their representation as gradients of convex functions [Ryff, 1970, Brenier, 1991, McCann, 1995] allows for a generalization of monotone functions in higher dimension, which makes them a good candidate for statistical applications such as quantile regression [Carlier et al., 2016b, 2017]. The original OT formulation consists of a linear program [Kantorovich, 1942] that writes, for a ground cost $c : \Omega \times \Omega \to \mathbb{R}_+$,

$$\min_{(X,Y)\sim\pi\in\Pi(\alpha,\beta)} \mathbb{E}_\pi \left[c(X,Y)\right]$$

where the minimum is taken over $\Pi(\alpha, \beta)$, defined as the set of all transport plans π with fixed marginal distributions $\alpha \in \mathcal{P}(\Omega)$ and $\beta \in \mathcal{P}(\Omega)$, which reads

$$\Pi(\alpha,\beta) \stackrel{\text{def.}}{=} \{\pi \in \mathcal{P}(\Omega^2), \forall (A,B) \subset \Omega^2, \pi(A\times\Omega) = \alpha(A), \pi(\Omega\times B) = \beta(B)\}$$

This problem can be solved using the network simplex or interior-point methods, with a complexity of at most $O(n^3 \log(n))$ (see for instance [Goldberg and Tarjan, 1989]) for two discrete distributions of size n. In the case of two equal uniform discrete marginal distributions, also known as *linear assignment problem*, the optimal π is a permutation matrix [Bertsimas and Tsitsiklis, 1997], and the exact problem can be solved using the early Hungarian algorithm [Borchardt and Jacobi, 1865] or the auction algorithm [Bertsekas, 1981] and their variants. A typical choice lies in $c = d$, with d a distance on Ω, in which case the minimum yields the 1-Wasserstein distance, denoted W_1:

$$\mathrm{W}_1(\alpha,\beta) \stackrel{\text{def.}}{=} \min_{\pi\in\Pi(\alpha,\beta)} \int_{\Omega^2} \|x-y\| \mathrm{d}\pi(x,y)$$

which is known to be a norm and to metrize weak convergence (see [Santambrogio, 2015], Proposition 5.1 and Theorem 5.11). Approximate computations have eased its application to high dimensional problems [Levy and Schwindt, 2018, Peyré and Cuturi, 2019]. Strong regularizers such as the entropy [Wilson, 1969, Erlander and Stewart, 1990, Cuturi, 2013] have long been considered to force the solution to have a spread non-sparse support, which stabilizes the computation while ensuring the objective is strongly convex. In practice, Sinkhorn's algorithm [Cuturi, 2013] enables fast parallelizable

computations to solve the ε-regularized counterpart of the original problem, namely

$$\min_{(X,Y)\sim\pi\in\Pi(\alpha,\beta)} \mathbb{E}_\pi\left[c(X,Y)\right] - \varepsilon\mathcal{E}(\pi|\alpha\otimes\beta)$$

where $\mathcal{E}(\pi|\alpha\otimes\beta)$ stands for the relative entropy of the joint coupling π with respect to the product measure $\alpha\otimes\beta$, which reads

$$\mathcal{E}(\pi) \stackrel{\text{def.}}{=} \int_{\Omega\times\Omega} \log\Big(\frac{\mathrm{d}\pi(x,y)}{\mathrm{d}\alpha(x)\mathrm{d}\beta(y)}\Big)\mathrm{d}\pi(x,y)$$

This formulation is known to be a near-linear time approximation of the original problem [Altschuler et al., 2017], and can be extended to benefit from stochastic optimization [Genevay et al., 2016], acceleration techniques [Altschuler et al., 2017, Scieur et al., 2016, Dvurechensky et al., 2018], improved complexity on gridded spaces using convolutions [Solomon et al., 2015], multi-scale approaches [Schmitzer, 2016], online settings [Mensch and Peyré, 2020], and to cope with multi-marginal problems [Benamou et al., 2015], as well as unbalanced transport [Chizat, 2017]. OT is particularly appreciated for its ability to leverage the underlying geometry of the data, which can be strengthened by enforcing structure constraints [Alvarez-Melis et al., 2018]. Asymptotic behavior of empirical Wasserstein distances has been extensively studied over the last decades, see for instance [del Barrio et al., 1999, Del Barrio et al., 2005, del Barrio and Loubes, 2019, Rippl et al., 2016] and recently extended to regularized distances [Bigot et al., 2019a, Klatt et al., 2020]. Though OT is known to suffer from the curse of dimensionality [Dudley, 1969, Weed and Bach, 2019], regularized counterparts benefit from better sample complexities [Genevay et al., 2019, Mena and Niles-Weed, 2019]. Closely related variational problems include Wasserstein gradient flows [Jordan et al., 1998, Ambrosio et al., 2008] and Wasserstein barycenters [Agueh and Carlier, 2011, Le Gouic and Loubes, 2016, Bigot and Klein, 2015], for which algorithmic adaptations have been proposed [Cuturi and Doucet, 2014]. OT has been extended as the Gromov-Wasserstein distance [Mémoli, 2011] to cope with probability measures that do not share a common space. Though conditioned by a non-convex quadratic program, numerical frameworks based on conditional gradient [Flamary and Courty, 2017] or entropic regularization have been proposed [Peyré et al., 2016], and its interpolation properties have been highlighted [Vayer et al., 2020]. All these theoretical and computational aspects have broadened the applicative settings of optimal transport, even beyond the aforementioned fields to astrophysics, for modeling the early universe [Frisch et al., 2002], music transcription [Flamary et al., 2016], genomics [Evans and Matsen,

2012], statistical learning, to assess the convergence of various algorithms [Canas and Rosasco, 2012], fluid dynamics [Gallouët and Mérigot, 2018], economics, for matching markets modeling [Dupuy and Galichon, 2014] or fairness [Gordaliza et al., 2019].

Quantile Regression. First introduced in the early 19^{th} century by Legendre [Legendre, 1805] and Gauss [Gauss, 1809] to model the shape of the earth and movements of celestial bodies, the use of least squares still gathers interest to this day to estimate conditional means, due to their computational ease and optimality under normal errors [Gauss, 1822]. However, Edgeworth pointed out the median as a preferable alternative to the mean, particularly in the case of Gaussian mixtures [Edgeworth, 1888]. The ability to consider other quantiles of the response variable was pioneered in [Koenker and Bassett, 1978], that estimate the t-quantile ($t \in [0;1]$) of variable $\varepsilon = Y - q_t(x)$ conditional to $X = x$ by minimizing the loss function $\mathbb{E}[t\varepsilon^+ + (1-t)\varepsilon^- | X]$, where ε^+ and ε^- respectively refer to the positive and negative parts of ε. It is common practice to stipulate a linear form of the quantiles $q_t(x) = \beta_t^\top x + \alpha_t$, in which case the problem boils down to solving

$$\min_{\alpha_t, \beta_t} \mathbb{E}\left[\left(Y - \beta_t^\top X - \alpha_t\right)^+ + (1-t)\left(\beta_t^\top X + \alpha_t\right)\right]$$

Strong incentives to analyze conditional distributions at arbitrary quantiles include a range of applicative settings, from healthcare [Koenker and Hallock, 2001, Austin et al., 2005, Azagba and Sharaf, 2012], bioinformatics [Song et al., 2017], education [Eide and Showalter, 1998], finance [Zietz et al., 2008], ecology [Cade and Noon, 2003] to reduction of inequalities [Chamberlain, 1994, Buchinsky, 1994, 1998, Melly, 2005]. For instance, [Koenker and Hallock, 2001] apply quantile regression to the case of infant birthweight, showing that offering prenatal care has much larger impact on the lower quantiles of the distribution. [Chamberlain, 1994, Buchinsky, 1994] have considered the technique to leverage the impact of union status and education on wage inequalities, showing for instance that union status has a much larger effect on lower quantiles of the wage distribution. [Azagba and Sharaf, 2012] have shown that increasing the intake of fruits and vegetables is more effective to mitigate the risk of obesity at the higher quantiles of the body mass index. Quantile regression coefficients can be interpreted as estimators for treatment effects given a control population [Lehmann, 1974, Doksum, 1974], which extends to the case of p different treatments [Koenker, 2005]. There is, to this day, no consensus on how to extend quantile regression to the case of a multivariate response. Among other proposals [Chaudhuri, 1996, Koltchinskii,

1997, Serfling, 2004, Hallin et al., 2010, Belloni and Winkler, 2011, Kong and Mizera, 2012], [Carlier et al., 2016b, 2017] introduce a notion of multivariate quantile based on optimal transport. They define the conditional quantile of $Y|X = x$ as the Brenier's map between a fixed distribution (for instance, multivariate uniform on a cube) and the law of $Y|X = x$. Thanks to polar factorization [Ryff, 1970, Brenier, 1991, McCann, 1995], this (multivariate) quantile function is known to be the gradient of a convex function, extending the notion of monotonicity to the multivariate case, and allowing to retrieve the whole monotone function at once, as opposed to the original "t by t" approach. In practice, this problem is solved by correlation maximization under an additional mean-independence constraint, namely

$$\max_{(U,X,Y)\sim\pi} \mathbb{E}_\pi \left[U^\top Y\right] \quad \text{s.t. } U \sim \mathcal{U}\left([0,1]^d\right),\ (X,Y) \sim \nu,\ \mathbb{E}\left[X|U\right] = 0$$

As hinted at above, practical computations of such multivariate quantiles is still a major bottleneck to the wide spread of this method, that relies on linear programming. In this thesis, we propose to widen its use by considering a regularized version of the problem. Section 2 below highlights these contributions, which are detailed in Chapter 3 – Section 6.

We now present in more technical details our original contributions, from both the theoretical and empirical standpoints.

2. Summary of Contributions

Chapter 1: Stochastic Deep Networks

This chapter provides a unifying framework to process discrete measures in neural architectures, backed by theoretical and empirical contributions.

Previous works. While initially tailored for images [Krizhevsky et al., 2012] and speech [Hinton et al., 2012], deep neural networks have been designed to support increasingly complex structured data types, such as shapes [Wu et al., 2015b], sounds [Lee et al., 2009], texts [Lecun et al., 1998], graphs [Henaff et al., 2015]. Such architectures rely on the composition of elementary operations handling vectors that stream well on GPUs, and that can be automatically differentiated using back-propagation. Their extension to *sequences* of vectors had enormous impact [Hochreiter and Schmidhuber, 1997]. More recently, learning from unordered samples has drawn attention since the seminal works of [Ravanbakhsh et al., 2016, Zaheer et al., 2017, Qi et al., 2017a] that design neural architectures tailored for point set inputs.

In this light, architectures generating point clouds have been developed [Fan et al., 2017, Achlioptas et al., 2018, Yi et al., 2019]. Discussions on their limitations have also emerged [Wagstaff et al., 2019, Segol and Lipman, 2020]. Previous works were also aware of the importance of order, and manage to handle sequences recursively with attention mechanisms [Vinyals et al., 2016, 2015], which has paved the way for a stream of follow-up works in the field of natural language processing [Vaswani et al., 2017, Lee et al., 2019]. Similar ideas can be found in point process models, which allow for the analysis of counting measures or random sets. Poisson [Rajaram et al., 2005] and Hawkes processes [Belanger et al., 2018, Mei and Eisner, 2017] are among the most popular models that offer basis for deep parameterization [Xiao et al., 2017a, Du et al., 2016, Mei and Eisner, 2017, Xiao et al., 2017b], mostly using likelihood-based approaches [Belanger et al., 2018, Du et al., 2016, Mei and Eisner, 2017].

All in all, while learning with underlying probability measures has been considered in different fields [Muandet et al., 2012, Poczos et al., 2013, Pevny and Kovarik, 2019], providing a unifying deep learning framework supporting raw probabilities in accordance with the convergence in law is, to the best of our knowledge, a new concept. Various applicative settings create a strong incentive for devising probability distribution-based neural networks. In computer vision for instance, as opposed to embedding the inputs on a grid, representing 3D objects as probability measures alleviates the computational burden and helps preserve topological structure as well as natural invariances. Moreover, in fields ranging from physics [Godin et al., 2007], biology [Grover et al., 2011], ecology [Tereshko, 2000] to census data [Guckenheimer et al., 1977], encoding populations at a macroscopic level with probability measures, without requiring to monitor individual trajectories and regardless of the population size, eases the pressure of experimental costs or privacy concerns.

Though analogies can be seen between discrete uniform probability measures and point clouds, as architectures thereof are both expected to be permutation invariant, often equivariant to geometric transformations (translations, rotations) and capture local structure of points [Chen et al., 2014, Cheng et al., 2016, Guttenberg et al., 2016], their natural topologies differ. In sharp contrast with architectures dealing with point clouds that use the Hausdorff distance, we resort to the convergence in law, also known as the weak-* convergence of measures, that is metrized by the Wasserstein distance. As such, some architectures continuous for the Hausdorff distance are not continuous for the convergence in law, for instance due to max pooling steps

[Qi et al., 2017a]. Optimal transport (OT) has recently been growing in popularity in machine learning, notably due to its approximate computations obtained with strongly convex regularizers such as the entropy [Cuturi, 2013], eligible for fast parallelizations. The advantages of this regularization provided the bases for the use of OT in various applicative settings [Courty et al., 2017, Rolet et al., 2016, Huang et al., 2016]. Although Wasserstein metrics have long been taken into consideration for inference purposes [Bassetti et al., 2006], their introduction in deep learning architectures is somewhat recent, whether it be for generative tasks [Bernton et al., 2017, Arjovsky et al., 2017, Genevay et al., 2018] or regression purposes [Frogner et al., 2015, Hashimoto et al., 2016].

Contributions. The purpose of this work is to propose an extension of these approaches through a uniting framework that enables to process probability measures directly in deep architectures, regardless of the task considered.

(i) **Learning from probability measures:** we introduce a general pipeline to process *probability measures* or *random vectors* as inputs to both supervised and unsupervised machine learning tasks, which relies numerically on the Lagrangian representation of measures (obtained by sampling). Parameterized by interaction functionals $f : \mathbb{R}^q \times \mathbb{R}^q \to \mathbb{R}^r$, our original layers map $\mathbb{R}^q$-supported random vectors (denoted $X \in \mathcal{R}(\mathbb{R}^q)$) to $\mathbb{R}^r$-supported counterparts, in the following way

$$T_f : X \in \mathcal{R}(\mathbb{R}^q) \mapsto \mathbb{E}_{X' \sim X}[f(X, X')] \in \mathcal{R}(\mathbb{R}^r) \tag{1}$$

where X' is an independent copy of X, that has the same law. Maps (1) are also characterized by a natural invariance in the ordering of the data observations. Resulting architectures are designed as iterative transformations of random vectors using such layers, namely

$$X \in \mathcal{R}(\mathbb{R}^{q_0}) \mapsto Y = T_{f_T} \circ \cdots \circ T_{f_1}(X) \in \mathcal{R}(\mathbb{R}^{q_T}) \tag{2}$$

where $f_t : \mathbb{R}^{q_{t-1}} \times \mathbb{R}^{q_{t-1}} \to \mathbb{R}^{q_t}$. Such networks are versatile enough to (i) map measures to measures; and (ii) bridge the gap between measures and Euclidean spaces (with deterministic outputs). They are thus suited to the wide variety of machine learning applications.

(ii) **Robustness and Universal Approximation:** on the theoretical side, these architectures are granted Lipschitz robustness in the sense

of the Wasserstein-1 distance

$$\forall\,(X,Y)\in\mathcal{R}(\mathbb{R}^q)^2,\ \mathrm{W}_1(T_f(X),T_f(Y))\leqslant 2rC(f)\,\mathrm{W}_1(X,Y) \qquad (3)$$

as long as the interaction functional f is $C(f)$-Lipschitz itself in its individual variables. They also inherit from the universal approximation capability of neural networks:

Theorem 1. *Let $\mathcal{F}:\mathcal{R}(\Omega)\to\mathcal{R}(\Omega')$ be a continuous map for the convergence in law, where $\Omega\subset\mathbb{R}^q$ and $\Omega'\subset\mathbb{R}^r$ are compact. Then $\forall\eta>0$ there exists three continuous maps f,g,h such that*

$$\forall X\in\mathcal{R}(\Omega),\quad \mathrm{W}_1(\mathcal{F}(X),T_h\circ\Lambda\circ T_g\circ T_f(X))\leqslant\eta. \qquad (4)$$

where $\Lambda:X\mapsto(X,U)$ concatenates a uniformly distributed random vector U.

(iii) **Empirical illustrations:** on the applicative side, we provide instanciations of such networks and show their versatility on a set of both supervised and unsupervised applications, namely classification, prediction and generative networks (see for instance Figure 1 for examples of generated 2D measures).

These contributions have been published in [De Bie et al., 2019].

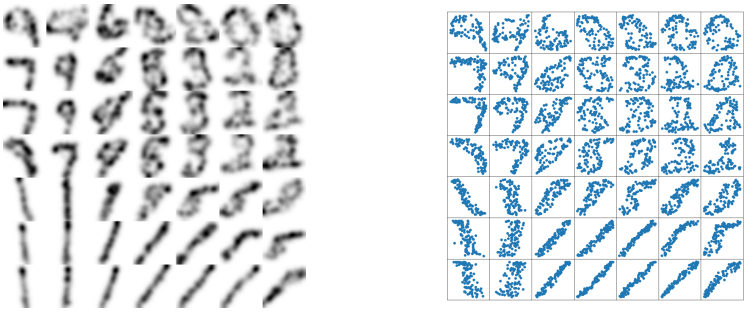


Figure 1: Examples of generated measures in 2D, learnt from the MNIST dataset as discrete measures. Blue dots stand for corresponding Diracs' positions.

Chapter 2: Distribution-Based Invariant Deep Networks for Automated Machine-Learning

This chapter provides theoretical and empirical grounds to perform regression using neural networks on discrete measure inputs, with customized invariance requirements.

Previous works. Learning from samples with a neural architecture compliant with domain- and application-dependent **invariance or equivariance** properties ensures a more robust model, better capturing the data geometry. Neural architectures benefiting from such properties have been pioneered by [Ravanbakhsh et al., 2016, Zaheer et al., 2017, Qi et al., 2017a] in the case of point sets subject to invariance or equivariance, including some works with a particular focus on dataset inputs [Edwards and Storkey, 2017], which have been extended to permutation equivariance across sets [Hartford et al., 2018]. Similar ideas can be found in attention-based mechanisms for sequences [Vaswani et al., 2017, Lee et al., 2019, Murphy et al., 2019]. In the same vain, invariant and equivariant architectures have been expanded to support graphs [Herzig et al., 2018, Kondor et al., 2018, Maron et al., 2019a, Chen et al., 2019, Albooyeh et al., 2020]. Characterizations of invariance or equivariance under group actions have been proposed in the finite [Gens and Domingos, 2014, Cohen and Welling, 2016, Ravanbakhsh et al., 2017] or infinite case [Wood and Shawe-Taylor, 1996, Kondor and Trivedi, 2018]. A general characterization of linear layers on the top of a representation that are invariant or equivariant with respect to the whole permutation group has been proposed by [Maron et al., 2019a, Keriven and Peyré, 2019]. Expressive power of the proposed networks through universality results are known to hold in the case of sets [Zaheer et al., 2017], point clouds [Qi et al., 2017a], equivariant point clouds [Segol and Lipman, 2020], discrete measures [De Bie et al., 2019], invariant [Maron et al., 2019b] and equivariant [Keriven and Peyré, 2019] graph neural networks. Closest to our work, [Maron et al., 2020] devises a neural architecture invariant with respect to the ordering of samples and their features. The originality of our approach is that we do not fix in advance the number of samples, and consider probability distributions instead of point clouds.

In this work, distribution-based neural architectures [De Bie et al., 2019] are extended to cope with an **additional invariance in the features and labels**, namely, the space supporting the distribution. This extra invariance is required to tackle the long-known **Auto-ML problem** (short for *automated machine learning*) [Rice, 1976, Feurer et al., 2015, Hutter et al., 2018], which aims to identify *a priori* the machine learning (ML) configuration best suited to a dataset, in the sense of a given performance indicator (that entails both the learning algorithm and the hyperparameters thereof). The auto-ML rationale falls within the so-called democratization of machine learning [Hutter et al., 2018]. However, as major bottlenecks towards that goal, the absence of a learning algorithm dominating other algorithms

on all datasets [Wolpert, 1996], together with the combinatorial structure of the search space make the auto-ML problem particularly arduous.

The ability to characterize a dataset by a set of relevant features, referred to as *meta-features* allows for solving the auto-ML problem through another supervised learning problem: given archives recording the performance of several ML algorithms on various datasets [Vanschoren et al., 2013], each dataset being described as a vector of meta-features, the best-performing algorithm (among these configurations) on a new dataset could be predicted from its meta-features. These meta-features are expected to be expressive summaries of input datasets, that preserve dataset similarities and are rather inexpensive to compute. Particular meta-features have been introduced, whether it be hand-crafted statistics [Feurer et al., 2015, Muñoz et al., 2018] or given by the performance of fast learning algorithms [Pfahringer et al., 2000]. Closest to our work, DATASET2VEC [Jomaa et al., 2019] extracts meta-features from point-set-represented datasets, through the classification task of identifying whether sub-samples of datasets are extracted from the same distribution. In sharp contrast, we advocate for the distribution representation of datasets endowed with the topology of the convergence in law. Other, though less related approaches consist in learning a generic model with quick adaptability to new tasks [Finn et al., 2018, Yoon et al., 2018, Perrone et al., 2018]).

Contributions. In this chapter, we advocate for the measure representation of datasets while offering theoretical and empirical grounds to design dataset meta-features by performing regression with customized invariance requirements.

(i) **Distribution-based Invariant Regression:** we design neural architectures achieving regression with customized invariance requirements, referred to as *invariant regression*, with probability measure inputs, where the natural invariance in the ordering of the instances is complemented by invariances in the ordering of the data features. Our motivating application is the design of dataset meta-features in automated machine learning, where inputs are *datasets* composed of both (d_X-sized) data instances and (d_Y-sized) meta-labels. Interaction functionals (1 are then required to satisfy the invariance property

$$\forall \sigma \in S_{d_X} \times S_{d_Y}, \forall (x,y) \in (\mathbb{R}^{d_X+d_Y})^2, \varphi(\sigma(x),\sigma(y)) = \varphi(x,y) \quad (5)$$

where S_d denotes the d-sized permutation group, and $S_{d_X} \times S_{d_Y}$ acts on $\mathbb{R}^{d_X+d_Y}$ as: for $x \in \mathbb{R}^{d_X}, y \in \mathbb{R}^{d_Y}$ and $\sigma = (\sigma_X, \sigma_Y) \in S_{d_X} \times$

$S_{d_Y}, \sigma(x,y) = [(x_{\sigma_X^{-1}(i)})_{i=1\dots d_X}; (y_{\sigma_Y^{-1}(j)})_{j=1\dots d_Y}] \in \mathbb{R}^{d_X+d_Y}$. The first layer of an invariant architecture of the form (2) is then required to be invariant for the whole network to be. In this setting, quantitative analysis is performed using the permutation-invariant Wasserstein-1 distance, namely, for two Ω-supported probability measures α, β (denoted $\alpha, \beta \in \mathcal{M}_1^+(\Omega)$)

$$\overline{\mathrm{W}}_1(\alpha,\beta) = \min_{\sigma \in S_{d_X} \times S_{d_Y}} \mathrm{W}_1(\sigma_\sharp \alpha, \beta) \tag{6}$$

where σ still denotes (for simplicity) the push-forward operator between $\alpha \in \mathcal{M}_1^+(\Omega)$ and $\sigma_\sharp \alpha \in \mathcal{M}_1^+(\Omega)$, which are considered indistinguishable.

(ii) **Robustness and Universal Approximation:** such architectures inherit from the Lipschitz property (3) as well as robustness with respect to small deformations, in the permutation-invariant Wasserstein-1 sense:

Proposition 1. *For $\tau : \mathbb{R}^d \to \mathbb{R}^d$ and $\xi : \mathbb{R}^r \to \mathbb{R}^r$ two Lipschitz maps, one has, for all $\alpha, \beta \in \mathcal{M}_1^+(\Omega)$,*

$$\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha)) \leqslant \sup_{x \in f_\varphi(\tau(\Omega))} \|\xi(x) - x\|_2 + 2r \operatorname{Lip}(\varphi) \sup_{x \in \Omega} \|\tau(x) - x\|_2$$

Also, if τ is equivariant, the following holds:

$$\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) \leqslant 2r \operatorname{Lip}(\varphi) \operatorname{Lip}(\tau) \operatorname{Lip}(\xi) \overline{\mathrm{W}}_1(\alpha,\beta)$$

Such architectures are also granted universal approximation capabilities:

Theorem 2. *Let $\mathcal{F} : \mathcal{M}_1^+(\Omega) \to \mathbb{R}$ a $S_{d_X} \times S_{d_Y}$-invariant map continuous for the convergence in law, where Ω is compact. Then $\forall \eta > 0$, there exists two continuous maps ψ, φ such that*

$$\forall \alpha \in \mathcal{M}_1^+(\Omega), \quad |\mathcal{F}(\alpha) - \psi \circ T_\varphi(\alpha)| < \eta$$

where φ is $S_{d_X} \times S_{d_Y}$-invariant and independent of $\mathcal{F}$.

(iii) **Empirical illustrations:** we demonstrate the validity of the proposed architectures in the context of automated machine learning, to design dataset meta-features conditional to various meta-tasks, from distribution identification to performance model learning. The meta-features

designed as such outperform previous approaches, whether it be handcrafted meta-features designed in the past two decades [Feurer et al., 2015, Muñoz et al., 2018] or their recent learnt counterparts [Jomaa et al., 2019, Maron et al., 2020].

These contributions have been published in [De Bie et al., 2020].

Chapter 3: Regularized Vector Quantile Regression

This chapter provides a novel scalable numerical framework to perform Vector Quantile Regression (VQR) based on entropic regularization, complemented by statistical asymptotics analysis.

Previous works. Quantile regression, introduced by the seminal work of Koenker and Bassett (1978) [Koenker and Bassett, 1978], has become a popular tool to analyze the whole distribution of a response variable Y to a set of predictors X. It goes beyond classical median regression by allowing regression at any quantile $t \in [0;1]$ of the distribution. Originally, the t-quantile of variable $\varepsilon = Y - q_t(x)$ conditional to $X = x$ is estimated by minimizing the loss function $\mathbb{E}\left[t\varepsilon^+ + (1-t)\,\varepsilon^- | X\right]$, where ε^+ and ε^- respectively refer to the positive and negative parts of ε. Stipulating a linear form of the quantiles $q_t(x) = \beta_t^\top x + \alpha_t$, and without loss of generality that $\mathbb{E}[X] = 0$, the problem boils down to solving

$$\min_{\alpha_t, \beta_t} \mathbb{E}\left[\left(Y - \beta_t^\top X - \alpha_t\right)^+ + (1-t)\,\alpha_t\right] \tag{7}$$

whose dual formulation is known to be [Koenker and Bassett, 1978]

$$\max_{V_t} \mathbb{E}\left[V_t Y\right], \quad V_t \in [0;1], \quad \mathbb{E}\left[X V_t\right] = 0, \quad \mathbb{E}\left[V_t\right] = (1-t) \tag{8}$$

As known since its original introduction [Koenker and Bassett, 1978], this problem has a linear programming formulation. Associated with mild assumptions, complementary slackness leads to writing

$$V_t = \mathbb{1}\{Y > \alpha_t + \beta_t^\top X\} \tag{9}$$

which turns constraints of (8) into

$$\begin{gathered} \mathbb{E}\left[\mathbb{1}\{Y > \alpha_t + \beta_t^\top X\}\right] = \mathbb{P}(Y > \alpha_t + \beta_t^\top X) = (1-t) \\ \mathbb{E}\left[X \mathbb{1}\{Y > \alpha_t + \beta_t^\top X\}\right] = 0 \end{gathered} \tag{10}$$

Strong incentives for designing a multivariate counterpart of (7) include capturing joint dependencies in the response variables, given the predictors, as well as recovering the whole *monotone* quantile function at once, as opposed to the "t by t" approach.

[Carlier et al., 2016b, 2017] have proposed a multivariate extension of quantile regression based on optimal transport. They have shown [Carlier et al., 2016b] that imposing a monotonicity constraint on the quantile curves

$$t \mapsto \alpha_t + \beta_t^\top X \text{ increasing on } [0;1] \tag{11}$$

and defining $U = \int_0^1 V_t \mathrm{d}t$ turns constraints (10) into

$$\begin{array}{l} U \text{ is uniformly distributed over } [0;1], \text{ denoted } U \sim \mathcal{U}\left([0,1]\right) \\ X \text{ is mean-independent from } U, \text{ namely } \mathbb{E}\left[X|U\right] = \mathbb{E}\left[X\right] = 0 \end{array} \tag{12}$$

Therefore, they consider as natural prolongation (see [Carlier et al., 2016b], Theorem 3.3) the extension of the Monge-Kantorovich problem of optimal transport, with an additional constraint of mean-independence

$$\max_{(U,X,Y)\sim\pi} \mathbb{E}_\pi\left[UY\right] \quad \text{s.t. } U \sim \mathcal{U}\left([0,1]\right), \ (X,Y) \sim \nu, \ \mathbb{E}\left[X|U\right] = \mathbb{E}\left[X\right] = 0 \tag{13}$$

where ν is the (given) distribution of the data. As opposed to the "t by t" approach, this global approach is strongly related to polar factorization [Ryff, 1970, Brenier, 1991, McCann, 1995] in the sense that it allows for the strong representation

$$Y = Q_{Y|X}(U,X), \quad U|X \sim \mathcal{U}([0;1]) \tag{14}$$

where $u \mapsto Q_{Y|X}(u,X)$ is non-decreasing almost surely. Stipulating an affine form of the quantile, the Vector Quantile Regression (VQR) problem for a d-dimensional response variable Y, $d \geqslant 2$, is the multivariate analogous of (13)

$$\max_{(U,X,Y)\sim\pi} \mathbb{E}_\pi\left[U^\top Y\right] \quad \text{s.t. } U \sim \mathcal{U}\left([0,1]^d\right), \ (X,Y) \sim \nu, \ \mathbb{E}\left[X|U\right] = 0 \tag{15}$$

Similarly to (14), the strong representation holds, where the vector quantile of Y conditional to $X = x$ is then the Brenier's map between $\mathcal{U}([0;1]^d)$ and the law of $Y|X = x$, namely the gradient of a convex function.

In this context, the uniform U can be interpreted as a reference outcome for defining treatment effects [Carlier et al., 2016b], where the distribution of an outcome for the untreated population is then uniform; (14) as well as the equivalent objective for (15) $\mathbb{E}_\pi\left[\|Y-U\|^2\right]$ can also lead to interpret U

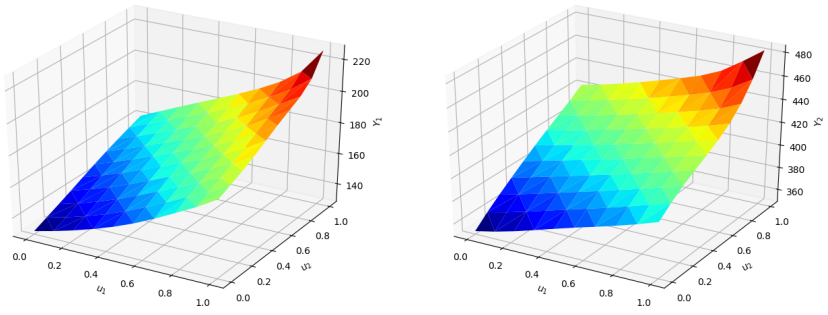
as non-linear latent factors [Carlier et al., 2016b], independent of predictors X, that best explain the variations in Y. The connection between U and a notion of continuous rank have also been highlighted (see for instance [Koenker, 2005], Chapter 3.5 or [Carlier et al., 2016b]).

Optimality conditions are characterized as well when specification cannot be taken for granted [Carlier et al., 2017], which provides an alternative strong representation in that case.

While other approaches for the multivariate extension of quantile regression have been proposed [Chaudhuri, 1996, Koltchinskii, 1997, Serfling, 2004, Hallin et al., 2010, Belloni and Winkler, 2011, Kong and Mizera, 2012], this work focuses on retrieving two desirable properties of quantiles in higher dimension, namely monotonicity and transport from a fixed distribution.

Previous methods for solving (15) rely on a vectorized version of the linear program (15) [Carlier et al., 2016b], yet the potential benefits of incorporating entropic regularization to this problem have been highlighted [Carlier et al., 2017].

Contributions. The main contributions of this chapter include a numerical framework to perform multivariate quantile regression as well as a statistical basis for hypothesis testing.



Y_1 =Weight Y_2 =Thigh Circumference

Figure 2: RVQR 2D quantiles obtained by regressing $Y = (Y_1, Y_2)$ on $X = (1,$ Height) for the small height individuals in the ANSUR II MALE sample.

(i) **Regularized Vector Quantile Regression:** we introduce the Regularized Vector Quantile Regression (RVQR) problem, whose primal

formulation minimizes an ε-regularized counterpart of (15), namely

$$\begin{aligned} &\max_{(U,X,Y)\sim\pi} \mathbb{E}_\pi\left[U^\top Y\right] - \varepsilon \mathbb{E}_\pi\left[\log \pi(U,X,Y)\right] \\ &\text{s.t. } U \sim \mathcal{U}\left([0,1]^d\right) \overset{\text{def.}}{=} \mu, \quad (X,Y)\sim\nu, \quad \mathbb{E}\left[X|U\right] = \mathbb{E}\left[X\right] \end{aligned} \tag{16}$$

whose discrete formulation reads

$$\begin{aligned} &\max_{\pi_{ij}\geqslant 0} \sum_{ij} \pi_{ij}\left(u_i^\top y_j\right) - \varepsilon \sum_{ij} \pi_{ij}\log\pi_{ij} \\ &\text{s.t. } \sum_j \pi_{ij} = \mu_i, \quad \sum_i \pi_{ij} = \nu_j, \quad \sum_j \pi_{ij} x_j = \sum_j \nu_j x_j \end{aligned} \tag{17}$$

Its dual is the (unconstrained) RVQR problem

$$\min_{\psi,b} \sum_j \psi_j \nu_j + \varepsilon \sum_i \mu_i \log\left[\sum_j \exp\frac{1}{\varepsilon}\left[u_i^\top y_j - b_i^\top x_j - \psi_j\right]\right] \tag{18}$$

which alternatively writes

$$\min_{\varphi,\psi,b} \sum_i \mu_i\varphi_i + \sum_j \psi_j\nu_j + \varepsilon \sum_{ij} \exp\left(\frac{1}{\varepsilon}[u_i^\top y_j - \varphi_i - b_i^\top x_j - \psi_j]\right).$$

Though initially conditioned by an additional mean-independence constraint, the RVQR problem (18) inherits from the regularity and scalability of entropy-regularized optimal transport [Cuturi, 2013, Peyré and Cuturi, 2019].

(ii) **Numerical Resolution:** we propose a numerical scheme to perform RVQR in practice, that relies on solving the dual formulation (18), which is a smooth and unconstrained problem, through accelerated [Nesterov, 1983] gradient descent, which gives optimal convergence rates for first-order methods. With empirical illustrations on real datasets (see for instance Figure 11 for examples of obtained 2D quantiles), we retrieve classical quantile regression [Koenker and Bassett, 1978] in the one-dimensional case, and show the computational advantages of regularization in higher dimension.

(iii) **Statistical Analysis:** we analyze statistical properties of the RVQR problem in the finite sample case, yielding a law of large numbers and a central limit theorem for the finite-dimensional dual potentials:

Proposition 2. *The normalized RVQR finite-sample dual potentials are asymptotically Gaussian.*

which paves the way for hypothesis testing on the RVQR regression coefficients.

Notations

Ambiant space. For two metric space $\mathcal{X}$ and $\mathcal{Y}$, we denote by

- $\mathcal{C}(\mathcal{X})$ the space of continuous real-valued functions on $\mathcal{X}$;
- $\mathcal{C}(\mathcal{X}, \mathcal{Y})$ the space of continuous $\mathcal{Y}$-valued functions on $\mathcal{X}$;
- $\mathcal{M}_1^+(\mathcal{X})$ the set of Radon probability (i.e. with unit mass) measures supported on $\mathcal{X}$;
- $\mathcal{R}(\mathcal{X})$ the set of random vectors supported on $\mathcal{X}$.

Measures. We use capitals to denote random vectors (for instance, X). For a given random vector $X \in \mathcal{R}(\mathcal{X})$, we denote $\alpha_X \in \mathcal{M}_1^+(\mathcal{X})$ its law, which writes $X \sim \alpha_X$. Two random vectors X, X' having the same law are also denoted $X \sim X'$ or alternatively $\alpha_X = \alpha_{X'}$. It satisfies for any continuous map $f \in \mathcal{C}(\mathcal{X}), \mathbb{E}(f(X)) = \int_{\mathcal{X}} f(x) \mathrm{d}\alpha_X(x)$. Its expectation is denoted $\mathbb{E}(X) = \int_{\mathcal{X}} x \mathrm{d}\alpha_X(x) \in \mathcal{X}$. In general, a compact support is denoted Ω. The Dirac measure at point x is δ_x. We denote $\alpha_n = \frac{1}{n} \sum_{i=1}^n \delta_{x_i}$ the empirical measure obtained from an i.i.d sample $(x_1, \ldots, x_n)$. Let $\alpha \in \mathcal{M}_1^+(\mathcal{X})$ and $\beta \in \mathcal{M}_1^+(\mathcal{Y})$, we define $\Pi(\alpha, \beta) \stackrel{\text{def.}}{=} \{\pi \in \mathcal{M}_1^+(\mathcal{X} \times \mathcal{Y}), \forall (A, B) \subset \mathcal{X} \times \mathcal{Y}, \pi(A \times \mathcal{Y}) = \alpha(A), \pi(\mathcal{X} \times B) = \beta(B)\}$ the set of probability distributions on $\mathcal{X} \times \mathcal{Y}$ with marginals α and β.

Measure operators. For a continuous map $f : \mathcal{X} \to \mathcal{Y}$, we denote $f_\sharp : \mathcal{M}_1^+(\mathcal{X}) \to \mathcal{M}_1^+(\mathcal{Y})$ the associated *push-forward operator*, which is a linear map between distribution satisfying, for $\alpha \in \mathcal{M}_1^+(\mathcal{X})$ and $B \subset \mathcal{Y}$, $(f_\sharp \alpha)(B) = \alpha(f^{-1}(B))$; or equivalently, for $g \in \mathcal{C}(\mathcal{Y}), \int_{\mathcal{Y}} g(y) \mathrm{d}(f_\sharp \alpha)(y) = \int_{\mathcal{X}} g \circ f(x) \mathrm{d}\alpha(x)$.

For $(\alpha, \beta) \in \mathcal{M}_1^+(\mathcal{X}) \times \mathcal{M}_1^+(\mathcal{Y})$, their *tensor product* measure, denoted $\alpha \otimes \beta \in \mathcal{M}_1^+(\mathcal{X} \times \mathcal{Y})$, satisfies, for $A \subset \mathcal{X}$ and $B \subset \mathcal{Y}, (\alpha \otimes \beta)(A, B) = \alpha(A)\beta(B)$; or equivalently, for $g \in \mathcal{C}(\mathcal{X} \times \mathcal{Y})$, $\int_{\mathcal{X} \times \mathcal{Y}} g(x, y) \mathrm{d}(\alpha \otimes \beta)(x, y) = \int_{\mathcal{X}} \int_{\mathcal{Y}} g(x, y) \mathrm{d}\alpha(x) \mathrm{d}\beta(y)$.

Vectors and matrices. We use small letters (eg, $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$) for vectors and capitals for matrices (eg, A). For a matrix $A = [A_{ij}]$, its transpose is denoted $A^\top$. For two vectors (a, b) (respectively, two matrices (A, B)), their inner product is denoted $\langle a, b \rangle = \sum_i a_i b_i$ (respectively $\langle A, B \rangle$ denotes the Frobenius inner product). The probability n-simplex is denoted $\Sigma_n = \{\mathbf{a} \in (\mathbb{R}_+)^n, \sum_i a_i = 1\}$.

Invariances. We denote S_d the d-sized permutation group. For $\mathbf{x} \in \mathbb{R}^d$ and $\sigma \in S_d$, $\sigma(\mathbf{x}) \stackrel{\text{def.}}{=} (x_{\sigma^{-1}(i)})_{i=1\dots d}$. For $\mathcal{X} \subset \mathbb{R}^d$, the operator mapping $\alpha \in \mathcal{M}_1^+(\mathcal{X})$ to $\sigma_\sharp \alpha \in \mathcal{M}_1^+(\mathcal{X})$ is still denoted σ by simplicity. A function $\mathcal{F} : \mathcal{M}_1^+(\mathcal{X}) \to \mathbb{R}$ is then said to be S_d-invariant if for all $\alpha \in \mathcal{M}_1^+(\mathcal{X})$ and $\sigma \in S_d, \mathcal{F}(\sigma_\sharp \alpha) = \mathcal{F}(\alpha)$. In that sense, α and its permuted counterpart $\sigma_\sharp \alpha$ are then indistinguishable, hence $\mathcal{M}_1^+(\mathcal{X})$ is endowed with the equivalence relation $\sim$ such that $\alpha \sim \beta \iff \exists \sigma \in S_d, \sigma_\sharp \alpha = \beta$. The corresponding quotient space is denoted $\mathcal{M}_1^+(\mathcal{X})_{/\sim}$ or alternatively $\mathcal{R}(\mathcal{X})_{/\sim}$.

Chapter 1: Stochastic Deep Networks

Densities or probability distributions offer a promising alternative data representation to presently well-studied point sets or graphs in neural networks. This is particularly clear in computer vision, where this design can alleviate computational hurdle, as well as preserves the topological structure and retains invariances. Yet, current architectures are either application-oriented, therefore lack versatility, or suffer from the drawback of not metrizing the convergence in law.

In this chapter, we introduce a general neural network pipeline to handle probability measures in their Lagrangian form, which corresponds to sampling. This framework is versatile enough to either process probability measures using recurrent mechanisms, or bridge the gap between probability measures and Euclidean spaces. It is therefore well suited to the variety of machine learning applications, expected to process probability measures as both inputs and outputs.

We prove that these architectures benefit from the desirable property of Lipschitz robustness, and are actually universal approximators for functions mapping measures to measures, that are continuous for the convergence in law. We provide instanciations of such networks in various applicative settings, ranging from classification, generative networks, to predictive tasks.

This chapter is based on [De Bie et al., 2019].

1. Introduction

Deep networks can now handle increasingly complex structured data types, starting historically from images [Krizhevsky et al., 2012] and speech [Hinton et al., 2012] to deal now with shapes [Wu et al., 2015b], sounds [Lee et al., 2009], texts [Lecun et al., 1998] or graphs [Henaff et al., 2015]. In each of these applications, deep networks rely on the composition of several elementary functions, whose tensorized operations stream well on GPUs, and whose computational graphs can be easily automatically differentiated through back-propagation. Initially designed for vectorial features, their extension to *sequences* of vectors using recurrent mechanisms, both as inputs [Hochreiter and Schmidhuber, 1997] or outputs [Sutskever et al., 2014] had an enormous impact, as showcased in machine translation systems [Wu et al., 2016].

Our goal is to devise neural architectures that can handle *probability distributions* under any of their usual form: as discrete measures supported on (possibly weighted) point clouds, or densities one can sample from. Such probability distributions are challenging to handle using recurrent networks because no order between observations can be used to treat them recursively (although some adjustments can be made, as discussed in [Vinyals et al., 2016]) and because, in the discrete case, their size may vary across observations. There is, however, a strong incentive to define neural architectures that can handle distributions as inputs or outputs. This is particularly evident in computer vision, where the naive representation of complex 3D objects as vectors in spatial grids is often too costly memorywise, leads to a loss in detail, destroys topology and is blind to relevant invariances such as shape deformations. These issues were successfully tackled in a string of papers well adapted to such 3D settings [Qi et al., 2017a,b, Fan et al., 2017], including in the generative case [Achlioptas et al., 2018, Yi et al., 2019], even though discussions on their limitations have emerged [Wagstaff et al., 2019, Segol and Lipman, 2020]. In other cases, ranging from physics [Godin et al., 2007], biology [Grover et al., 2011], ecology [Tereshko, 2000] to census data [Guckenheimer et al., 1977], populations cannot be followed at an individual level due to experimental costs or privacy concerns. In such settings where only macroscopic states are available, *densities* appear as the right object to perform inference tasks.

Previous works.

Specificities of Probability Distributions. Data described in point clouds or sampled i.i.d. from a density are given *unordered*. Therefore

architectures dealing with them are expected to be *permutation invariant*; they are also often expected to be equivariant to geometric transformations of input points (translations, rotations) and to capture *local structures* of points. Permutation invariance or equivariance [Ravanbakhsh et al., 2016, 2017], or with respect to general groups of transformations [Gens and Domingos, 2014, Cohen and Welling, 2016, Ravanbakhsh et al., 2017] have been characterized, but without tackling the issue of locality. Pairwise interactions [Chen et al., 2014, Cheng et al., 2016, Guttenberg et al., 2016] are appealing and helpful in building permutation equivariant layers handling local information. Other strategies consist in augmenting the training data by all permutations or finding its *best ordering* [Vinyals et al., 2016]. [Qi et al., 2017a,b] are closer to our work in the sense that they combine the search for local features to permutation invariance, achieved by max pooling.

(Point) Sets *vs.* Probability (Distributions). An important distinction should be made between point *sets*, and point *clouds* which stand usually for discrete probability measures with uniform masses. The natural topology of (point) sets is the Hausdorff distance. That distance is very different from the natural topology for probability distributions, that of the convergence in law, *a.k.a* the weak* topology of measures. The latter is metrized (among other metrics) by the Wasserstein (optimal transport) distance, which plays a key role in our work. This distinction between sets and probability is crucial, because the architectures we propose here are designed to capture stably and efficiently regularity of maps to be learned with respect to the convergence in law. Note that this is a crucial distinction between our work and that proposed in PointNet [Qi et al., 2017a] and PointNet++ [Qi et al., 2017b], which are designed to be smooth and efficients architectures for the Hausdorff topology of point sets. Indeed, they are *not* continuous for the topology of measures (because of the max-pooling step) and cannot approximate efficiently maps which are smooth (e.g. Lipschitz) for the Wasserstein distance. After the publication of this work, we came across [Pevny and Kovarik, 2019], which, similarly to ours, considers learning from probability distributions, however restricted to the regression case, and providing universal approximators. Contrary to their work, we provide a unified framework that considers probability measures as both inputs and outputs, and offer regularity analysis in this more general case.

Another relevant line of work comes from *point process models*, which deal with dynamics of random counting measures or random sets and provide a coherent framework for event modeling, with flexible handling of time.

Popular models such as Poisson processes and Hawkes processes [Belanger et al., 2018, Rajaram et al., 2005, Mei and Eisner, 2017] offer basis for deep parametrizations and extensions, often in the form of recurrent networks [Xiao et al., 2017a, Du et al., 2016, Mei and Eisner, 2017, Xiao et al., 2017b]. Likelihood-based approaches [Belanger et al., 2018, Du et al., 2016, Mei and Eisner, 2017] overwhelmingly dominate the field, while few works use the Wasserstein metric [Xiao et al., 2017a].

Centrality of optimal transport. The Wasserstein distance plays a central role in our architectures that are able to handle measures. Optimal transport has recently gained popularity in machine learning due to fast approximations, which are typically obtained using strongly-convex regularizers such as the entropy [Cuturi, 2013]. The benefits of this regularization paved the way to the use of OT in various settings [Courty et al., 2017, Rolet et al., 2016, Huang et al., 2016]. Although Wasserstein metrics have long been considered for inference purposes [Bassetti et al., 2006], their introduction in deep learning architectures is fairly recent, whether it be for generative tasks [Bernton et al., 2017, Arjovsky et al., 2017, Genevay et al., 2018] or regression purposes [Frogner et al., 2015, Hashimoto et al., 2016]. The purpose of our work is to provide an extension of these works, to ensure that deep architectures can be used at a granulary level on measures directly. In particular, our work shares some of the goals laid out in [Hashimoto et al., 2016], which considers recurrent architectures for measures (a special case of our framework). The most salient distinction with respect to our work is that our building blocks take into account multiple interactions between samples from the distributions, while their architecture has no interaction but takes into account diffusion through the injection of random noise.

Contributions.

In this chapter, we design deep architectures that can (i) map measures to measures; (ii) bridge the gap between measures and Euclidean spaces. They can thus accept as input for instance discrete distributions supported on (weighted) point clouds with an arbitrary number of points, can generate point clouds with an arbitrary number of points (arbitrary refined resolution) and are naturally invariant to permutations in the ordering of the support of the measure. The mathematical idealization of these architectures are infinite dimensional by nature, and they can be computed numerically either by sampling (Lagrangian mode) or by density discretization (Eulerian mode). The Eulerian mode resembles classical convolutional deep network, while

the Lagrangian mode, which we focus on, defines a new class of deep neural models.

Our first contribution is to detail this new framework for supervised and unsupervised learning problems over probability measures, making a clear connexion with the idea of iterative transformation of random vectors. These architectures are based on two simple building blocks: interaction functionals and self-tensorization. This machine learning pipeline works hand-in-hand with the use of optimal transport, both as a mathematical performance criterion (to evaluate smoothness and approximation power of these models) and as a loss functional for both supervised and unsupervised learning.

Our second contribution is theoretical: we prove both quantitative Lipschitz robustness of these architectures for the topology of the convergence in law and universal approximation power.

Our last contribution is a showcase of several instantiations of such deep stochastic networks for classification (mapping measures to vectorial features), generation (mapping back and forth measures to code vectors) and prediction (mapping measures to measures, which can be integrated in a recurrent network).

2. Stochastic Deep Architectures

In this section, we define *elementary blocks*, mapping random vectors to random vectors, which constitute a *layer* of our proposed architectures, and depict how they can be used to build deeper networks.

2.1. Notion of Elementary Block

Our deep architectures are defined by stacking a succession of simple elementary blocks that we now define.

Definition 1 (Elementary Block)**.** *Given a function $f : \mathbb{R}^q \times \mathbb{R}^q \to \mathbb{R}^r$, its associated elementary block $T_f : \mathcal{R}(\mathbb{R}^q) \to \mathcal{R}(\mathbb{R}^r)$ is defined as*

$$\forall\, X \in \mathcal{R}(\mathbb{R}^q), \quad T_f(X) \stackrel{\text{def.}}{=} \mathbb{E}_{X' \sim X}(f(X, X')) \tag{19}$$

where X' is a random vector independent from X having the same distribution.

Discrete random vectors. A particular instance, which is the setting we use in our numerical experiments, is when X is distributed uniformly on a set $(x_i)_{i=1}^n$ of n points i.e. when $\alpha_X = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$. In this case, $Y = T_f(X)$ is also distributed on n points

$$\alpha_Y = \frac{1}{n}\sum_{i=1}^n \delta_{y_i} \quad \text{where} \quad y_i = \frac{1}{n}\sum_{j=1}^n f(x_i, x_j).$$

This elementary operation (19) displaces the distribution of X according to pairwise interactions measured through the map f. As done usually in deep architectures, it is possible to localize the computation at some scale τ by imposing that $f(x, x')$ is zero for $\|x - x'\| \geqslant \tau$, which is also useful to reduce the computation time.

Fully-connected case. As it is customary for neural networks, the map $f : \mathbb{R}^q \times \mathbb{R}^q \to \mathbb{R}^r$ we consider for our numerical applications are affine maps composed by a pointwise non-linearity, i.e.

$$f(x, x') = (\lambda(y_k))_{k=1}^r \quad \text{where} \quad y = A \cdot [x; x'] + b \in \mathbb{R}^r$$

where $\lambda : \mathbb{R} \to \mathbb{R}$ is a pointwise non-linearity (in our experiments, $\lambda(s) = \max(s, 0)$ is the ReLu map), $\cdot$ stands for the matrix-vector product and $[.;.]$ denotes concatenation. The parameter is then $\theta = (A, b)$ where $A \in \mathbb{R}^{r \times 2q}$ is a matrix and $b \in \mathbb{R}^r$ is a bias.

Deterministic layers. Classical "deterministic" deep architectures are recovered as special cases when X is a constant vector, assuming some value $x \in \mathbb{R}^q$ with probability 1, i.e. $\alpha_X = \delta_x$. A stochastic layer can output such a deterministic vector, which is important for instance for classification scores in supervised learning (see Section 4 for an example) or latent code vectors in auto-encoders (see Section 4 for an illustration). In this case, the map $f(x, x') = g(x')$ does not depend on its first argument, so that $Y = T_f(X)$ is constant equal to $y = \mathbb{E}_X(g(X)) = \int_{\mathbb{R}^q} g(x)\mathrm{d}\alpha_X(x)$. Such a layer thus computes a summary statistic vector of X according to g.

Push-Forward. In sharp contrast to the previous remark, one can consider the case $f(x, x') = h(x)$ so that f only depends on its first argument. One then has $T_f(X) = h(X)$, which corresponds to the notion of push-forward of measure, denoted $\alpha_{T_f(X)} = h_\sharp \alpha_X$. For instance, for a discrete law $\alpha_X = \frac{1}{n}\sum_i \delta_{x_i}$ then $\alpha_{T_f(X)} = \frac{1}{n}\sum_i \delta_{h(x_i)}$. The support of the law of X is thus deformed by h.

Higher Order Interactions and Tensorization. Elementary Blocks are generalized to handle higher-order interactions by considering $f : (\mathbb{R}^q)^N \to \mathbb{R}^r$, one then defines

$$T_f(X) \stackrel{\text{def.}}{=} \mathbb{E}_{X_2,\ldots,X_N}(f(X, X_2, \ldots, X_N))$$

where $(X_2, \ldots, X_N)$ are independent and identically distributed copies of X. An equivalent and elegant way to introduce these interactions in a deep architecture is by adding a tensorization layer, which maps $X \mapsto X_2 \otimes \ldots \otimes X_N \in \mathcal{R}((\mathbb{R}^q)^{N-1})$. Section 3 details the regularity and approximation power of these tensorization steps.

2.2. Building Stochastic Deep Architectures

These elementary blocks are stacked to construct deep architectures. A *stochastic deep architecture* is thus a map

$$X \in \mathcal{R}(\mathbb{R}^{q_0}) \mapsto Y = T_{f_T} \circ \cdots \circ T_{f_1}(X) \in \mathcal{R}(\mathbb{R}^{q_T}), \tag{20}$$

where $f_t : \mathbb{R}^{q_{t-1}} \times \mathbb{R}^{q_{t-1}} \to \mathbb{R}^{q_t}$. Typical instances of these architectures includes:

- *Predictive:* this is the general case where the architecture inputs a random vector and outputs another random vector. This is useful to model for instance time evolution using recurrent networks, and is used in Section 4 to tackle a dynamic prediction problem.

- *Discriminative:* in which case Y is constant equal to a vector $y \in \mathbb{R}^{q_T}$ (i.e. $\alpha_Y = \delta_y$) which can represent either a classification score or a latent code vector. Following remarks in Section 2.1, this is achieved by imposing that f_T only depends on its second argument. Section 4 shows applications of this setting to classification and variational auto-encoders (VAE) [Kingma and Welling, 2014].

- *Generative:* in which case the network should input a deterministic code vector $\tilde{x}_0 \in \mathbb{R}^{\tilde{q}_0}$ and should output a random vector Y. This is achieved by adding extra randomization through a fixed random vector $\bar{X}_0 \in \mathcal{R}(\mathbb{R}^{q_0 - \tilde{q}_0})$ (for instance a Gaussian noise) and stacking $X_0 = (\tilde{x}_0, \bar{X}_0) \in \mathcal{R}(\mathbb{R}^{q_0})$. Section 4 shows an application of this setting to VAE generative models. Note that while we focus for simplicity on VAE models, it is possible to use our architectures for GANs [Goodfellow et al., 2014] as well.

Recurrent Nets as Gradient Flows. Following the work of [Hashimoto et al., 2016], in the special case $\mathbb{R}^q = \mathbb{R}^r$, one can also interpret iterative applications of such a T_f (i.e. considering a recurrent deep network) as discrete optimal transport gradient flows [Santambrogio, 2015] (for the W_2 distance, see also Definition (21) in section 3) in order to minimize a quadratic interaction energy $\mathcal{E}(\alpha) \stackrel{\text{def.}}{=} \int_{\mathbb{R}^q \times \mathbb{R}^q} F(x,x')\mathrm{d}\alpha(x)\mathrm{d}\alpha(x')$ (we assume for ease of notation that F is symmetric). Indeed, introducing a step size $\tau > 0$, setting $f(x,x') = x - 2\tau\nabla_x F(x,x')$, one sees that the measure α_{X_ℓ} defined by the iterates $X_{\ell+1} = T_f(X_\ell)$ of a recurrent nets is approximating at time $t = \ell\tau$ the Wasserstein gradient flow $\alpha(t)$ of the energy $\mathcal{E}$. As detailed for instance in [Santambrogio, 2015], such a gradient flow is the solution of the PDE $\frac{\partial \alpha}{\partial t} = \mathrm{div}(\alpha \nabla(\mathcal{E}'(\alpha)))$ where $\mathcal{E}'(\alpha) = \int_{\mathbb{R}^q} F(x,\cdot)\mathrm{d}\alpha(x)$ is the "Euclidean" derivative of $\mathcal{E}$. The pioneering work of [Hashimoto et al., 2016] only considers linear and entropy functionals of the form $\mathcal{E}(\alpha) = \int (F(x) + \log(\frac{\mathrm{d}\alpha}{\mathrm{d}x}))\mathrm{d}\alpha(x)$ which leads to evolutions $\alpha(t)$ being Fokker-Plank PDEs. Our work can thus be interpreted as extending this idea to the more general setting of interaction functionals (see Section 3 for the extension beyond pairwise interactions).

3. Theoretical Guarantees

In order to get some insight on these deep architectures, we now highlight some theoretical results detailing the regularity and approximation power of these functionals. This theoretical analysis relies on the Wasserstein distance, which allows us to make quantitative statements associated to the convergence in law.

3.1. Convergence in Law Topology

Wasserstein distance. In order to measure regularity of the involved functionals, and also to define loss functions to fit these architectures (see Section 4), we consider the p-Wasserstein distance (for $1 \leqslant p < +\infty$) between two probability distributions $(\alpha, \beta) \in \mathcal{M}_+^1(\mathbb{R}^q)$

$$\mathrm{W}_p^p(\alpha,\beta) \stackrel{\text{def.}}{=} \min_{\pi_1=\alpha, \pi_2=\beta} \int_{(\mathbb{R}^q)^2} \|x-y\|^p \mathrm{d}\pi(x,y) \tag{21}$$

where $\pi_1, \pi_2 \in \mathcal{M}_+^1(\mathbb{R}^q)$ are the two marginals of a coupling measure π, and the minimum is taken among coupling measures $\pi \in \mathcal{M}_+^1(\mathbb{R}^q \times \mathbb{R}^q)$.

A classical result (see [Santambrogio, 2015]) asserts that W_1 is a norm,

and can be conveniently computed using

$$\mathrm{W}_1(\alpha,\beta) = \mathrm{W}_1(\alpha-\beta) = \max_{\mathrm{Lip}(g)\leqslant 1} \int_{\mathcal{X}} g \mathrm{d}(\alpha-\beta),$$

where $\mathrm{Lip}(g)$ is the Lipschitz constant of a map $g : \mathcal{X} \to \mathbb{R}$ (with respect to the Euclidean norm unless otherwise stated).

With an abuse of notation, we write $\mathrm{W}_p(X,Y)$ to denote $\mathrm{W}_p(\alpha_X,\alpha_Y)$, but one should be careful that we are considering distances between laws of random vectors. An alternative formulation is

$$\mathrm{W}_p(X,Y) = \min_{X',Y'} \mathbb{E}(\left\|X'-Y'\right\|^p)^{1/p}$$

where (X',Y') is a couple of vectors such that X' (resp. Y') has the same law as X (resp. Y), but of course X' and Y' are not necessarily independent. The Wasserstein distance metrizes the convergence in law (denoted $\rightharpoonup$) in the sense that $X_k \rightharpoonup X$ is equivalent to $\mathrm{W}_1(X_k,X) \to 0$.

In the numerical experiments, we estimate W_p using Sinkhorn's algorithm [Cuturi, 2013], which provides a smooth approximation amenable to (possibly stochastic, see [Genevay et al., 2016]) gradient descent optimization schemes, whether it be for generative or predictive tasks (see Section 4).

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

$$\forall\, (X,Y) \in \mathcal{R}(\mathbb{R}^q)^2,\ \mathrm{W}_1(T(X),T(Y)) \leqslant C\, \mathrm{W}_1(X,Y). \tag{22}$$

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

3.2. Regularity of Building blocks

Elementary blocks. The following proposition shows that elementary blocks are robust to input perturbations. As a consequence, architectures composed of such blocks benefit from Lipschitz robustness as well.

Proposition 3 (Lipschitzness of Elementary Blocks)**.** *If for all x, $f(x,\cdot)$ and $f(\cdot,x)$ are $C(f)$-Lipschitz, then T_f is $2rC(f)$-Lipschitz in the sense of* (22)*.*

As hinted at in Section 2.1, such Elementary Blocks are actually defined as the composition of the push-forward operator and a partial integration operation which we now define. For the sake of clarity, we postpone the proof of Proposition 3 until regularity of both these operations has been detailed.

Push-forward. The push-forward operator allows for modifications of the support while maintaining the geometry of the input measure.

Definition 2 (Push-forward)**.** *For a function $f : \mathcal{X} \to \mathcal{Y}$, we define the push-forward $f_\sharp \alpha \in \mathcal{M}(\mathcal{Y})$ of $\alpha \in \mathcal{M}(\mathcal{X})$ by T as defined by*

$$\forall\, g \in \mathcal{C}(\mathcal{Y}), \quad \int_{\mathcal{Y}} g \mathrm{d}(f_\sharp \alpha) \overset{\text{def.}}{=} \int_{\mathcal{X}} g(f(x)) \mathrm{d}\alpha(x). \tag{23}$$

Note that $f_\sharp : \mathcal{M}(\mathcal{X}) \to \mathcal{M}(\mathcal{Y})$ is a linear operator.

Proposition 4 (Lipschitzness of push-forward)**.** *One has*

$$\mathrm{W}_{\mathcal{Y}}(f_\sharp \alpha, f_\sharp \beta) \leqslant \mathrm{Lip}(f)\, \mathrm{W}_{\mathcal{X}}(\alpha, \beta), \tag{24}$$
$$\mathrm{W}_{\mathcal{Y}}(f_\sharp \alpha, g_\sharp \alpha) \leqslant \|f - g\|_{L^1(\alpha)}, \tag{25}$$

where $\mathrm{Lip}(f)$ *designates the Lipschitz constant of* f*.*

Proof. $\forall h : \mathcal{Y} \to \mathbb{R}$ s.t. $\mathrm{Lip}(h) \leqslant 1$, $\frac{h \circ f}{\mathrm{Lip}(f)}$ is 1-Lipschitz, therefore

$$\int_{\mathcal{X}} \frac{h \circ f}{\mathrm{Lip}(f)} \mathrm{d}(\alpha - \beta) \leqslant \mathrm{W}_{\mathcal{X}}(\alpha, \beta)$$

hence inequality (24). Similarly, $\forall h$ s.t. $\mathrm{Lip}(h) \leqslant 1$,

$$\int_{\mathcal{X}} (h \circ f - h \circ g) \mathrm{d}\alpha \leqslant \int_{\mathcal{X}} \|f(x) - g(x)\|_2 \mathrm{d}\alpha(x)$$

hence inequality (25). □

Integration. We now define a (partial) integration operator.

Definition 3 (Integration)**.** *For $f \in \mathcal{C}(\mathcal{Z} \times \mathcal{X}; \mathcal{Y} = \mathbb{R}^r)$, and $\alpha \in \mathcal{M}(\mathcal{X})$ we denote*

$$f[\cdot, \alpha] \overset{\text{def.}}{=} \int_{\mathcal{X}} f(\cdot, x) \mathrm{d}\alpha(x) : \mathcal{Z} \to \mathcal{Y}.$$

Proposition 5 (Lipschitzness of integration)**.** *With some fixed $\zeta \in \mathcal{M}_+^1(\mathcal{Z})$, one has*

$$\|f[\cdot,\alpha] - f[\cdot,\beta]\|_{L^1(\zeta)} \leqslant r \operatorname{Lip}_2(f) \operatorname{W}_{\mathcal{X}}(\alpha,\beta).$$

where we denoted by $\operatorname{Lip}_2(f)$ a bound on the Lipschitz contant of the function $f(z,\cdot)$ for all z.

Proof.

$$\begin{aligned}
||f[\cdot,\alpha] - f[\cdot,\beta]||_{L^1(\zeta)} &= \int_{\mathcal{Z}} \|f[\cdot,\alpha](z) - f[\cdot,\beta](z)\|_2 \mathrm{d}\zeta(z) \\
&= \int_{\mathcal{Z}} \left\| \int_{\mathcal{X}} f(z,x) \mathrm{d}(\alpha-\beta)(x) \right\|_2 \mathrm{d}\zeta(z) \\
&\leqslant \int_{\mathcal{Z}} \left\| \int_{\mathcal{X}} f(z,x) \mathrm{d}(\alpha-\beta)(x) \right\|_1 \mathrm{d}\zeta(z) \\
&= \int_{\mathcal{Z}} \sum_{i=1}^{r} \left| \int_{\mathcal{X}} f_i(z,x) \mathrm{d}(\alpha-\beta)(x) \right| \mathrm{d}\zeta(z) \\
&\leqslant \sum_{i=1}^{r} \operatorname{Lip}_2(f_i) \operatorname{W}_{\mathcal{X}}(\alpha,\beta) \\
&\leqslant r \operatorname{Lip}_2(f) \operatorname{W}_{\mathcal{X}}(\alpha,\beta)
\end{aligned}$$

where we denoted by $\operatorname{Lip}_2(f_i)$ a bound on the Lipschitz contant of the function $f_i(z,\cdot)$ (i-th component of f) for all z, since again, $\frac{f_i}{\operatorname{Lip}_2(f_i)}$ is 1-Lipschitz. □

Now that Lipschitzness of the push-forward operator and the partial integration operation have been established, Lipschitz robustness of our Elementary Block can be detailed.

Proof. (of Proposition 3) Let us stress that the elementary block $T_f(X)$ defined in (19) only depends on the law α_X. In the following, for a measure α we denote $\mathcal{T}_f(\alpha_X)$ the law of $T_f(X)$. The goal is thus to show that $\mathcal{T}_f$ is Lipschitz for the Wasserstein distance.

For a measure $\alpha \in \mathcal{M}(\mathcal{X})$ (where $\mathcal{X} = \mathbb{R}^q$), the measure $\beta = \mathcal{T}_{\mathcal{F}}(\alpha) \in \mathcal{M}(\mathcal{Y})$ (where $\mathcal{Y} = \mathbb{R}^r$) is defined via the identity, for all $g \in \mathcal{C}(\mathcal{Y})$,

$$\int_{\mathcal{Y}} g(y) \mathrm{d}\beta(y) = \int_{\mathcal{X}} g\left(\int_{\mathcal{X}} f(z,x) \mathrm{d}\alpha(x) \right) \mathrm{d}\alpha(z).$$

Let us first remark that an elementary block, when view as operating on measures, can be decomposed using the aforementioned push-forward and

integration operators, since

$$\mathcal{T}_{\mathcal{F}}(\alpha) = f[\cdot, \alpha]_\sharp \alpha.$$

Using the fact that $\mathrm{W}_{\mathcal{X}}$ is a norm,

$$\begin{aligned}
&\mathrm{W}_{\mathcal{X}}(\mathcal{T}_{\mathcal{F}}(\alpha), \mathcal{T}_f(\beta)) \\
&\leqslant \mathrm{W}_{\mathcal{X}}(\mathcal{T}_{\mathcal{F}}(\alpha), f[\cdot, \beta]_\sharp \alpha) + \mathrm{W}_{\mathcal{X}}(f[\cdot, \beta]_\sharp \alpha, \mathcal{T}_f(\beta)) \\
&\leqslant \|f[\cdot, \alpha] - f[\cdot, \beta]\|_{L^1(\alpha)} + \mathrm{Lip}(f[\cdot, \beta])\, \mathrm{W}_{\mathcal{X}}(\alpha, \beta),
\end{aligned}$$

where we used the Lipschitzness of the push-forward, Proposition 4. Moreover, for $(z_1, z_2) \in \mathcal{X}^2$,

$$\begin{aligned}
\|f[z_1, \beta] - f[z_2, \beta]\|_2 \leqslant \|f[z_1, \beta] - f[z_2, \beta]\|_1 &= \left\| \int_{\mathcal{X}} (f(z_1, \cdot) - f(z_2, \cdot)) \mathrm{d}\beta \right\|_1 \\
&\leqslant \sum_{i=1}^{r} \left| \int_{\mathcal{X}} (f_i(z_1, \cdot) - f_i(z_2, \cdot)) \mathrm{d}\beta \right| \\
&\leqslant \sum_{i=1}^{r} \int_{\mathcal{X}} |f_i(z_1, \cdot) - f_i(z_2, \cdot)| \mathrm{d}\beta \\
&\leqslant \sum_{i=1}^{r} \mathrm{Lip}_1(f_i) \|z_1 - z_2\|_2 \leqslant r\, \mathrm{Lip}_1(f) \|z_1 - z_2\|_2,
\end{aligned}$$

where we denoted by $\mathrm{Lip}_1(f_i)$ a bound on the Lipschitz contant of the function $f_i(\cdot, x)$ for all x. Hence $\mathrm{Lip}(f[\cdot, \beta]) \leqslant r\, \mathrm{Lip}_1(f)$. In addition, Lipschitzness of integration, Proposition 5 yields

$$\mathrm{W}_{\mathcal{X}}(\mathcal{T}_{\mathcal{F}}(\alpha), \mathcal{T}_f(\beta)) \leqslant r\, \mathrm{Lip}_2(f)\, \mathrm{W}_{\mathcal{X}}(\alpha, \beta) + r\, \mathrm{Lip}_1(f)\, \mathrm{W}_{\mathcal{X}}(\alpha, \beta)$$

□

It is worth noting that, as a composition of Lipschitz functions defines Lipschitz maps, the architectures of the form (20) are thus Lipschitz, with a Lipschitz constant upper-bounded by $\prod_t 2q_t C(f_t)$, where we used the notations of Proposition 3.

Tensorization. As highlighted in Section 2.1, tensorization plays an important role to define higher-order interaction blocks.

Definition 4 (Tensor product). *Given $(X, Y) \in \mathcal{R}(\mathcal{X}) \times \mathcal{R}(\mathcal{Y})$, a tensor product random vector is $X \otimes Y \overset{\text{def.}}{=} (X', Y') \in \mathcal{R}(\mathcal{X} \times \mathcal{Y})$ where X' and Y' are independent and have the same laws as X and Y. This means that $\mathrm{d}\alpha_{X\otimes Y}(x, y) = \mathrm{d}\alpha_X(x)\mathrm{d}\alpha_Y(y)$ is the tensor product of the measures.*

Remark 1 (Tensor Product between Discrete Measures). If we consider random vectors supported on point clouds, with laws $\alpha_X = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$ and $\alpha_Y = \frac{1}{m}\sum_{j=1}^m \delta_{y_j}$, then $X \otimes Y$ is a discrete random vector supported on nm points, since $\alpha_{X\otimes Y} = \frac{1}{nm}\sum_{i,j} \delta_{(x_i,y_j)}$.

The following proposition shows that tensorization blocks maintain the stability property of a deep architecture.

Proposition 6 (Lipschitzness of tensorization). *For $(X, X', Y, Y') \in \mathcal{R}(\mathcal{X})^2 \times \mathcal{R}(\mathcal{Y})^2$, one has*

$$\mathrm{W}_1(X \otimes Y, X' \otimes Y') \leqslant \mathrm{W}_1(X, X') + \mathrm{W}_1(Y, Y').$$

Proof. One has

$$\begin{aligned}
\mathrm{W}_1(\alpha \otimes \beta, \alpha' \otimes \beta') &= \max_{\mathrm{Lip}(g)\leqslant 1} \int_1 g(x, y)[\mathrm{d}\alpha(x)\mathrm{d}\beta(y) - \mathrm{d}\alpha'(x)\mathrm{d}\beta'(y)] \\
&= \max_{\mathrm{Lip}(g)\leqslant 1} \int_{\mathcal{X}}\int_{\mathcal{Y}} g(x, y)[\mathrm{d}\beta(y) - \mathrm{d}\beta'(y)]\mathrm{d}\alpha(x) \\
&\quad + \int_{\mathcal{Y}}\int_{\mathcal{X}} g(x, y)[\mathrm{d}\alpha(x) - \mathrm{d}\alpha'(x)]\mathrm{d}\beta(y),
\end{aligned}$$

which yields the result. □

3.3. Approximation Theorems

Universality of elementary block. The following theorem shows that any continuous map between random vectors can be approximated to arbitrary precision using three elementary blocks. Note that it includes through Λ a fixed random input which operates as an "excitation block" similar to the generative VAE models studied in Section 4.2.

Theorem 3. *Let $\mathcal{F} : \mathcal{R}(\mathcal{X}) \to \mathcal{R}(\mathcal{Y})$ be a continuous map for the convergence in law, where $\mathcal{X} \subset \mathbb{R}^q$ and $\mathcal{Y} \subset \mathbb{R}^r$ are compact. Then $\forall \eta > 0$ there exists three continuous maps f, g, h such that*

$$\forall X \in \mathcal{R}(\mathcal{X}), \quad \mathrm{W}_1(\mathcal{F}(X), T_h \circ \Lambda \circ T_g \circ T_f(X)) \leqslant \eta. \tag{26}$$

where $\Lambda : X \mapsto (X, U)$ concatenates a uniformly distributed random vector U.

The architecture that we use to prove this theorem is displayed on Figure 3, bottom (left). Since f, g and h are smooth maps, according to the universality theorem of neural networks [Cybenko, 1989, Leshno et al., 1993] (assuming some restriction on the non-linearity λ, namely its being a nonconstant, bounded and continuous function), it is possible to replace each of them (at the expense of increasing η) by a sequence of fully connected layers (as detailed in Section 2.1). This is detailed further down this section.

Since deterministic vectors are a special case of random vectors (see Section 2.1), this results encompasses as a special case universality for vector-valued maps $\mathcal{F} : \mathcal{R}(\Omega) \to \mathbb{R}^r$ (used for instance in classification in Section 4.1) and in this case only 2 elementary blocks are needed. Of course the classical universality of multi-layer perceptron [Cybenko, 1989, Leshno et al., 1993] for vectors-to-vectors maps $\mathcal{F} : \mathbb{R}^q \to \mathbb{R}^r$ is also a special case (using a single elementary block).

Before stating a proof for Theorem 3, we introduce and prove two useful lemmas: (i) the first one shows how the gap between a probability measure and its discretized counterpart can be controlled; (ii) the second one shows the existence of a continuous noise-reshaping function mapping a uniform noise to a target distribution.

Approximation by discrete measures. The following lemma shows how to control the approximation error between an arbitrary random variable and a discrete variable obtained by computing moments against localized functions on a grid.

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

$$\mathrm{W}_1(\hat{\alpha}_N, \alpha) \leqslant \max_{1 \leqslant j \leqslant N} \Delta_j.$$

Proof. We define $\pi \in \mathcal{M}_+^1(\Omega^2)$, a transport plan coupling marginals α and $\hat{\alpha}_N$, by imposing for all $f \in \mathcal{C}(\Omega^2)$,

$$\int_{\Omega^2} f \mathrm{d}\pi = \sum_{j=1}^N \int_{S_j} f(x, x_j) \varphi_j(x) \mathrm{d}\alpha(x).$$

π indeed is a transport plan, since for all $g \in \mathcal{C}(\Omega)$,

$$\begin{aligned}\int_{\Omega^2} g(x)\mathrm{d}\pi(x,y) &= \sum_{j=1}^N \int_{S_j} g(x)\varphi_j(x)\mathrm{d}\alpha(x) = \sum_{j=1}^N \int_{\Omega} g(x)\varphi_j(x)\mathrm{d}\alpha(x) \\ &= \int_\Omega g(x)\left(\sum_{j=1}^N \varphi_j(x)\right)\mathrm{d}\alpha(x) = \int_\Omega g\mathrm{d}\alpha.\end{aligned}$$

Also,

$$\begin{aligned}\int_{\Omega^2} g(y)\mathrm{d}\pi(x,y) &= \sum_{j=1}^N \int_{S_j} g(x_j)\varphi_j\mathrm{d}\alpha = \sum_{j=1}^N g(x_j)\int_\Omega \varphi_j\mathrm{d}\alpha \\ &= \sum_{j=1}^N \alpha_j g(x_j) = \int_\Omega g\mathrm{d}\hat{\alpha}_N.\end{aligned}$$

By definition of the Wasserstein-1 distance,

$$\begin{aligned}\mathrm{W}(\hat{\alpha}_N,\alpha) &\leqslant \int_{\Omega^2} \|x-y\|\mathrm{d}\pi(x,y) = \sum_{j=1}^N \int_{S_j} \varphi_j(x)\|x-x_j\|\mathrm{d}\alpha(x) \\ &\leqslant \sum_{j=1}^N \int_{S_j} \varphi_j\Delta_j\mathrm{d}\alpha \\ &\leqslant \left(\sum_{i=1}^N \int_\Omega \varphi_i\mathrm{d}\alpha\right)\max_{1\leqslant j\leqslant N}\Delta_j = \max_{1\leqslant j\leqslant N}\Delta_j.\end{aligned}$$

□

Parametric Push-Forward. An ingredient of the proof of the universality Theorem 3 is the construction of a noise-reshaping function H which maps a uniform noise to another distribution parametrized by b.

Lemma 2. *There exists a continuous map $(b,u) \in \Sigma_m \times [0,1]^r \mapsto H(b,u)$ so that the random vector $H(b,U)$ has law $\beta \overset{\text{def.}}{=} (1-\eta)D^*_{\mathcal{Y}}(b) + \eta\mathcal{U}$, where U has density $\mathcal{U}$ (uniformly distributed on $[0,1]^r$).*

Proof. Since both the input measure $\mathcal{U}$ and the output measure β have densities and have support on convex set, one can use for map $H(b,\cdot)$ the optimal transport map between these two distributions for the squared Euclidean cost,

which is known to be a continuous function, see for instance [Santambrogio, 2015][Sec. 1.7.6]. It is also possible to define a more direct transport map (which is not in general optimal), known as Dacorogna-Moser transport, see for instance [Santambrogio, 2015][Box 4.3]. □

We are now ready to state a proof for Theorem 3.

Proof. (of Theorem 3) In the following, we denote the probability simplex as $\Sigma_n = \left\{a \in \mathbb{R}_+^n \; ; \; \sum_i a_i = 1\right\}$. Without loss of generality, we assume $\mathcal{X} \subset [0,1]^q$ and $\mathcal{Y} \subset [0,1]^r$. We consider two uniform grids of n and m points $(x_i)_{i=1}^n$ of $[0,1]^q$ and $(y_j)_{j=1}^m$ of $[0,1]^r$. On these grids, we consider the usual piecewise affine P1 finite element bases $(\varphi_i)_{i=1}^n$ and $(\psi_j)_{j=1}^m$, which are continuous hat functions supported on cells $(R_i)_i$ and $(S_j)_j$ which are cubes of width $2/n^{1/q}$ and $2/m^{1/r}$. We define discretization operators as

$$D_{\mathcal{X}} : \alpha \in \mathcal{M}_1^+(\mathcal{X}) \mapsto \left(\int_{R_i} \varphi_i \mathrm{d}\alpha\right)_{i=1}^n \in \Sigma_n$$

and

$$D_{\mathcal{Y}} : \beta \in \mathcal{M}_1^+(\mathcal{Y}) \mapsto \left(\int_{S_j} \psi_j \mathrm{d}\beta\right)_{j=1}^m \in \Sigma_n.$$

We also define

$$D_{\mathcal{X}}^* : a \in \Sigma_n \mapsto \sum_i a_i \delta_{x_i} \in \mathcal{M}_1^+(\mathcal{X})$$

and

$$D_{\mathcal{Y}}^* : b \in \Sigma_m \mapsto \sum_j b_j \delta_{y_i} \in \mathcal{M}_1^+(\mathcal{Y}).$$

The map $\mathcal{F}$ induces a discrete map $G : \Sigma_n \to \Sigma_m$ defined by $G \stackrel{\text{def.}}{=} D_{\mathcal{Y}} \circ \mathcal{F} \circ D_{\mathcal{X}}^*$. Remark that $D_{\mathcal{X}}^*$ is continuous from Σ_n (with the usual topology on $\mathbb{R}^n$) to $\mathcal{M}_+^1(\mathcal{X})$ (with the convergence in law topology), $\mathcal{F}$ is continuous (for the convergence in law), $D_{\mathcal{Y}}$ is continuous from $\mathcal{M}_+^1(\mathcal{Y})$ (with the convergence in law topology) to Σ_m (with the usual topology on $\mathbb{R}^m$). This shows that G is continuous.

For any $b \in \Sigma_m$, Lemma 2 proved above defines a continuous map H so that, defining U to be a random vector uniformly distributed on $[0,1]^r$ (with law $\mathcal{U}$), $H(b,U)$ has law $(1-\eta)D_{\mathcal{Y}}^*(b) + \eta\mathcal{U}$.

We now have all the ingredients, and define the three continuous maps for the elementary blocks as

$$f(x,x') = (\varphi_i(x'))_{i=1}^n \in \mathbb{R}^n, \quad g(a,a') = G(a') \in \mathbb{R}^m,$$

$$\text{and} \quad h((b,u),(b',u')) = H(b,u) \in \mathcal{Y}.$$

The corresponding architecture is displayed on Figure 3, bottom. Using these maps, one needs to control the error between $\mathcal{F}$ and $\hat{\mathcal{F}} \stackrel{\text{def.}}{=} T_h \circ \Lambda \circ T_g \circ T_f = H_\sharp \circ \Lambda \circ D_{\mathcal{Y}} \circ \mathcal{F} \circ D^*_{\mathcal{X}} \circ \mathcal{D}_{\mathcal{X}}$ where we denoted $H_\sharp(b) \stackrel{\text{def.}}{=} H(b,\cdot)_\sharp \mathcal{U}$ the law of $H(b,U)$ (i.e. the pushforward of the uniform distribution $\mathcal{U}$ of U by $H(b,\cdot)$).

(i) We define $\hat{\alpha} \stackrel{\text{def.}}{=} D^*_{\mathcal{X}} \mathcal{D}_{\mathcal{X}}(\alpha)$. The diameters of the cells R_i is $\Delta_j = \sqrt{q}/n^{1/q}$, so that Lemma 1 proved above shows that $\mathrm{W}_1(\alpha,\hat{\alpha}) \leqslant \sqrt{q}/n^{1/q}$. Since $\mathcal{F}$ is continuous for the convergence in law, choosing n large enough ensures that $\mathrm{W}_1(\mathcal{F}(\alpha),\mathcal{F}(\hat{\alpha})) \leqslant \eta$.

(ii) We define $\hat{\beta} \stackrel{\text{def.}}{=} D^*_{\mathcal{Y}} D_{\mathcal{Y}} \mathcal{F}(\hat{\alpha})$. Similarly, using m large enough ensures that $\mathrm{W}_1(\mathcal{F}(\hat{\alpha}),\hat{\beta}) \leqslant \eta$.

(iii) Lastly, let us define $\tilde{\beta} \stackrel{\text{def.}}{=} H_\sharp \circ D_{\mathcal{Y}}(\hat{\beta}) = \hat{\mathcal{F}}(\alpha)$. By construction of the map H in Lemma 2, one has hat $\tilde{\beta} = (1-\eta)\hat{\beta} + \eta\mathcal{U}$ so that $\mathrm{W}_1(\tilde{\beta},\hat{\beta}) = \eta\,\mathrm{W}_1(\hat{\beta},\mathcal{U}) \leqslant C\eta$ for the constant $C = 2\sqrt{r}$ since the measures are supported in a set of diameter $\sqrt{r}$.

Putting these three bounds (i), (ii) and (iii) together using the triangular inequality shows that $\mathrm{W}_1(\mathcal{F}(\alpha),\hat{\mathcal{F}}(\alpha)) \leqslant (2+C)\eta$. □

We now detail how the maps f, g, and h involved in Theorem 3 can each be approximated by neural networks [Cybenko, 1989, Leshno et al., 1993], so that the measure-valued function of interest $\mathcal{F}$ can be approached by a neural architecture as well.

Proof. (Approximation by neural networks related to Theorem 3) For the sake of simplicity, we first give the proof in the case $\mathcal{F} : \mathcal{R}(\Omega) \to \mathbb{R}^r$, i.e. no mapping h is needed (the proof being similar for the general case), and only two Elementary Blocks are needed. Furthermore, without loss of generality, we consider the real-valued case $r = 1$. Let $\eta > 0$, then Theorem 3 shows that $\mathcal{F}$ can be approximated arbitrarily close (up to $\frac{\eta}{3}$) by a composition of functions of the form $f\left(\mathbb{E}_{X\sim\alpha}\left(g(X)\right)\right)$. We now show how to approximate the continuous functions f and g by two neural networks

(i) $g_\theta(x) \stackrel{\text{def.}}{=} C_1\lambda(A_1 x + b_1) : \mathbb{R}^d \to \mathbb{R}^N$,

(ii) $f_\xi(x) \stackrel{\text{def.}}{=} C_2\lambda(A_2 x + b_2) : \mathbb{R}^N \to \mathbb{R}$,

such that

$$\forall \alpha \in \mathcal{M}^1_+(\Omega), \quad \left|\mathcal{F}(\alpha) - f_\xi\left(\mathbb{E}_{X\sim\alpha}\left(g_\theta(X)\right)\right)\right| < \eta.$$

where N, p_1, p_2 are integers, $A_1 \in \mathbb{R}^{p_1 \times d}$, $A_2 \in \mathbb{R}^{p_2 \times N}$, $C_1 \in \mathbb{R}^{N \times p_1}$, $C_2 \in \mathbb{R}^{1 \times p_2}$ weight matrices and $b_1 \in \mathbb{R}^{p_1}$, $b_2 \in \mathbb{R}^{p_2}$ are biases.

By triangular inequality, we upper-bound the difference of interest

$$|\mathcal{F}(\alpha) - f_\xi\left(\mathbb{E}_{X\sim\alpha}\left(g_\theta(X)\right)\right)|$$

by a sum of three terms:

(i) $|\mathcal{F}(\alpha) - f\left(\mathbb{E}_{X\sim\alpha}\left(g(X)\right)\right)|$

(ii) $|f\left(\mathbb{E}_{X\sim\alpha}\left(g(X)\right)\right) - f_\xi\left(\mathbb{E}_{X\sim\alpha}\left(g(X)\right)\right)|$

(iii) $|f_\xi\left(\mathbb{E}_{X\sim\alpha}\left(g(X)\right)\right) - f_\xi\left(\mathbb{E}_{X\sim\alpha}\left(g_\theta(X)\right)\right)|$

and bound each term by $\frac{\eta}{3}$, which yields the result. The bound on the first term directly comes from theorem 1 and yields constant N which depends on η. The bound on the second term is a direct application of the universal approximation theorem [Cybenko, 1989, Leshno et al., 1993]. Indeed, since α is a probability measure, input values of f lie in a compact subset of $\mathbb{R}^N$: $\left\|\int_\Omega g(x)\mathrm{d}\alpha\right\|_\infty \leqslant \max_{x\in\Omega}\max_i |g_i(x)|$, hence the theorem [Cybenko, 1989, Leshno et al., 1993] is applicable as long as λ is a nonconstant, bounded and continuous function. Let us focus on the third term. Uniform continuity of f_ξ yields the existence of $\delta > 0$ s.t. $||u - v||_1 < \delta$ implies $|f_\xi(u) - f_\xi(v)| < \frac{\eta}{3}$. Let us apply the universal approximation theorem: each component g_i of g can be approximated by a neural network $g_{\theta,i}$ up to $\frac{\delta}{N}$. Therefore:

$$\begin{aligned}\left\|\mathbb{E}_{X\sim\alpha}\left(g(X) - g_\theta(X)\right)\right\|_1 &\leqslant \mathbb{E}_{X\sim\alpha}\|g(X) - g_\theta(X)\|_1 \\ &\leqslant \sum_{i=1}^{N}\int_\Omega |g_i(x) - g_{\theta,i}(x)|\mathrm{d}\alpha(x) \leqslant N \times \frac{\delta}{N} = \delta\end{aligned}$$

since α is a probability measure. This proves the bound on the third term, with $u = \mathbb{E}_{X\sim\alpha}\left(g(X)\right)$ and $v = \mathbb{E}_{X\sim\alpha}\left(g_\theta(X)\right)$ in the definition of uniform continuity.

We proceed similarly in the general case $\mathcal{F} : \mathcal{R}(\mathcal{X}) \to \mathcal{R}(\mathcal{Y})$, and upper-bound the Wasserstein distance by a sum of four terms by triangular inequality. The same ingredients (namely uniform continuity together with the universal approximation theorem [Cybenko, 1989, Leshno et al., 1993], since all functions f, g and h have input and output constrained in in compact sets) allow us to conclude. □

Universality of tensorization. We now further investigate the advantages of tensorization of the input measures, namely its capacity to acquire universal approximation abilities. The following Theorem shows that in fact, one can approximate any real-valued continuous map using a high enough order of tensorization followed by an elementary block.

Theorem 4. *Let $\mathcal{F} : \mathcal{R}(\Omega) \to \mathbb{R}$ a continuous map for the convergence in law, where $\Omega \subset \mathbb{R}^q$ is compact. Then $\forall \eta > 0$, there exists $n > 0$ and a continuous function f such that*

$$\forall\, X \in \mathcal{R}(\Omega), \quad |\mathcal{F}(X) - T_f \circ \theta_n(X)| \leqslant \eta \tag{27}$$

where $\theta_n(X) = X \otimes \ldots \otimes X$ is the n-fold self tensorization.

Proof. We denote $\mathcal{C}(\mathcal{M}_+^1(\Omega))$ the space of functions taking probability measures on a compact set Ω to $\mathbb{R}$ which are continuous for the weak-$*$ topology. We denote the set of integrals of tensorized polynomials on Ω as

$$\mathcal{A}_\Omega \stackrel{\text{def.}}{=} \left\{ \begin{array}{c} \mathcal{F} : \mathcal{M}_+^1(\Omega) \to \mathbb{R}, \exists n \in \mathbb{N}, \exists \varphi : \Omega^n \to \mathbb{R}, \\ \forall \mu \in \mathcal{M}_+^1(\Omega), \mathcal{F}(\mu) = \displaystyle\int_{\Omega^n} \varphi \mathrm{d}\mu^{\otimes n} \end{array} \right\}.$$

The goal is to show that $\mathcal{A}_\Omega$ is dense in $\mathcal{C}(\mathcal{M}_+^1(\Omega))$.

Since Ω is compact, Banach-Alaoglu theorem shows that $\mathcal{M}_+^1(\Omega)$ is weakly-$*$ compact. Therefore, in order to use Stone-Weierstrass theorem, to show the density result, we need to show that $\mathcal{A}_\Omega$ is an algebra that separates points, and that, for all probability measure α, $\mathcal{A}_\Omega$ contains a function that does not vanish at α. For this last point, taking $n = 1$ and $\varphi = 1$ defines the function $\mathcal{F}(\alpha) = \int_\Omega \mathrm{d}\alpha = 1$ that does not vanish in α since it is a probability measure. Let us then show that $\mathcal{A}_\Omega$ is a subalgebra of $\mathcal{C}(\mathcal{M}_+^1(\Omega))$:

(i) stability by a scalar follows from the definition of $\mathcal{A}_\Omega$;

(ii) stability by sum: given $(\mathcal{F}_1, \mathcal{F}_2) \in \mathcal{A}_\Omega^2$ (with associated functions (φ_1, φ_2) of degrees (n_2, n_2)), denoting $n \stackrel{\text{def.}}{=} \max(n_1, n_2)$ and

$$\varphi(x_1, \ldots, x_n) \stackrel{\text{def.}}{=} \varphi_1(x_1, \ldots, x_{n_1}) + \varphi_2(x_1, \ldots, x_{n_2})$$

shows that $\mathcal{F}_1 + \mathcal{F}_2 = \int_{\Omega^n} \varphi \mathrm{d}\mu^{\otimes n}$ and hence $\mathcal{F}_1 + \mathcal{F}_2 \in \mathcal{A}_\Omega$;

(iii) stability by product: similarly as for the sum, denoting this time $n = n_1 + n_2$ and introducing

$$\varphi(x_1, \ldots, x_n) = \varphi_1(x_1, \ldots, x_{n_1}) \times \varphi_2(x_{n_1+1}, \ldots, x_n)$$

shows that $\mathcal{F} = \mathcal{F}_1 \times \mathcal{F}_2 \in \mathcal{A}_\Omega$, using Fubini's theorem.

Lastly, we show the separation of points: if two probability measures (α, β) on Ω are different (not equal almost everywhere), there exists a set $\Omega_0 \subset \Omega$ such that $\alpha(\Omega_0) \neq \beta(\Omega_0)$; taking $n = 1$ and $\varphi = \mathbb{1}_{\Omega_0}$, we obtain, after smoothing φ to make it continuous, a function $\mathcal{F} \in \mathcal{A}_\Omega$ such that $\mathcal{F}(\alpha) \neq \mathcal{F}(\beta)$. $\square$

The architecture used for this theorem is displayed on the bottom (right) of Figure 3. The function f appearing in (27) plays a similar role as in (26), but note that the two-layers factorizations provided by these two theorems are very different. It is an interesting avenue for future work to compare them theoretically and numerically.

4. Applications

To exemplify the use of our stochastic deep architectures, we consider classification, generation and dynamic prediction tasks. The goal is to highlight the versatility of these architectures and their ability to handle as input and/or output both probability distributions and vectors. In all cases, the procedures displayed similar results when rerun, hence results can be considered as quite stable and representative. We also illustrate the gain in maintaining the measure representation along several layers of the architecture. The code used to produce all results in this section is available at: `https://github.com/gdebie/stochastic-deep-networks`.

4.1. Classification tasks

MNIST Dataset. We perform classification on the 2D MNIST dataset of handwritten digits. To convert a MNIST image into a 2D point cloud, we threshold pixel values (threshold $\rho = 0.5$) and use as a support of the input empirical measure the $n = 256$ pixels of highest intensity, represented as points $(x_i)_{i=1}^n \subset \mathbb{R}^2$ (if there are less than $n = 256$ pixels of intensity over ρ, we repeat input coordinates), which are remapped along each axis by mean and variance normalization. Each image is therefore turned into a sum of $n = 256$ Diracs $\frac{1}{n}\sum_i \delta_{x_i}$. Our stochastic network architecture is displayed on the top of Figure 3 and is composed of 5 elementary blocks $(T_{f_k})_{k=1}^5$ with an interleaved self-tensorisation layer $X \mapsto X \otimes X$. The first elementary block T_{f_1} maps measures to measures, the second one T_{f_2} maps a measure to a deterministic vector (i.e. does not depend on its first coordinate, see Section 2.1), and the last layers are classical vectorial fully-connected ones. We use a ReLu non-linearity λ (see Section 2.1). The weights are learnt with a weighted cross-entropy loss function over a training set of 55,000 examples and tested on a set of 10,000 examples. Initialization is performed through

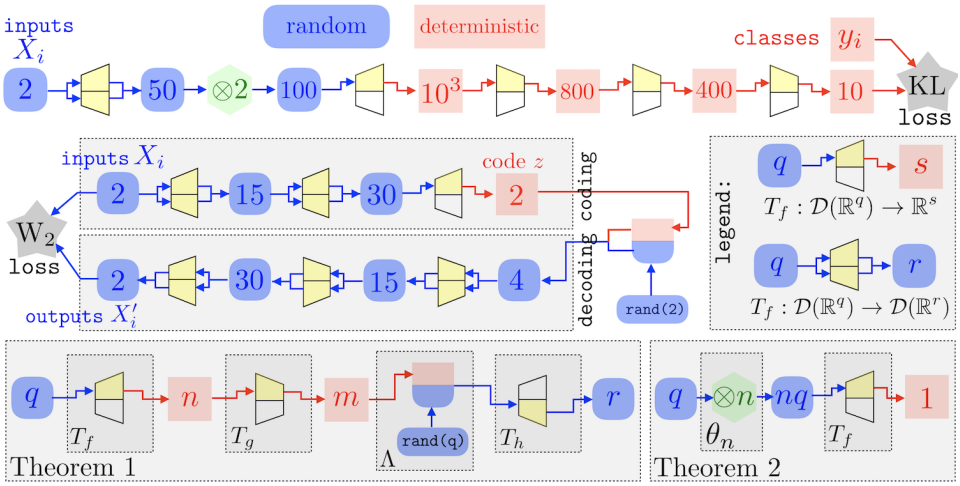


Figure 3: Top and center: two examples of deep stochastic architectures applied to the MNIST dataset: top for classification purpose (Section 4.1), center for generative model purpose (Section 4.2). Bottom: architecture for the proof of Theorems 3 and 4.

the Xavier method [Glorot and Bengio, 2010] and learning with the Adam optimizer [Kingma and Ba, 2014]. Table 1 displays our results, compared with the PointNet [Qi et al., 2017a] baseline. We observe that maintaining stochasticity among several layers is beneficial (as opposed to replacing one Elementary Block with a fully connected layer allocating the same amount of memory).

Table 1: MNIST classification results

	input type	error (%)
PointNet	point set	0.78
Ours	measure (1 stochastic layer)	1.07
Ours	measure (2 stochastic layers)	**0.76**

ModelNet40 Dataset. We evaluate our model on the ModelNet40 [Wu et al., 2015a] shape classification benchmark. The dataset contains 3D CAD models from 40 man-made categories, split into 9,843 examples for training and 2,468 for testing. We consider $n = 1,024$ samples on each surface, obtained by a farthest point sampling procedure. Our classification network is similar to the one displayed on top of Figure 3, excepted that the layer

dimensions are $[3, 10, 500, 800, 400, 40]$. Our results are displayed in figure 2. As previously observed in 2D, peformance is improved by maintaining stochasticity among several layers, for the same amount of allocated memory.

Table 2: ModelNet40 classification results

	input type	accuracy (%)
3DShapeNets	volume	77
Pointnet	point set	89.2
Ours	measure (1 stochastic layer)	82.0
Ours	measure (2 stochastic layers)	**83.5**

4.2. Generative networks

We further evaluate our framework for generative tasks, on a VAE-type model [Kingma and Welling, 2014] – note that it would be possible to use our architectures for GANs [Goodfellow et al., 2014] as well. The task consists in generating outputs resembling the data distribution by decoding a random variable z sampled in a latent space $\mathcal{Z}$. The model, an encoder-decoder architecture, is learnt by comparing input and output measures using the W_2 Wasserstein distance loss, approximated using Sinkhorn's algorithm [Cuturi, 2013, Genevay et al., 2018]. Following [Kingma and Welling, 2014], a Gaussian prior is imposed on the latent variable z. The encoder and the decoder are two mirrored architectures composed of two elementary blocks and three fully-connected layers each. The corresponding stochastic network architecture is displayed on the bottom of 3. Figure 4 displays an application on the MNIST database where the latent variable $z \in \mathbb{R}^2$ parameterizes a 2D of manifold of generated digits. We use as input and output discrete probability measures of $n = 100$ Diracs, displayed as point clouds on the right of Figure 4.

4.3. Dynamics Prediction

Birds of a Feather. The Cucker-Smale flocking model [Cucker and Smale, 2007] is non-linear dynamical system modelling the emergence of coherent behaviors, as for instance in the evolution of a flock of birds, by solving for positions and speed $x_i(t) \overset{\text{def.}}{=} (p_i(t) \in \mathbb{R}^d, v_i(t) \in \mathbb{R}^d)$ for $i = 1, \dots, n$

$$\dot{p}(t) = v(t), \quad \text{and} \quad \dot{v}(t) = \mathcal{L}(p(t))v(t) \tag{28}$$

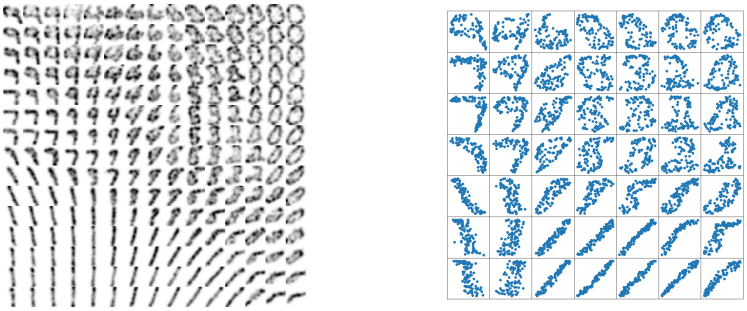


Figure 4: Left: Manifold of digits generated by the VAE network displayed on the bottom of 3. Right: Corresponding point cloud (displaying only a subset of the left images).

where $\mathcal{L}(p) \in \mathbb{R}^{n\times n}$ is the Laplacian matrix associated to a group of points $p \in (\mathbb{R}^d)^n$

$$\mathcal{L}(p)_{i,j} \overset{\text{def.}}{=} \frac{1}{1+\|p_i - p_j\|^m},\ \mathcal{L}(p)_{i,i} = -\sum_{j\neq i} \mathcal{L}(p)_{i,j}.$$

In the numerics, we set $m = 0.6$. This setting can be adapted to *weighted* particles $(x_i(t), \mu_i)_{i=1\cdots n}$, where each weight μ_i stands for a set of physical attributes impacting dynamics – for instance, mass – which is what we consider here. This model equivalently describes the evolution of the measure $\alpha(t) = \sum_{i=1}^n \mu_i \delta_{x_i(t)}$ in phase space $(\mathbb{R}^d)^2$, and following remarks in Section 2.2 on the ability of our architectures to model dynamical system involving interactions, (28) can be discretized in time which leads to a recurrent network making use of a single elementary block T_f between each time step. Indeed, our block allows to maintain stochasticity among all layers – which is the natural way of proceeding to follow densities of particles over time.

It is however not the purpose of this work to study such a recurrent network and we aim at showcasing here whether deep (non-recurrent) architectures of the form (20) can accurately capture the Cucker-Smale model. More precisely, since in the evolution (28) the mean of $v(t)$ stays constant, we can assume $\sum_i v_i(t) = 0$, in which case it can be shown [Cucker and Smale, 2007] that particles ultimately reach stable positions $(p(t), v(t)) \mapsto (p(\infty), 0)$. We denote $\mathcal{F}(\alpha(0)) \overset{\text{def.}}{=} \sum_{i=1}^n \mu_i \delta_{p_i(\infty)}$ the map from some initial configuration in the phase space (which is described by a probability distribution $\alpha(0)$) to the limit probability distribution (described by a discrete measure supported

on the positions $p_i(\infty)$). The goal is to approximate this map using our deep stochastic architectures. To showcase the flexibility of our approach, we consider a *non-uniform* initial measure $\alpha(0)$ and approximate its limit behavior $\mathcal{F}(\alpha(0))$ by a uniform one ($\mu_i = \frac{1}{n}$).

In our experiments, the measure $\alpha(t)$ models the dynamics of several (2 to 4) flocks of birds moving towards each other, exhibiting a limit behavior of a single stable flock. As shown in Figures 5 and 6, positions of the initial flocks are normally distributed, centered respectively at edges of a rectangle $(-4; 2), (-4; -2), (4; 2), (4; -2)$ with variance 1. Their velocities (displayed as arrows with lengths proportional to magnitudes in Figures 5 and 6) are uniformly chosen within the quarter disk $[0; -0.1] \times [0.1; 0]$. Their initial weights μ_i are normally distributed with mean 0.5 and sd 0.1, clipped by a ReLu and normalized. Figures 5 (representing densities) and 6 (depicting corresponding points' positions) show that for a set of $n = 720$ particles, quite different limit behaviors are successfully retrieved by a simple network composed of five elementary blocks with layers of dimensions $[2, 10, 20, 40, 60]$, learnt with a Wasserstein [Genevay et al., 2018] fitting criterion (computed with Sinkhorn's algorithm [Cuturi, 2013]).

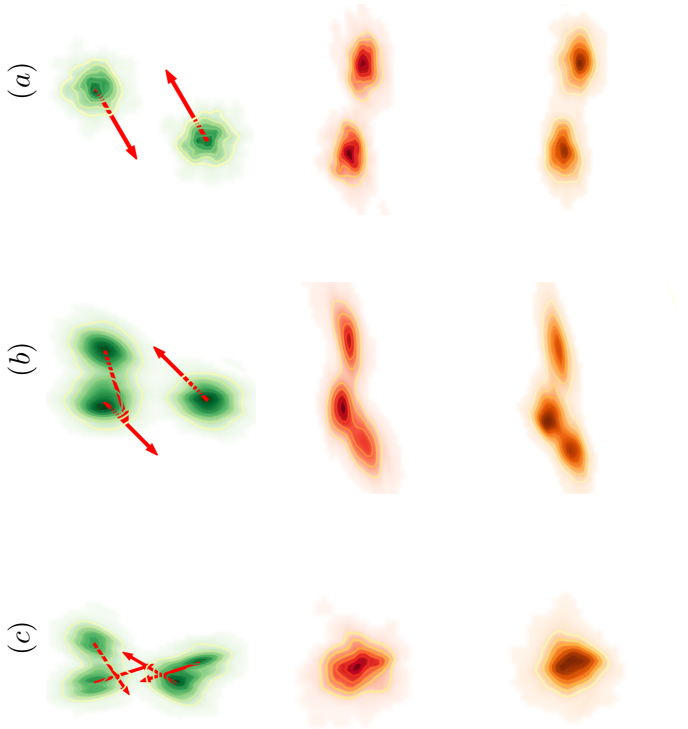


Figure 5: Prediction of the asymptotic density of the flocking model, for various initial speed values $v(0)$ and $n = 720$ particles. Eg. for top left cloud: (a) $v(0) = (0.050; -0.085)$; (b) $v(0) = (0.030; -0.094)$; (c) $v(0) = (0.056; -0.081)$.

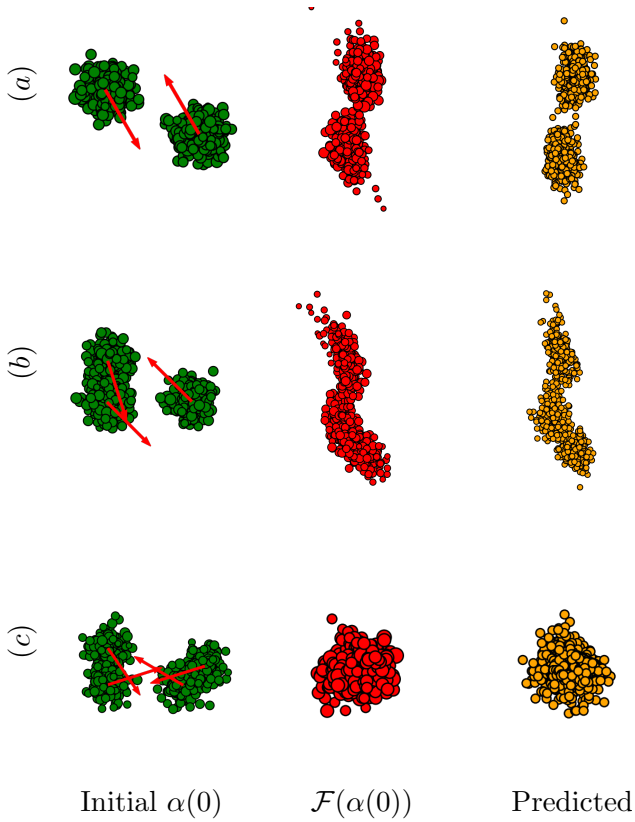


Figure 6: Prediction of particles' positions corresponding to Figure 5. Dots' diameters are proportial to weights μ_i (predicted ones all have the same size since $\mu_i = \frac{1}{n}$).

Chapter 2: Distribution-Based Invariant Deep Networks for Automated Machine Learning

Densities or probability distribution offer a propitious input or output representation for a wide variety of data types, ranging for 3D shapes in computer vision, to population modeling in biology, ecology, physics, chemistry or census. Indeed, such a design provides the means to follow populations at a macroscopic level over time without requiring individual knowledge on particles' positions, which is often inconceivable due to experimental costs or privacy concerns.

In this chapter, we demonstrate that this representation is well suited to datasets as well. Based on a set of labeled datasets, we show that their representation as measures, in a Lagrangian form, offers new perspectives to tackle the long-known problem of automated machine learning (Auto-ML), whose aim is to uncover a priori the best-performing machine learning pipeline for a task at hand.

As performance of machine learning pipelines is invariant in the ordering of dataset features as well as its labels, we introduce a neural network framework able to perform regression on probability measures, at a granular level, with such invariance requirements (referred to as invariant regression). This Distribution-based Invariant Deep Architecture (DIDA) inherits from desirable Lipschitz robustness properties and is actually a universal approximator for invariant regression functionals continuous for the convergence in law. We provide instanciations of such networks for different tasks, the end-goal being the design of expressive dataset summaries referred to as meta-features.

This chapter is based on [De Bie et al., 2020].

1. Introduction

Deep networks architectures, initially devised for structured data such as images [Krizhevsky et al., 2012] and speech [Hinton et al., 2012], have been extended to respect some invariance or equivariance [Shawe-Taylor, 1993] of more complex data types. This includes for instance point clouds [Qi et al., 2017a], graphs [Henaff et al., 2015] and probability distributions [De Bie et al., 2019], which are invariant with respect to permutations of the input points. In such cases, invariant architectures improve practical performance while inheriting the universal approximation properties of neural networks [Cybenko, 1989, Leshno et al., 1993].

In this chapter, distribution-based neural architectures [De Bie et al., 2019] are extended to cope with an additional invariance: the space of features and labels (i.e. the space supporting the distributions) is also assumed to be invariant under permutation of its coordinates. This extra invariance is important to tackle Auto-ML problems [Rice, 1976, Muñoz et al., 2018, Feurer et al., 2015, Hutter et al., 2018, Bardenet et al., 2013, Hutter et al., 2011, Klein et al., 2017, Rakotoarison et al., 2019, Elsken et al., 2019]. Auto-ML aims to identify *a priori* the machine learning configuration (both the learning algorithm and hyper-parameters thereof) best suited to the dataset under consideration in the sense of a given performance indicator. Would a dataset be associated with accurate descriptive features, referred to as meta-features, the Auto-ML problem could be handled via solving yet another supervised learning problem: given archives recording the performance of various machine learning configurations on various datasets [Vanschoren et al., 2013], with each dataset described as a vector of meta-features, the best-performing algorithm (among these configurations) on a new dataset could be predicted from its meta-features. The design of accurate meta-features however has eluded research since the 80s (with the except of [Jomaa et al., 2019], more below), to such an extent that the prominent Auto-ML approaches currently rely on learning a performance model specific to each dataset [Feurer et al., 2015, Rakotoarison et al., 2019].

Previous works.

Learning from finite discrete distributions. Learning from sets of samples subject to invariance or equivariance properties opens up a wide range of applications: in the sequence-to-sequence framework, relaxing the order in which the input is organized might be beneficial [Vinyals et al., 2016]. The ability to follow populations at a macroscopic level, using distributions

on their evolution along time without requiring to follow individual trajectories, and regardless of the population size, is appreciated when modelling dynamic cell processes [Hashimoto et al., 2016]. The use of sets of pixels, as opposed to e.g., voxellized approaches in computer vision [De Bie et al., 2019], offers a better scalability in terms of data dimensionality and computational resources.

Most generally, the fact that the considered hypothesis space and related neural architecture comply with domain-dependent invariances ensures a better robustness of the eventually learned model, better capturing the data geometry. Such neural architectures have been pioneered by [Qi et al., 2017a, Zaheer et al., 2017] for learning from point clouds subject to permutation invariance or equivariance. These have been extended to permutation equivariance across sets [Hartford et al., 2018] and relational databases [Graham et al., 2019]. Invariance or equivariance under group actions have been characterized, whether it be in the finite [Gens and Domingos, 2014, Cohen and Welling, 2016, Ravanbakhsh et al., 2017] or infinite case [Wood and Shawe-Taylor, 1996, Kondor and Trivedi, 2018]. A general identification of linear layers on the top of a representation that are invariant or equivariant with respect to the whole permutation group has been proposed by [Maron et al., 2019a, Keriven and Peyré, 2019]. Universality results are known to hold in the case of sets [Zaheer et al., 2017], point clouds [Qi et al., 2017a], equivariant point clouds [Segol and Lipman, 2020], discrete measures [De Bie et al., 2019], invariant [Maron et al., 2019b] and equivariant [Keriven and Peyré, 2019] graph neural networks. The approach most related to our work is that of [Maron et al., 2020], presenting a neural architecture invariant with respect to the ordering of samples and their features. The originality of our approach is that we do not fix in advance the number of samples, and consider probability distributions instead of point clouds. This allows us to leverage the natural topology of optimal transport to assess theoretically the universality and smoothness of our architectures, which is adapted to tackle the Auto-ML problem.

Auto-ML. The absence of learning algorithms efficient on all datasets [Wolpert, 1996] makes Auto-ML − i.e. the automatic identification of the machine learning pipelines yielding the best performance on the task at hand − a main bottleneck toward the so-called democratizing of the machine learning technology [Hutter et al., 2018]. The Auto-ML field has been sparking interest for more than four decades [Rice, 1976], spread from hyperparameter optimization [Bergstra et al., 2011] to the optimization

of the whole pipeline [Feurer et al., 2015]. Formally, Auto-ML defines a mixed integer and discrete optimization problem (finding the machine learning pipeline algorithms and their hyper-parameters), involving a black-box expensive objective function. The organization of international challenges spurred the development of various efficient Auto-ML systems, instrinsically relying on Bayesian optimization [Feurer et al., 2015, Thornton et al., 2013], Monte-Carlo tree search [Drori et al., 2018] on top of a surrogate model, or their combination [Rakotoarison et al., 2019].

As said, the ability to characterize tasks (datasets, in the remainder of this chapter) via vectors of *meta-features* would solve Auto-ML through learning the performance model. Meta-features, expected to describe the joint distribution underlying the dataset, should also be inexpensive to compute. Particular meta-features called *landmarks* [Pfahringer et al., 2000] are given by the performance of fast machine learning algorithms; indeed, knowing that a decision tree reaches a given level of accuracy on a dataset gives some information on this dataset; see also [Muñoz et al., 2018]. Another direction is explored by [Jomaa et al., 2019], defining the DATASET2VEC representation. Specifically, meta-features are extracted through solving the classification problem of whether two patches of data (subset of examples, described according to a subset of features) are extracted from the same dataset. Meta-learning [Finn et al., 2018, Yoon et al., 2018] and hyper-parameter transfer learning [Perrone et al., 2018], more remotely related to the presented approach, respectively aim to find a generic model with quick adaptability to new tasks, achieved through few-shot learning, and to transfer the performance model learned for a task, to another task.

Contributions.

The contributions of this chapter is twofold. On the algorithmic side, a *distribution-based invariant deep architecture* (DIDA) able to learn such meta-features is presented in Section 2. The challenge is that a meta-feature associated to a set of samples must be invariant both under permutation of the samples, and under permutation of their coordinates. Moreover, the architecture must be flexible enough to accept discrete distributions with diverse support and feature sizes. The proposed DIDA approach extends the state of the art [Maron et al., 2020, Jomaa et al., 2019] in two ways. Firstly, it is designed to handle discrete or continuous probability distributions, as opposed to point sets (Section 2). As said, this extension enables to leverage the more general topology of the Wasserstein distance as opposed

to that of the Haussdorf distance (Section 3). This framework is used to derive theoretical guarantees of stability under bounded distribution transformations, as well as universal approximation results, extending [Maron et al., 2020] to the continuous setting. Secondly, the empirical validation of the approach on two tasks defined at the dataset level demonstrates the merit of the approach compared to the state of the art [Maron et al., 2020, Jomaa et al., 2019, Muñoz et al., 2018] (Section 4).

2. Distribution-Based Invariant Networks for Meta-Feature Learning

This section describes our distribution-based invariant layers, mapping a probability distribution to another one while respecting invariances. It details how such layers can form trainable architectures performing regression with customized invariance requirements, referred to as *invariant regression*, and achieve meta-feature learning.

2.1. Invariant Functions of Discrete Distributions

Let $X = \{z_i \overset{\text{def.}}{=} (x_i, y_i) \in \mathbb{R}^d\}_{i=1}^n$ denote a dataset including n labelled samples, with $x_i \in \mathbb{R}^{d_X}$ an instance and $y_i \in \mathbb{R}^{d_Y}$ the associated multi-label. With d_X and d_Y respectively being the dimensions of the instance and label spaces, let $d \overset{\text{def.}}{=} d_X + d_Y$. By construction, X is invariant under permutation on the sample ordering; it is viewed as an n-size discrete distribution $\frac{1}{n}\sum_{i=1}^n \delta_{z_i}$ in $\mathbb{R}^d$ (or alternatively, as the associated random vector, hence notation X), as opposed to a point cloud. While we present in more detail the case of discrete uniform distributions, this framework is naturally suited to arbitrary distributions. Therefore, we recall for the sake of clarity that $\mathcal{M}_1^+(\mathbb{R}^d)$ still denotes the space of arbitrary distributions, whether it be continuous or discrete (of arbitrary size), and $\mathcal{R}(\mathbb{R}^d)$ still denotes the associated space of random vectors.

As the performance of a machine learning algorithm is most generally invariant with respect to permutations operating on the feature or label spaces, the neural architectures leveraged to learn the meta-features must enjoy the same property. Formally, let $G \overset{\text{def.}}{=} S_{d_X} \times S_{d_Y}$ denote the group of permutations independently operating on the feature and label spaces. For $\sigma = (\sigma_X, \sigma_Y) \in G$, the image $\sigma(X)$ of a labelled sample is defined as $(\sigma_X(x), \sigma_Y(y))$, with $x = (x[k])_{k=1}^{d_X}$ and $\sigma_X(x) \overset{\text{def.}}{=} (x[\sigma_X^{-1}(k)])_k$. For simplicity and by abuse of notations, the operator mapping a distribution

$X = (z_i)_i$ to $\{\sigma(z_i)\} \stackrel{\text{def.}}{=} \sigma_\sharp X$ is still denoted σ. We denote $\mathcal{M}_1^+(\Omega)$ the space of distributions supported on some set $\Omega \subset \mathbb{R}^d$ (respectively, $\mathcal{R}(\Omega)$ the space of random vectors), and we assume that the domain Ω is invariant under permutations in G.

The goal of this chapter is to define trainable deep architectures, implementing functions φ defined on $\mathcal{R}(\Omega \subset \mathbb{R}^d)$ such that these are invariant under G, i.e. $\varphi(\sigma_\sharp X) = \varphi(X)$ for any $\sigma \in G$. By construction, a multi-label dataset is invariant under permutations of the samples, of the features, and of the multi-labels. Therefore, any meta-feature, that is, a feature describing a multi-label dataset, is required to satisfy the above sample and feature permutation invariance properties. Such functions will be trained to define meta-features.

2.2. Distribution-Based Invariant Layers

Taking inspiration from [De Bie et al., 2019], the basic building-blocks of the neural architecture defined in Section 2.1 of Chapter 1 are extended to satisfy the feature- and label-invariance requirements.

Definition 5. *(Distribution-based invariant layers) Let an interaction functional $\varphi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^r$ be G-invariant, i.e.*

$$\forall \sigma \in G, \quad \forall (z, z') \in (\mathbb{R}^d)^2, \quad \varphi(z, z') = \varphi(\sigma(z), \sigma(z')). \tag{29}$$

A distribution-based invariant layer T_φ is defined as

$$T_\varphi : X \in \mathcal{R}(\mathbb{R}^d) \mapsto \mathbb{E}_{X' \sim X}[\varphi(X, X')] \in \mathcal{R}(\mathbb{R}^r) \tag{30}$$

where X' is a random vector independent of X having the same distribution.

Remark 2. It is easy to see that, defined as such, $T_\varphi : \mathcal{R}(\mathbb{R}^d) \to \mathcal{R}(\mathbb{R}^r)$ is indeed invariant.

Nature of the invariance. Note that the invariance requirement on φ actually is less demanding than requiring $\varphi(z, z') = \varphi(\sigma(z), \tau(z'))$ for any two distinct permutations σ and τ in G.

Discrete distribution. In the experiments, datasets are represented as random vectors uniformly distributed on a set $(z_i)_{i=1}^n$, in which case the invariant layer T_φ maps $X = (z_i)_{i=1}^n \in \mathcal{R}(\mathbb{R}^d)$ to

$$T_\varphi(X) \stackrel{\text{def.}}{=} \left(\frac{1}{n} \sum_{j=1}^n \varphi(z_1, z_j), \ldots, \frac{1}{n} \sum_{j=1}^n \varphi(z_n, z_j) \right) \in \mathcal{R}(\mathbb{R}^r).$$

Moment and Push-forward. Two particular cases are when φ only depends on its first or second input:

(i) if $\varphi(z, z') = \psi(z')$, then T_φ computes a global "moment" descriptor of the input, as $T_\varphi(X) = \mathbb{E}_X[\psi(X)]$, which, in the discrete case, reads $\frac{1}{n}\sum_{j=1}^n \psi(z_j) \in \mathbb{R}^r$.

(ii) if $\varphi(z, z') = \xi(z)$, then T_φ transports the input distribution via ξ through a push-forward, which, in the case of discrete distributions, reads $T_\varphi(X) = \{\xi(z_i)\}_i \subset \mathbb{R}^r$.

Spaces of arbitrary dimension. Both in practice and in theory, it is important to define T_φ layers (in particular the first one of the architecture) that can be applied to distributions on $\mathbb{R}^{d_X} \times \mathbb{R}^{d_Y}$ of arbitrary dimensions d_X and d_Y. This can be achieved by constraining φ to be of the form, with $z = (x, y)$ and $z' = (x', y')$:

$$\varphi(z, z') = v\left(\sum_{k=1}^{d_X}\sum_{\ell=1}^{d_Y} u(x[k], x'[\ell], y[k], y'[\ell])\right)$$

where $u : \mathbb{R}^4 \to \mathbb{R}^t$ and $v : \mathbb{R}^t \to \mathbb{R}^r$ are independent of d.

Generalization to arbitrary groups. The definition of invariant functions φ (and the corresponding architectures) can be generalized to arbitrary group operating on $\mathbb{R}^d$ (in particular sub-groups of the permutation group). A simple way to design an invariant function is to consider $\varphi(z, z') = \psi(z+z')$ where ψ is G-invariant. In the linear case, [Maron et al., 2020], Theorem 5 shows that these types of functions are the only ones, but this is not anymore true for non-linear functions.

Localized computation. The complexity of computing $\frac{1}{n}\sum_j \varphi(z_i, z_j)$ in practice can be reduced by considering only z_j in a neighborhood of z_i. The layer then extracts local information around each of the points.

Higher Order Interactions and Tensorization. Invariant layers can also be generalized to handle higher order interactions functionals, namely $T_\varphi(X) \stackrel{\text{def.}}{=} \mathbb{E}_{X_2,\ldots,X_N\sim X}[\varphi(X, X_2, \ldots, X_N)]$. An equivalent and elegant way to introduce these interactions in a deep architecture is by adding a tensorization layer, which maps $X \mapsto X_2 \otimes \ldots \otimes X_N \in \mathcal{R}((\mathbb{R}^d)^{N-1})$. Section 3 details the regularity and approximation power of these tensorization steps.

Link to kernel methods. The use of an interaction functional φ is inspired from kernel ideas, albeit with significant differences: (i) using T_φ, the detail of the pairwise interactions $\varphi(z_i, z_j)$ is lost through averaging; (ii) φ takes into account labels; (iii) φ is learnt. Further work will be devoted to investigating the properties of the $T_\varphi(z_i)$ matrix.

2.3. Learning Dataset Meta-features from Distributions

The proposed *invariant regression* neural architectures defined on point distributions (DIDA) are defined as

$$X \in \mathcal{R}(\mathbb{R}^d) \mapsto \mathcal{F}_\zeta(X) \stackrel{\text{def.}}{=} f_{\varphi_m} \circ f_{\varphi_{m-1}} \circ \ldots \circ f_{\varphi_1}(X) \in \mathbb{R}^{d_{m+1}} \tag{31}$$

where ζ are the trainable parameters of the architecture (detailed below). Note that this architecture shares similarities to the one presented in Section 2.2 of Chapter 1, however focused on the discriminative case (with a constant output vector), and including an additional invariance requirement. Here $\varphi_k : \mathbb{R}^{d_k} \times \mathbb{R}^{d_k} \to \mathbb{R}^{d_{k+1}}$, $d_1 = d$ and φ_m only depends on its second argument, such that $\mathcal{F}_\zeta(X) \in \mathbb{R}^{d_{m+1}}$ should be understood as being a vector (as opposed to a distribution), whose coordinates are referred to as *meta-features*.

Note that only φ_1 is required to be G-invariant and dimension-agnostic for the architecture to be as well. This map φ_1, defined as suggested in Section 2.2, is thus learned using inputs of varying dimension as a G-invariant layer with $d_Y = 1$, where u maps $(x, x', y, y') \in \mathbb{R}^4$ to $[\rho(A_u[x; x'] + b_u); \mathbb{1}_{y \neq y'}] \in \mathbb{R}^t$, v maps $e \in \mathbb{R}^t$ to $\rho(A_v e + b_v) \in \mathbb{R}^r$, with $A_u \cdot + b_u$, $A_v \cdot + b_v$ are affine functions, ρ is a non-linearity and $[.; .]$ denotes concatenation.

As the following layers φ_k $(k = 2, \ldots, m)$ need not be invariant, they are parameterized as $\varphi_k = \rho(A_k \cdot + b_k)$ using a pair A_k, b_k of (matrix,vector). The parameters of the DIDA architecture are thus $\zeta \stackrel{\text{def.}}{=} (A_u, b_u, A_v, b_v, \{A_k, b_k\}_k)$. They are learned in a supervised fashion, with a loss function depending on the task at hand (see Section 4). Maintaining the distributional nature among several layers is shown to improve performance in practice (see Section 4). By construction, these architectures are invariant with respect to the orderings of both the points composing the input distributions and their coordinates. The input distributions can be composed of any number of points in any dimension, which is a distinctive feature with respect to [Maron et al., 2020].

3. Theoretical Analysis

To get some insight on these architectures, we now detail their robustness to perturbations and their approximation abilities with respect to the convergence in law, which is the natural topology for distributions.

3.1. Optimal Transport Comparison of Datasets

Point clouds vs. distributions. It is important to note that learning from datasets, referred to as *meta-learning* for simplicity in the sequel, requires such datasets be seen as probability distributions, as opposed to point clouds. For instance, having twice the same point in a dataset really corresponds to doubling its mass, i.e. it should have twice more importance than the other points. We thus argue that the natural topology to analyze meta-learning methods is the one of the convergence in law, which can be quantified using Wasserstein optimal transport distances. This is in sharp contrast with point clouds architectures (see for instance [Qi et al., 2017a]), making use of max-pooling and relying on the Haussdorff distance to analyze the architecture properties. While this analysis is standard for low-dimensional (2D and 3D) applications in graphics and vision, this is not suitable for our purpose, because max-pooling is not a continuous operation for the topology of convergence in law.

Wasserstein distance. In order to quantify the regularity of the involved functionals, we resort to the 1-Wasserstein distance between two probability distributions $\alpha, \beta \in (\mathcal{M}_1^+(\mathbb{R}^d))^2$ (referring the reader to [Santambrogio, 2015, Peyré and Cuturi, 2019] for a comprehensive presentation of Wasserstein distance):

$$\mathrm{W}_1(\alpha,\beta) \stackrel{\text{def.}}{=} \min_{\pi_1=\alpha,\pi_2=\beta} \int_{\mathbb{R}^d\times\mathbb{R}^d} \|x-y\| \mathrm{d}\pi(x,y) \stackrel{\text{def.}}{=} \min_{X\sim\alpha,Y\sim\beta} \mathbb{E}(\|X-Y\|)$$

where the minimum is taken over measures on $\mathbb{R}^d \times \mathbb{R}^d$ with marginals $\alpha, \beta \in \mathcal{M}_1^+(\mathbb{R}^d)$. W_1 is known to be a norm [Santambrogio, 2015], that can be conveniently computed using

$$\mathrm{W}_1(\alpha,\beta) = \mathrm{W}_1(\alpha-\beta) = \max_{\mathrm{Lip}(g)\leqslant 1} \int_{\mathbb{R}^d} g \mathrm{d}(\alpha-\beta),$$

where $\mathrm{Lip}(g)$ is the Lipschitz constant of $g : \mathbb{R}^d \to \mathbb{R}$ with respect to the Euclidean norm (unless otherwise stated). For simplicity and by abuse of notations, $\mathrm{W}_1(X,Y)$ is used instead of $\mathrm{W}_1(\alpha,\beta)$ when $X \sim \alpha$ and $Y \sim \beta$.

The convergence in law denoted $\rightharpoonup$ is equivalent to the convergence in Wasserstein distance in the sense that $X_k \rightharpoonup X$ is equivalent to $\mathrm{W}_1(X_k, X) \to 0$.

Permutation-invariant Wasserstein distance. The Wasserstein distance is quotiented according to the permutation-invariance equivalence classes: for $\alpha, \beta \in \mathcal{M}_1^+(\mathbb{R}^d)$

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

such that $\overline{\mathrm{W}}_1(\alpha, \beta) = 0 \iff \alpha \sim \beta$. $\overline{\mathrm{W}}_1$ defines a norm on the quotient space $\mathcal{M}_1^+(\mathbb{R}^d)_{/\sim}$, which is, for the sake of simplicity, still denoted $\mathcal{M}_1^+(\mathbb{R}^d)$ or $\mathcal{R}(\mathbb{R}^d)$ in the following.

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

$$\forall\, (X, Y) \in (\mathcal{R}(\mathbb{R}^d))^2,\ \overline{\mathrm{W}}_1(f(X), f(Y)) \leqslant C \overline{\mathrm{W}}_1(X, Y). \tag{32}$$

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.

3.2. Regularity of Distribution-Based Invariant Layers

The following propositions show the robustness of invariant layers with respect to different variations of their input, assuming the following regularity condition on the interaction functional:

$$\forall z \in \mathbb{R}^d, \quad \varphi(z, \cdot) \quad \text{and} \quad \varphi(\cdot, z) \quad \text{are} \quad \mathrm{Lip}(\varphi) - \text{Lipschitz}. \tag{33}$$

We first show that invariant layers are Lipschitz regular. This ensures that deep architectures of the form (31) map close inputs onto close outputs.

Proposition 7. *Invariant layers T_φ of type* (30) *are* $(2r\,\mathrm{Lip}(\varphi))$*-Lipschitz in the sense of* (32).

Proof. (Proposition 7). For $\alpha, \beta \in \mathcal{M}_1^+(\mathbb{R}^d)$, Proposition 1 from [De Bie et al., 2019] yields $\mathrm{W}_1(T_\varphi(\alpha), T_\varphi(\beta)) \leqslant 2r \operatorname{Lip}(\varphi) \mathrm{W}_1(\alpha, \beta)$, hence, for $\sigma \in G$,

$$\begin{aligned}\mathrm{W}_1(\sigma_\sharp T_\varphi(\alpha), T_\varphi(\beta)) &\leqslant \mathrm{W}_1(\sigma_\sharp T_\varphi(\alpha), T_\varphi(\alpha)) + \mathrm{W}_1(T_\varphi(\alpha), T_\varphi(\beta)) \\ &\leqslant \mathrm{W}_1(\sigma_\sharp T_\varphi(\alpha), T_\varphi(\alpha)) + 2r \operatorname{Lip}(\varphi) \mathrm{W}_1(\alpha, \beta)\end{aligned}$$

hence, taking the infimum over σ yields

$$\begin{aligned}\overline{\mathrm{W}}_1(T_\varphi(\alpha), T_\varphi(\beta)) &\leqslant \overline{\mathrm{W}}_1(T_\varphi(\alpha), T_\varphi(\alpha)) + 2r \operatorname{Lip}(\varphi) \mathrm{W}_1(\alpha, \beta) \\ &\leqslant 2r \operatorname{Lip}(\varphi) \mathrm{W}_1(\alpha, \beta)\end{aligned}$$

Since T_φ is invariant, for $\sigma \in G$, $T_\varphi(X) = T_\varphi(\sigma_\sharp X)$,

$$\overline{\mathrm{W}}_1(T_\varphi(\alpha), T_\varphi(\beta)) \leqslant 2r \operatorname{Lip}(\varphi) \mathrm{W}_1(\sigma_\sharp \alpha, \beta)$$

Taking the infimum over σ yields the result. □

Secondly, we consider perturbations with respect to diffeomorphisms. This stability is essential to cope with situations where, for instance, an auto-encoder τ has been trained, so that a dataset $X = (z_1, \ldots, z_n)$ and its encoded-decoded representation $\tau_\sharp X = (\tau(z_1), \ldots, \tau(z_n))$ are expected to yield similar meta-features. The following proposition shows that $T_\varphi(\tau_\sharp X)$ and $T_\varphi(X)$ are indeed close if τ is close to the identity, which is expected when using auto-encoders. It also shows that similarly, if both inputs and outputs are modified by regular deformations τ and ξ, then the output are also close.

Proposition 8. *For $\tau : \mathbb{R}^d \to \mathbb{R}^d$ and $\xi : \mathbb{R}^r \to \mathbb{R}^r$ two Lipschitz maps, one has, for all $\alpha, \beta \in \mathcal{M}_1^+(\Omega)$,*

$$\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha)) \leqslant \sup_{x \in T_\varphi(\tau(\Omega))} \|\xi(x) - x\|_2 + 2r \operatorname{Lip}(\varphi) \sup_{x \in \Omega} \|\tau(x) - x\|_2$$

Also, if τ is equivariant, the following holds:

$$\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) \leqslant 2r \operatorname{Lip}(\varphi) \operatorname{Lip}(\tau) \operatorname{Lip}(\xi) \overline{\mathrm{W}}_1(\alpha, \beta)$$

Proof. (Proposition 8). To upper bound $\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha))$ for $\alpha \in \mathcal{M}_1^+(\Omega)$, we proceed as follows, using proposition 3 from [De Bie et al., 2019] and proposition 7:

$$\begin{aligned}\mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha_\varphi(\alpha)), T_\varphi(\alpha)) &\leqslant \mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\tau_\sharp \alpha)) + \mathrm{W}_1(T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha)) \\ &\leqslant \|\xi - id\|_{L^1(T_\varphi(\tau_\sharp \alpha))} + \operatorname{Lip}(T_\varphi) \mathrm{W}_1(\tau_\sharp \alpha, \alpha) \\ &\leqslant \sup_{y \in T_\varphi(\tau(\Omega))} \|\xi(y) - y\|_2 + 2r \operatorname{Lip}(\varphi) \sup_{x \in \Omega} \|\tau(x) - x\|_2\end{aligned}$$

For $\sigma \in G$, we get

$$\begin{aligned}\mathrm{W}_1(\sigma_\sharp \xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha)) &\leqslant \mathrm{W}_1(\sigma_\sharp \xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \alpha)) \\ &\quad + \mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha))\end{aligned}$$

Taking the infimum over σ yields

$$\begin{aligned}\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha)) &\leqslant \mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), T_\varphi(\alpha)) \\ &\leqslant \sup_{y \in T_\varphi(\tau(\Omega))} \|\xi(y) - y\|_2 + 2r \operatorname{Lip}(\varphi) \sup_{x \in \Omega} \|\tau(x) - x\|_2\end{aligned}$$

which is the expected result. Similarly, for $\alpha, \beta \in (\mathcal{M}_1^+(\mathbb{R}^d))^2$,

$$\begin{aligned}\mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) &\leqslant \operatorname{Lip}(\xi)\, \mathrm{W}_1(T_\varphi(\tau_\sharp \alpha), T_\varphi(\tau_\sharp \beta)) \\ &\leqslant \operatorname{Lip}(\xi) \operatorname{Lip}(T_\varphi)\, \mathrm{W}_1(\tau_\sharp \alpha, \tau_\sharp \beta) \\ &\leqslant 2r \operatorname{Lip}(\varphi) \operatorname{Lip}(\xi) \operatorname{Lip}(\tau)\, \mathrm{W}_1(\alpha, \beta)\end{aligned}$$

hence, for $\sigma \in G$,

$$\begin{aligned}\mathrm{W}_1(\sigma_\sharp \xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) &\leqslant \mathrm{W}_1(\sigma_\sharp \xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \alpha)) \\ &\quad + \mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta))\end{aligned}$$

and taking the infimum over σ yields

$$\begin{aligned}\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) &\leqslant \mathrm{W}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) \\ &\leqslant 2r \operatorname{Lip}(\varphi) \operatorname{Lip}(\xi) \operatorname{Lip}(\tau)\, \mathrm{W}_1(\alpha, \beta)\end{aligned}$$

Since τ is equivariant: namely, for $\alpha \in \mathcal{M}_1^+(\mathbb{R}^d)$, $\sigma \in G, \tau_\sharp(\sigma_\sharp \alpha) = \sigma_\sharp(\tau_\sharp \alpha)$, hence, since T_φ is invariant, $T_\varphi(\tau_\sharp(\sigma_\sharp \alpha)) = T_\varphi(\sigma_\sharp(\tau_\sharp \alpha)) = T_\varphi(\tau_\sharp \alpha)$, hence for $\sigma \in G$,

$$\overline{\mathrm{W}}_1(\xi_\sharp T_\varphi(\tau_\sharp \alpha), \xi_\sharp T_\varphi(\tau_\sharp \beta)) \leqslant 2r \operatorname{Lip}(\varphi) \operatorname{Lip}(\xi) \operatorname{Lip}(\tau)\, \mathrm{W}_1(\sigma_\sharp \alpha, \beta)$$

Taking the infimum over σ yields the result. □

3.3. Universality of Invariant Layers

We now show that our architecture can approximate any continuous invariant map. More precisely, the following proposition shows that the combination of an invariant layer (30) and a fully-connected layer are enough to reach universal approximation capability. This statement holds for arbitrary distributions (not necessarily discrete) and for functions defined on spaces of arbitrary dimension in the sense of Section 2.2 (assuming some a priori bound on the dimensions).

Theorem 5. *Let $\mathcal{F} : \mathcal{M}_1^+(\Omega) \to \mathbb{R}$ a $S_{d_X} \times S_{d_Y}$-invariant map continuous for the convergence in law, where Ω is compact. Then $\forall \eta > 0$, there exists two continuous maps ψ, φ such that*

$$\forall \alpha \in \mathcal{M}_1^+(\Omega), \quad |\mathcal{F}(\alpha) - \psi \circ T_\varphi(\alpha)| < \eta$$

where φ is $S_{d_X} \times S_{d_Y}$-invariant and independent of $\mathcal{F}$.

Before providing a proof of Theorem 5, we first state two Lemmas that will be useful for the proof.

Lemma 3. *Let $(S_j)_{j=1}^N$ be a partition of a domain including Ω ($S_j \subset \mathbb{R}^d$) and let $x_j \in S_j$. Let $(\varphi_j)_{j=1}^N$ a set of bounded functions $\varphi_j : \Omega \to \mathbb{R}$ supported on S_j, such that $\sum_j \varphi_j = 1$ on Ω. For $\alpha \in \mathcal{M}_1^+(\Omega)$, we denote $\hat{\alpha}_N \overset{\text{def.}}{=} \sum_{j=1}^N \alpha_j \delta_{x_j}$ with $\alpha_j \overset{\text{def.}}{=} \int_{S_j} \varphi_j \mathrm{d}\alpha$. One has, denoting $\Delta_j \overset{\text{def.}}{=} \max_{x \in S_j} \|x_j - x\|$,*

$$\mathrm{W}_1(\hat{\alpha}_N, \alpha) \leqslant \max_{1 \leqslant j \leqslant N} \Delta_j.$$

Proof. We refer to Chapter 1, Section 3.3 for a proof. □

Lemma 4. *Let $f : \mathbb{R}^d \to \mathbb{R}^q$ a $1/p$-Hölder continuous function ($p \geqslant 1$), then there exists a constant $C > 0$ such that for all $\alpha, \beta \in \mathcal{M}_1^+(\mathbb{R}^d)$, $\mathrm{W}_1(f_\sharp \alpha, f_\sharp \beta) \leqslant C\, \mathrm{W}_1(\alpha, \beta)^{1/p}$.*

Proof. For any transport map π with marginals α and β, $1/p$-Hölderness of f with constant C yields $\int ||f(x) - f(y)||_2 \mathrm{d}\pi(x,y) \leqslant C \int ||x-y||_2^{1/p} \mathrm{d}\pi(x,y) \leqslant C \left(\int ||x-y||_2 \mathrm{d}\pi(x,y)\right)^{1/p}$ using Jensen's inequality ($p \leqslant 1$). Taking the infimum over π yields $\mathrm{W}_1(f_\sharp \alpha, f_\sharp \beta) \leqslant C\, \mathrm{W}_1(\alpha, \beta)^{1/p}$. □

We are now ready to provide a proof of Theorem 5. We first show the result in the case of S_d-invariant regression functionals ($G = S_d$) and extend the result to products of permutations ($G = S_{d_1} \times \ldots S_{d_N}$) in the next paragraph.

Proof. Let $\alpha \in \mathcal{M}_1^+(\mathbb{R}^d)$. We consider:

(i) $h : x = (x_1, \ldots, x_d) \in \mathbb{R}^d \mapsto \left(\sum_{1 \leqslant j_1 < \ldots < j_i \leqslant d} x_{j_1} \cdot \ldots \cdot x_{j_i}\right)_{i=1\ldots d} \in \mathbb{R}^d$ 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]);

(ii) h is obviously not injective, so we consider $\pi : \mathbb{R}^d \to \mathbb{R}^d/S_d$ 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;

(iii) 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 $g : x \in \mathbb{R}^d \mapsto (\varphi_1(x), \dots, \varphi_N(x)) \in \mathbb{R}^N$; φ introduced in the statement of Theorem 5 is defined as $\varphi \stackrel{\text{def.}}{=} g \circ h$;

(iv) $\psi : (\alpha_1, \dots, \alpha_N) \in \mathbb{R}^N \mapsto \mathcal{F}\left(\sum_{i=1}^N \alpha_i \delta_{\tilde{h}^{-1}(y_i)}\right) \in \mathbb{R}$.

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

(i) Lemma 3 yields that $h_\sharp\alpha$ and $\widehat{h_\sharp\alpha} = \sum_{i=1}^N \alpha_i \delta_{y_i}$ are close:

$$\mathrm{W}_1(h_\sharp\alpha, \widehat{h_\sharp\alpha}) \leqslant \sqrt{d}/N^{1/d}$$

(ii) The map $\tilde{h}^{-1}$ is regular enough ($1/d$-Hölder) such that according to Lemma 4, there exists a constant $C > 0$ such that

$$\mathrm{W}_1(\tilde{h}_\sharp^{-1}(h_\sharp\alpha), \tilde{h}_\sharp^{-1}\widehat{h_\sharp\alpha}) \leqslant C\,\mathrm{W}_1(h_\sharp\alpha, \widehat{h_\sharp\alpha})^{1/d} \leqslant Cd^{1/2d}/N^{1/d^2}$$

Hence $\overline{\mathrm{W}}_1(\alpha, \tilde{h}_\sharp^{-1}\widehat{h_\sharp\alpha}) := \inf_{\sigma\in S_d} \mathrm{W}_1(\sigma_\sharp\alpha, \tilde{h}_\sharp^{-1}\widehat{h_\sharp\alpha}) \leqslant Cd^{1/2d}/N^{1/d^2}$.

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 and Schmeisser, 2002] yields continuity and $1/d$-Hölderness of the reverse map. Hence $\tilde{h}^{-1}$ is $1/d$-Hölder.

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

□

It is worth noting that, contrary to the proof of Theorem 3 in Chapter 1, which considers measure-valued functionals, the concatenation of random noise is not required here. Another distinctive feature is the use of elementary symmetric polynomials that enforce here the desired invariance property.

Extension to products of permutation groups. The approximation ability of such layers extends to products of permutation groups, which is our experimental setting (see Section 4), as exemplified in the next corollary.

Corollary 1. *Let $\mathcal{F} : \mathcal{M}_1^+(\Omega) \to \mathbb{R}$ a continuous $S_{d_1} \times \ldots \times S_{d_n}$-invariant map ($\sum_i d_i = d$), where Ω is a symmetrized compact over $\mathbb{R}^d$. Then $\forall \eta > 0$, there exists two continuous maps ψ, φ such that*

$$\forall \alpha \in \mathcal{M}_+^1(\Omega), |\mathcal{F}(\alpha) - \psi \circ T_\varphi(\alpha)| < \eta$$

where φ is $S_{d_1} \times \ldots \times S_{d_n}$-invariant and 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 $G = S_{d_1} \times \ldots \times S_{d_n}$. We trade h in the proof of Theorem 5 for the collection of elementary symmetric polynomials in the first d variables; and in the last p variables: $h : (x_1, \ldots, x_d, y_1, \ldots, y_p) \in \mathbb{R}^{d+p} \mapsto ([\sum_{1 \leqslant j_1 < \ldots < j_i \leqslant d} x_{j_1} \ldots x_{j_i}]_{i=1}^d; [\sum_{1 \leqslant j_1 < \ldots < j_i \leqslant p} y_{j_1} \ldots y_{j_i}]_{i=1}^p) \in \mathbb{R}^{d+p}$ up to normalizing constants (see Lemma 6). We still define $\varphi \stackrel{\text{def.}}{=} g \circ h$, with $g : \mathbb{R}^{d+p} \to \mathbb{R}^N$, and keep the same ψ. Step 1 (in Lemma 5) 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 6), we show that $\tilde{h}^{-1}$ is $1/\max(d,p)-$Hölder. Combined with the proof of Theorem 5, 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. □

Lemma 5. *Let the collection of symmetric invariant polynomials*

$$[P_i(X_1, \ldots, X_d)]_{i=1}^d \stackrel{\text{def.}}{=} [\sum_{1 \leqslant j_1 < \ldots < j_i \leqslant d} X_{j_1} \ldots X_{j_i}]_{i=1}^d$$

and

$$[Q_i(Y_1, \ldots, Y_p)]_{i=1}^p \stackrel{\text{def.}}{=} [\sum_{1 \leqslant j_1 < \ldots < j_i \leqslant p} Y_{j_1} \ldots Y_{j_i}]_{i=1}^p$$

The $d+p-$sized family $(P_1, \ldots, P_d, Q_1, \ldots, Q_p)$ 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 $P(X_1, \ldots, X_d, Y_1, \ldots, Y_p)$ is also $S_d \times I_p-$invariant with coefficients in $\mathbb{R}[Y_1, \ldots, Y_p]$, 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 $P = S_{(Q_1,\dots,Q_p)}(P_1,\dots,P_d) \in \mathbb{R}[P_1,\dots,P_d,$
$Q_1,\dots,Q_p]$. □

Lemma 6. *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 Ω is compact, $f : \mathbb{R}^d \to \mathbb{R}^d$ is $1/d-$Hölder with constant C_1 and $g : \mathbb{R}^p \to \mathbb{R}^p$ 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. $\forall x, x_0 \in (\mathbb{R}^d)^2, \|f(x) - f(x_0)\|_1 \leqslant \|x - x_0\|_1^{1/d}$). For $(x,y),(x_0,y_0) \in \Omega^2$,

$$\begin{aligned}\|h(x,y) - h(x_0,y_0)\|_1 &\leqslant \|f(x) - f(x_0)\|_1 + \|g(y) - g(y_0)\|_1 \\ &\leqslant \|x - x_0\|_1^{1/d} + \|y - y_0\|_1^{1/p}\end{aligned}$$

since both f, g are Hölder. We denote D the diameter of Ω, such that both $\|x - x_0\|_1/D \leqslant 1$ and $\|y - y_0\|_1/D \leqslant 1$ hold. Therefore

$$\begin{aligned}\|h(x,y) - h(x_0,y_0)\|_1 &\leqslant D^{1/d}\left(\frac{\|x - x_0\|_1}{D}\right)^{1/d} + D^{1/p}\left(\frac{\|y - y_0\|_1}{D}\right)^{1/p} \\ &\leqslant 2^{1-1/d} D^{1/p-1/d} \|(x,y) - (x_0,y_0)\|_1^{1/d}\end{aligned}$$

using Jensen's inequality, hence the result. □

Extension to different spaces. Theorem 5 also extends to distributions supported on different spaces, by considering a joint embedding space of large enough dimension. This way, any invariant prediction function can (uniformly) be approximated by an invariant network, up to setting added coordinates to zero, as shown below.

Corollary 2. *Let $I = [0;1]$ and, for $k \in [1; d_m], \mathcal{F}_k : \mathcal{M}_1^+(I^k) \to \mathbb{R}$ continuous and S_k-invariant. Suppose $(\mathcal{F}_k)_{k=1\dots d_m-1}$ are restrictions of $\mathcal{F}_{d_m}$, namely, $\forall \alpha_k \in \mathcal{M}_1^+(I^k), \mathcal{F}_k(\alpha_k) = \mathcal{F}_{d_m}(\alpha_k \otimes \delta_0^{\otimes d_m - k})$. Then there exists ψ, g continuous, $h_1,\dots,h_{d_m}$ continuous invariant such that*

$$\forall k = 1 \dots d_m, \forall \alpha_k \in \mathcal{M}_1^+(I^k), |\mathcal{F}_k(\alpha_k) - \psi \circ \mathbb{E} \circ g(h_{k\sharp}\alpha_k)| < \eta.$$

Proof. The proof of Theorem 5 yields continuous ψ, g and a continuous invariant h_{d_m} such that $\forall \alpha \in \mathcal{M}_1^+(I^{d_m}), |\mathcal{F}_{d_m} - \psi \circ \mathbb{E} \circ g(h_{d_m\sharp}\alpha)| < \eta$ (with φ in the

statement of the Theorem defined as $\varphi \overset{\text{def.}}{=} g \circ h_{d_m}$). For $k = 1 \dots d_m - 1$, we denote $h_k : (x_1, \dots, x_k) \in \mathbb{R}^k \mapsto \left[(\sum_{1 \leqslant j_1 < \dots < j_i \leqslant k} x^{(j_1)} \cdot \dots \cdot x^{(j_i)})_{i=1 \dots k}, 0 \dots, 0 \right] \in \mathbb{R}^{d_m}$. With the hypothesis, for $k = 1 \dots d_m - 1$, $\alpha_k \in \mathcal{M}_1^+(I^k)$, the fact that $h_{k\sharp}(\alpha_k) = h_{d_m \sharp}(\alpha_k \otimes \delta_0^{\otimes d_m - k})$ yields the result. □

Approximation by invariant neural networks. A consequence of Theorem 5 is that any continuous invariant regression function taking (compactly supported) distributions can be approximated to arbitrary precision by an invariant neural network. This result is detailed below and uses the following ingredients: (i) an invariant layer with φ that can be approximated by an invariant network; (ii) the universal approximation theorem [Cybenko, 1989, Leshno et al., 1993]; (iii) uniform continuity to obtain uniform bounds.

Proof. (Approximation by neural networks related to Theorem 5) Based on the proof of Theorem 5, $\mathcal{F}$ is uniformly close to $\psi \circ \mathbb{E} \circ g \circ h$, where φ in the statement of the Theorem is defined as $\varphi \overset{\text{def.}}{=} g \circ h$:

(i) 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 λ is a non-linearity.

(ii) Since each component φ_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), $\rho_j : \mathbb{R}^{d+1} \to \mathbb{R}$, and $u : \mathbb{R} \to \mathbb{R}^{d+1}$ independent of j (see [Zaheer et al., 2017], theorem 7).

(iii) We can approximate ρ_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 $\varphi_\theta(x) = (\varphi_{j,\theta}(x))_j \overset{\text{def.}}{=} (\rho_{j,\theta}(\sum_{i=1}^d u_\theta(x_i)))_j$.

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

(i) $|\mathcal{F}(\alpha) - f \left(\mathbb{E}_{X \sim \alpha} \left(\varphi(X) \right) \right)|$

(ii) $|f \left(\mathbb{E}_{X \sim \alpha} \left(\varphi(X) \right) \right) - f_\theta \left(\mathbb{E}_{X \sim \alpha} \left(\varphi(X) \right) \right)|$

(iii) $|f_\theta \left(\mathbb{E}_{X \sim \alpha} \left(\varphi(X) \right) \right) - f_\theta \left(\mathbb{E}_{X \sim \alpha} \left(\varphi_\theta(X) \right) \right)|$

and bound each term by $\frac{\varepsilon}{3}$, which yields the result. The bound on the first term directly comes from theorem 5 and yields a constant N which depends on ε. The bound on the second term is a direct application of the universal approximation theorem (UAT) [Cybenko, 1989, Leshno et al., 1993]. Indeed, since α is a probability measure, input values of f lie in a compact subset of $\mathbb{R}^N$: $\left\|\int_\Omega g \circ h(x)\mathrm{d}\alpha\right\|_\infty \leqslant \max_{x\in\Omega}\max_i |g_i \circ h(x)|$, hence the theorem is applicable as long as λ is a nonconstant, bounded and continuous activation function. Let us focus on the third term. Uniform continuity of f_θ 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 φ_j of h can be approximated by a neural network $\varphi_{j,\theta}$. Therefore:

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

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

Universality of Tensorization. As hinted at in Section 2.2, tensor products play a role in designing invariant layers, allowing for more expressive power as illstrated in the following result. Indeed, as long as the test function is invariant, tensorization allows for the approximation of any invariant regression functional.

Theorem 6. *The algebra*

$$\mathcal{A}_\Omega \overset{\text{def.}}{=} \left\{ \begin{array}{r} \mathcal{F} : \mathcal{M}_1^+(\Omega) \to \mathbb{R}, \exists n \in \mathbb{N}, \exists \varphi : \Omega^n \to \mathbb{R} \text{ invariant}, \\ \forall \alpha \in \mathcal{M}_1^+(\Omega), \mathcal{F}(\alpha) = \displaystyle\int_{\Omega^n} \varphi \mathrm{d}\alpha^{\otimes n} \end{array} \right\}.$$

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 Ω is compact, by Banach-Alaoglu theorem, we obtain that $\mathcal{M}_1^+(\Omega)$ is weakly-* compact, hence $\mathcal{M}_1^+(\Omega)_{/\sim}$ 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 (φ_1, φ_2) of tensorization degrees (n_2, n_2)), we denote $n \stackrel{\text{def.}}{=} \max(n_1, n_2)$ and $\varphi(x_1, \ldots, x_n) \stackrel{\text{def.}}{=} \varphi_1(x_1, \ldots, x_{n_1}) + \varphi_2(x_1, \ldots, x_{n_2})$ which is indeed invariant, hence $\mathcal{F}_1 + \mathcal{F}_2 = \int_{\Omega^n} \varphi \mathrm{d}\alpha^{\otimes n} \in \mathcal{A}_\Omega$. Similarly, for stability by product: denoting this time $n = n_1 + n_2$, we introduce the invariant $\varphi(x_1, \ldots, x_n) = \varphi_1(x_1, \ldots, x_{n_1}) \times \varphi_2(x_{n_1+1}, \ldots, x_n)$, 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 $\varphi = \mathbb{1}_S$ (invariant since S is symmetrized) yields an $\mathcal{F}$ such that $\mathcal{F}(\alpha) \neq \mathcal{F}(\nu)$. □

4. Learning Meta-Features: Proof of Concept

The experimental validation presented in this section considers two goals of experiments: (i) assessing the ability of DIDA to learn accurate meta-features; (ii) assessing the merit of the DIDA invariant layer design, building invariant T_φ on the top of an interactional function φ (Eq. 30). As said, this architecture is expected to grasp contrasts among samples, e.g. belonging to different classes; the proposed experimental setting aims to empirically investigate this conjecture. The code used to produce figures in this section is available at: `https://github.com/herilalaina/dida`.

Baselines. These goals of experiments are tackled by comparing DIDA to three baselines: DSS layers [Maron et al., 2020]; hand-crafted meta-features (HC) [Muñoz et al., 2018] (Table 7 in Appendix – Section 5); DATASET2VEC [Jomaa et al., 2019]. We implemented DSS, the code being not available. In order to cope with varying dataset dimensions (as required by the UCI and OpenML benchmarks), the original DSS was augmented with an aggregator summing over the features. Three DSS baselines are considered: linear or non-linear invariant layers, possibly preceded by equivariant layers. Similarly, the original DATASET2VEC implementation has been augmented to address

our experimental setting. The baselines are detailed in Appendix – Section 5.

Experimental setting. Two tasks defined at the dataset level are considered: patch identification (section 4.1) and performance modelling (section 4.2). The dataset preprocessing protocols are detailed in Appendix – Section 5. On both tasks, the same DIDA architecture is considered (Fig 7), involving 2 invariant layers followed by 3 fully connected (FC) layers. Meta-features $\mathcal{F}_\zeta(X)$ consist of the output of the third FC layer, with ζ denoting the trained DIDA parameters. All experiments are run on 1 NVIDIA-Tesla-V100-SXM2 GPU with 32GB memory, using Adam optimizer with base learning rate 10^{-3} and batch size 32.

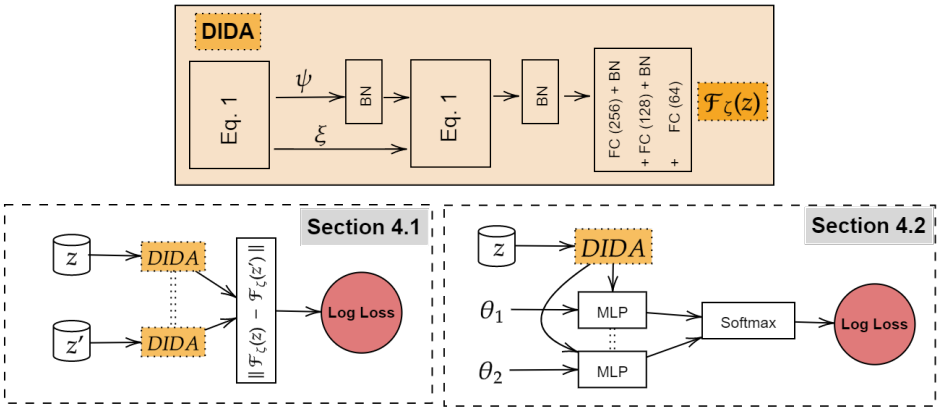


Figure 7: Learning meta-features with DIDA. Top: the DIDA architecture (BN stands for batch norm; FC for fully connected layer). Bottom left: Learning meta-features for patch identification using a Siamese architecture (section 4.1). Bottom right: learning meta-features for performance modelling, specifically to rank two hyper-parameter configurations θ_1 and θ_2 (section 4.2).

4.1. Distribution Identification

The patch identification task consists of detecting whether two blocks of data are extracted from the same original dataset [Jomaa et al., 2019]. Letting $\mathbf{u}$ denote a n-sample, d-dimensional dataset, an n_z, d_z patch X is constructed from $\mathbf{u}$ by selecting n_z examples in $\mathbf{u}$ (sampled uniformly with

replacement) and retaining their description along d_z features (sampled uniformly with replacement). The size n_z and number d_z of features of the patch are uniformly selected in fixed intervals (Table 6, Appendix – Section 5). To each pair of patches X, X' with same number of instances $n_z = n_{z'}$, is associated a binary meta-label $\ell(X, X')$ set to 1 iff X and X' are extracted from the same initial dataset $\mathbf{u}$. DIDA parameters ζ are trained to minimize the cross-entropy loss of model $\hat{\ell}_\zeta(X, X') = \exp\left(-\|\mathcal{F}_\zeta(X) - \mathcal{F}_\zeta(X')\|_2\right)$, with $\mathcal{F}_\zeta(X)$ and $\mathcal{F}_\zeta(X')$ the meta-features computed for X and X':

$$\min_\zeta - \sum_{X,X'} \ell(X, X') \log\Big(\hat{\ell}_\zeta(X, X')\Big) + (1 - \ell(X, X')) \log\Big(1 - \hat{\ell}_\zeta(X, X')\Big) \tag{34}$$

DIDA and all baselines are trained using a Siamese approach (Figure 7, bottom left): the same DIDA (or baseline) architecture is used to compute meta-features $\mathcal{F}_\zeta(X)$ and $\mathcal{F}_\zeta(X')$ from patches X and X', and trained to minimize the cross-entropy loss w.r.t. $\ell(X, X')$. The classification results on toy datasets and UCI datasets (Table 3, detailed in Appendix – Section 5) show the pertinence of the DIDA meta-features, particularly so on the UCI datasets where the number of features widely varies from one dataset to another. The relevance of the interactional invariant layer design is established on this problem as DIDA outperforms both DATASET2VEC, DSS as well as the function learned on the top of the hand-crafted meta-features.

An ablation study is conducted to assess the impact of (i) the feature permutation invariance; (ii) considering one *vs* two invariant layers of type (30). The so-called NO-FINV-DSS baseline, detailed in Appendix – Section 5, is built upon [Zaheer et al., 2017]; it only differs from the DSS baseline as it is *not* feature permutation invariant. With ca the same number of parameters as DSS, its performances are significantly lower (Table 3), showcasing the benefits of enforcing the feature invariance property. Secondly, we compare the 2-invariant layers DIDA, with the 1-invariant layer DIDA (1L-DIDA and 2L-DIDA for short): 1L-DIDA yields significantly lower performances, which confirms the advantages of maintaining the distributional nature among several layers, as already noted by [De Bie et al., 2019]. Note that the 1L-DIDA still outperforms the non feature-invariant baseline, while requiring much fewer parameters.

Method	# parameters	TOY	UCI
Hand-crafted	53,312	77.05 %± 1.63	58.36 %± 2.64
No-FInv-DSS (no invariance in features)	1,297,692	90.49 %± 1.73	64.69 %± 4.89
Dataset2Vec	257,088	96.19 %± 0.28	77.58 %± 3.13
DSS layers (Linear aggregation)	1338684	89.32 %± 1.85	76.23 %± 1.84
DSS layers (Non-linear aggregation)	1,338,684	96.24 %± 2.04	83.97 %± 2.89
DSS layers (Equivariant+invariant)	1,338,692	96.26 %± 1.40	82.94 %± 3.36
Dida (1 invariant layer)	323,028	91.37 %± 1.39	81.03 %± 3.23
Dida (2 invariant layers)	1,389,089	**97.2** % ± **0.1**	**89.70** % ± **1.89**

Table 3: Patch identification (binary classification accuracy) on 10 runs of Dida and considered baselines.

4.2. Performance Model Learning

The performance modelling task aims to assess *a priori* the accuracy of the classifier learned from a given machine learning algorithm with a given configuration θ (vector of hyper-parameters ranging in a hyper-parameter space Θ, Table 8 in Appendix – Section 5), on a dataset X (for brevity, the performance of θ on X) [Rice, 1976].

For each ML algorithm, ranging in Logistic regression (LR), SVM, k-Nearest Neighbours (k-NN), linear classifier learned with stochastic gradient descent (SGD), a set of meta-features is learned to predict whether some configuration θ_1 outperforms some configuration θ_2 on dataset X: to each triplet (X, θ_1, θ_2) is associated a binary value $\ell(X, \theta_1, \theta_2)$, set to 1 iff θ_2 yields better performance than θ_1 on X. Dida parameters ζ are trained to build model $\hat{\ell}_\zeta$, minimizing the (weighted version of) cross-entropy loss (34), where $\hat{\ell}_\zeta(X, \theta_1, \theta_2)$ is a 2-layer FC network with input vector $[\mathcal{F}_\zeta(X); \theta_1; \theta_2]$, depending on the considered ML algorithm and its configuration space.

In each epoch, a batch made of triplets (X, θ_1, θ_2) is built, with θ_1, θ_2 uniformly drawn in the algorithm configuration space (Table 8) and X a n-sample d-dimensional patch of a dataset in the OpenML CC-2018 [Bischl et al., 2019] with n uniformly drawn in $[700; 900]$ and d in $[3; 10]$. Algorithm 1 summarizes the training procedure.

The quality of the Dida meta-features is assessed from the ranking accuracy (Table 4), showing their relevance. The performance gap compared to the baselines is higher for the k-NN modelling task; this is explained as the sought performance model only depends on the local geometry of the examples. Still, good performances are observed over all considered

Method	SGD	SVM	LR	k-NN
Hand-crafted	71.18 %± 0.41	75.39 %± 0.29	86.41 %± 0.419	65.44 %± 0.73
DATASET2VEC	74.43 %± 0.90	81.75 %± 1.85	89.18 %± 0.45	72.90 %± 1.13
DSS (Linear aggregation)	73.46 %± 1.44	82.91 %± 0.22	87.93 %± 0.58	70.07 %± 2.82
DSS (Equivariant+Invariant)	73.54 %± 0.26	81.29 %± 1.65	87.65 %± 0.03	68.55 %± 2.84
DSS (Non-linear aggregation)	74.13 %± 1.01	83.38 %± 0.37	87.92 %± 0.27	73.07 %± 0.77
DIDA (1 invariant layer)	77.31 %± 0.16	84.05 %± 0.71	**90.16 %± 0.17**	74.41 %± 0.93
DIDA (2 invariant layers)	**78.41 %± 0.41**	**84.14 %± 0.02**	89.77 %± 0.50	**78.91 %± 0.54**

Table 4: Pairwise ranking of configurations, for ML algorithms SGD, SVM, LR and k-NN: performance on test set of DIDA, hand-crafted, DATASET2VEC and DSS (average and std deviation on 3 runs).

Algorithm 1 Performance Modeling

$\mathcal{F}_\zeta \leftarrow$ meta-feature extractor (DIDA, DSS, DATASET2VEC, or Hand-crafted)
MLP $\leftarrow$ NN[Linear(64)-ReLU-Linear(32)-ReLU-Linear(1)]
CLF $\leftarrow$ machine learning classifier (SGD, SVM, LR or k-NN)
error $\leftarrow$ 3-CV classification error function
for iteration=$1, 2, \ldots$ **do**
    Sample (θ_1, θ_2), two hyper-parameters of CLF ▷ Search space: Table 8
    Sample patch X from dataset **u** ▷ Patch dimension: Table 6
    pred $\leftarrow$ softmax(MLP($\mathcal{F}_\zeta(X), \theta_1$), MLP($\mathcal{F}_\zeta(X), \theta_2$))
    Backpropagate logloss(pred, 0 if error(X, CLF(θ_1)) $<$ error(X, CLF(θ_2)) else 1)
end for

algorithms. Note that the 2L-DIDA yields significantly better (respectively, similar) performances than 1L-DIDA on the k-NN model (resp. on all other models).

Meta-feature assessment. A regression setting is thereafter considered, aimed to predict the actual performance of a configuration θ based on the (frozen) meta-features $\mathcal{F}_\zeta(X)$. The regression accuracy is illustrated for the configurations of the k-NN algorithm on Figure 8, left (results for other algorithms are presented in Appendix – Section 5). The comparison with the regression models based on DSS meta-features or hand-crafted features (Figure 8, middle and right) shows the merits of the DIDA architecture; a tentative interpretation for the DIDA better performance is based on the interactional nature of DIDA architecture, better capturing local interactions.

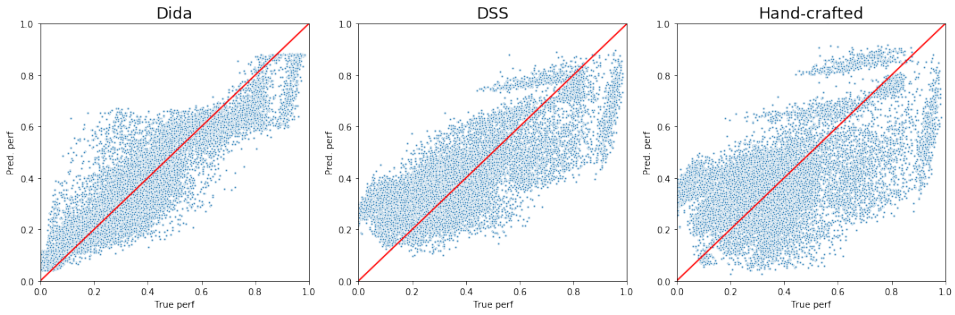


Figure 8: k-NN: True performance vs performance predicted by regression on top of the meta-features (i) learned by DIDA, (ii) DSS or (iii) Hand-crafted statistics.

5. Appendix

5.1. Benchmark details

Three benchmarks are used (Table 5): TOY and UCI, taken from [Jomaa et al., 2019], and OpenML CC-18 [Bischl et al., 2019]. TOY includes 10,000 datasets, where instances are distributed along mixtures of Gaussian, intertwinning moons and rings in $\mathbb{R}^2$, with 2 to 7 classes. UCI includes 121 datasets from the UCI Irvine repository [Dua and 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 6. 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 5: Benchmarks characteristics

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

Table 6: Patch Size

5.2. 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 3). Note that function *generate_patches*() is extracted from the DATASET2VEC source code.

Algorithm 2 Batch Identification

$\mathcal{F}_\zeta \leftarrow$ meta-feature extractor (DIDA Deep Sets, DSS, or Hand-crafted)
for iteration=$1, 2, \ldots$ **do**
    $X_1, X_2, y \leftarrow$ generate_patches() $\triangleright$ $y \leftarrow 1$ if X_1 and X_2 are from the same dataset else 0
    $mf_1 \leftarrow \mathcal{F}_\zeta(X_1)$
    $mf_2 \leftarrow \mathcal{F}_\zeta(X_2)$
    Backpropagate logloss($\exp(-\|mf_1 - mf_2\|_2)$, y)
end for

5.3. Baseline Details

Dataset2Vec details. The available implementation of DATASET2VEC[1] 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

$$D : X \in \mathcal{R}_n(\mathbb{R}^d) \mapsto h\left(\frac{1}{d_X d_Y}\sum_{m=1}^{d_X}\sum_{t=1}^{d_Y} g\left(\frac{1}{n}\sum_{i=1}^{n} f(x_i[m], y_i[t])\right)\right) \tag{35}$$

where functions f, g, h characterizing the architecture are chosen as depicted in the publicly available file *config.py*[2]. More precisely, f, g are FC(128)-ReLU-ResFC(128, 128, 128)-FC(128) and h is FC(128)-ReLU-FC(128)-ReLU where ResFC is a sequence of fully connected layer with skip connection.

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

$$L_{inv} : X \in \mathbb{R}^{n\times d} \mapsto L^H(\sum_{j=1}^{n} x_j) \in \mathbb{R}^K \tag{36}$$

where $L^H : \mathbb{R}^d \to \mathbb{R}^K$ 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)

$$L^H : x = (x_1, \ldots, x_d) \in \mathbb{R}^d \mapsto \rho\left(\sum_{i=1}^{d} \varphi(x_i)\right) \in \mathbb{R}^K \tag{37}$$

where $\varphi : \mathbb{R} \to \mathbb{R}^{d+1}$ and $\rho : \mathbb{R}^{d+1} \to \mathbb{R}^K$ are modelled as (i) purely linear functions; (ii) FC networks, which extends the initial linear setting (36). In our case, $H = S_{d_X} \times S_{d_Y}$, hence, two invariant layers of the form (36-37) are combined to suit both feature- and label-invariance requirements. Both outputs are concatenated and followed by an FC network to form the DSS

[1] See `https://github.com/hadijomaa/dataset2vec`

[2] See `https://github.com/hadijomaa/dataset2vec/blob/master/config.py`

meta-features. The last experiments use DSS equivariant layers (see [Maron et al., 2020], Theorem 1), which take the form

$$L_{eq} : X \in \mathbb{R}^{n\times d} \mapsto \left(L^1_{eq}(x_i) + L^2_{eq}(\sum_{j\neq i} x_j) \right)_{i\in[n]} \in \mathbb{R}^{n\times d} \tag{38}$$

where L^1_{eq} and L^2_{eq} 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.

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

$$L : X = (X_1, \ldots, X_n) \in Z(\mathbb{R}^d) \mapsto \frac{1}{n}\sum_{i=1}^{n} g(X_i) \in \mathbb{R}^K \tag{39}$$

where $g : \mathbb{R}^d \to \mathbb{R}^K$ 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 and 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

$$X \in \mathcal{R}(\mathbb{R}^d) \mapsto \text{MLP}([L(X); \bar{y}]) \tag{40}$$

Hand-crafted meta-features. For the sake of reproducibility, the list of meta-features used in Section 4 is given in Table 7. 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.

5.4. Performance Prediction

Experimental setting. Table 8 details all hyper-parameter configurations Θ considered in Section 4.2. As said, the learnt meta-features $\mathcal{F}_\zeta(X)$ can be used in a regression setting, predicting the performance of various ML algorithms on a dataset X. Several performance models have been considered 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 9 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 9. Figure 9 complements Table 9 in assessing the learnt Dida meta-features for performance model learning. It shows Dida's ability to capture more expressive meta-features than both DSS and hand-crafted ones, for all ML algorithms considered.

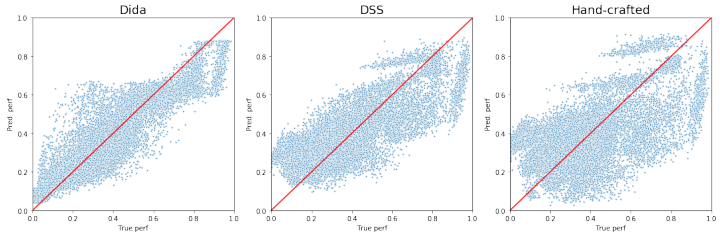
5.5. Stability of meta-features with respect to sample and feature sampling

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 5.3) are reported in Fig. 10. Both Dida and No-FInv-DSS are trained on Task 1.

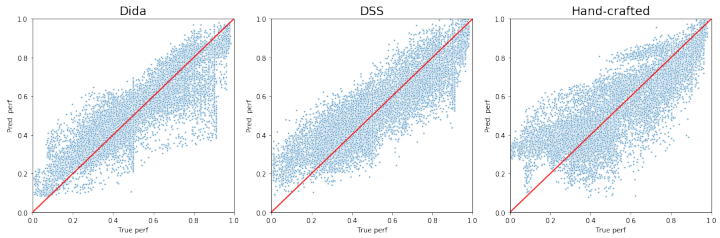
Specifically, the three settings aim to measure the robustness w.r.t. (A) the uniform selection of the samples only; (B) the uniform selection of the samples and the permutation of features; (C) the uniform selection of the samples and the features:

A Considering a fixed set of features, 128 patches are extracted from a dataset $\mathbf{u}$. For each patch X, Dida computes a meta-feature vector $\mathcal{F}_\zeta(X)$ in $\mathbb{R}^{64}$. The reference vector is the average of these meta-feature vectors. Fig. 10.A reports the mean and standard deviation of the distance between the meta-feature vectors and their mean (Fig. 10.A).

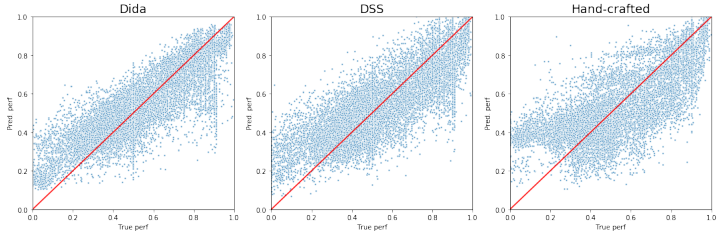
B Same as in A, except that for each patch, the features are permuted. The reference vector is the same as in [A]. The mean and standard deviation of the distances between these meta-feature vectors and the



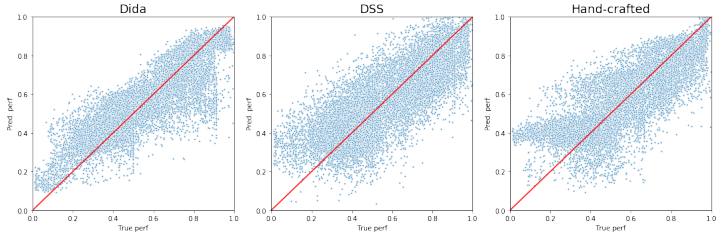
(a) k-NN



(b) Logistic Regression



(c) SVM



(d) SGD

Figure 9: 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.

reference vector thus reflect the impact of the permutation of features (Fig. 10.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. 10.C).

Fig. 10 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.

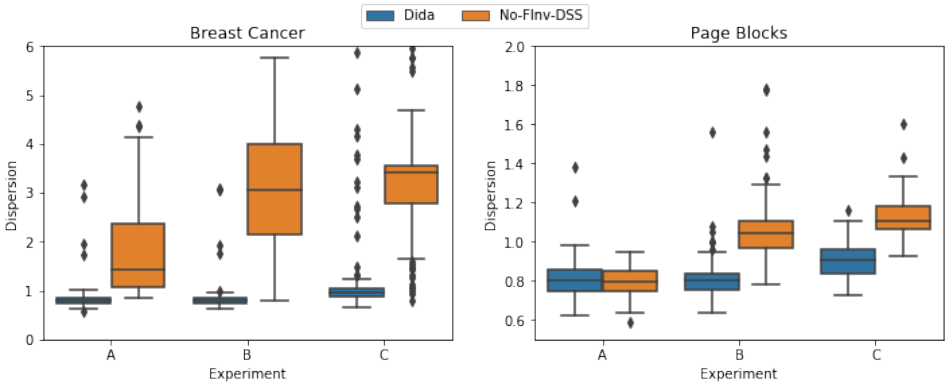


Figure 10: Robustness of meta-features: average and standard deviation of the distance between the meta-feature vectors and their reference vector along the A, B, and C settings (please see text). Left: Breast Cancer dataset. Right: Page Blocks dataset.

Meta-features	**Mean**	**Min**	**Max**
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 7: Hand-crafted meta-features

	Parameter	Parameter values	Scale
LR	warm start	True, Fase	
	fit intercept	True, Fase	
	tol	[0.00001, 0.0001]	
	C	[1e-4, 1e4]	log
	solver	newton-cg, lbfgs, liblinear, sag, saga	
	max_iter	[5, 1000]	
SVM	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]	
KNN	n_neighbors	[1, 100]	log
	p	[1, 2]	
	weights	uniform, distance	
SGD	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 8: Hyper-parameter configurations considered in Section 4.2.

Method	SGD	SVM	LR	KNN
Hand-crafted	0.016 ± 0.001	0.021 ± 0.001	0.018 ± 0.002	0.034 ± 0.001
DSS (Linear aggregation)	0.015 ± 0.007	0.020 ± 0.002	0.019 ± 0.001	0.025 ± 0.010
DSS (Equivariant+Invariant)	0.014 ± 0.002	0.017 ± 0.003	0.015 ± 0.003	0.028 ± 0.003
DSS (Non-linear aggregation)	0.015 ± 0.009	0.016 ± 0.003	0.014 ± 0.001	0.020 ± 0.005
DIDA	**0.012 ± 0.001**	**0.015 ± 0.001**	**0.010 ± 0.001**	**0.009 ± 0.000**

Table 9: 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).

Chapter 3: Regularized Vector Quantile Regression

Quantile regression, introduced by the seminal work of Koenker and Bassett [Koenker and Bassett, 1978] is recognized to this day as a powerful tool to analyze the response of an explained variable to a set of predictors, at any quantile of the distribution. It thus allows to recover the whole conditional distribution, as opposed to just the median.

There is to this day however no consensus on how to extend this method to the case of a multivariate response variable. Carlier, Galichon and Chernozhukov [Carlier et al., 2016b, 2017] have proposed the Vector Quantile Regression (VQR) expansion, based on Optimal Transport, that is linked to polar factorization [Ryff, 1970, Brenier, 1991, McCann, 1995] in the sense that the multivariate quantile is the gradient of a convex function. The proposed approach focuses on retrieving two desirable properties of quantiles in higher dimension, namely monotonicity and transport from a fixed distribution.

Until now, numerical solvers rely on linear programming that is hardly scalable to high dimensional settings. In this chapter, we introduce an entropy-regularized variant of this problem called Regularized Vector Quantile Regression (RVQR) that alleviates this computational hurdle in high dimension. We demonstrate the scalability of the approach through a range of experiments, and show that original quantiles are still retrieved in 1D. We also exhibit the statistical benefits of this problem in finite dimension, by recovering a central limit theorem in the finite sample case, that paves the way for hypothesis testing on regression coefficients.

This chapter is based on [Carlier et al., 2020].

1. Introduction

While ordinary least squares provides a convenient method to estimate the effect of predictors on the conditional mean of an outcome variable, quantile regression has emerged [Koenker and Bassett, 1978, Koenker, 2005] as a way to provide estimates of their impact at any conditional quantile of the response. Quantile regression has notoriously appeared in the last decades as a useful tool to evaluate public policies, and its ability to model extreme values accurately has also made it a popular approach in economics and finance: it has emerged in areas ranging from healthcare [Koenker and Hallock, 2001, Austin et al., 2005, Azagba and Sharaf, 2012], bioinformatics [Song et al., 2017], education [Eide and Showalter, 1998], finance [Zietz et al., 2008], ecology [Cade and Noon, 2003] to reduction of inequalities [Chamberlain, 1994, Buchinsky, 1994, 1998, Melly, 2005]. The ability to interpret quantile regression coefficients as estimates for treatment effects under a control population [Lehmann, 1974, Doksum, 1974, Koenker, 2005] had enormous impact.

In this chapter, we focus on Vector Quantile Regression (VQR), a multivariate extension of quantile regression introduced in the seminal works of [Carlier et al., 2016b, 2017]. This approach, based on optimal transport, relies in practice on linear programming. Our goal is to ease the computation of the conditional quantiles in this method, making it amenable to high dimensional settings. For that purpose, we advocate for solving a regularized version of the original problem. Indeed, strong regularizers such as the entropy [Wilson, 1969, Erlander and Stewart, 1990, Cuturi, 2013] have long been considered in numerical optimal transport to force the solution to have a spread non-sparse support, which stabilizes the computation while ensuring the objective is strongly convex. These desired computational and analytical properties are complemented by eligibility for stochastic [Genevay et al., 2016] and acceleration techniques [Altschuler et al., 2017, Scieur et al., 2016]. Its statistical properties [Genevay et al., 2019, Mena and Niles-Weed, 2019, Bigot et al., 2019a, Chizat et al., 2020] and algorithmic improvements [Altschuler et al., 2017, Mensch and Peyré, 2020] constitute an active field of research to this day.

Previous works.

Quantile regression. Early appearances of median regression can be traced back to the 18^{th} century work of Boscovich, and later Laplace, that considered a "method of situation" blending mean and median ideas. A

century later, [Edgeworth, 1888] formalized the idea of minimizing the sum of absolute residuals. His proposal was revived when recognized as linear programming in the 1950s and applied to economics [Arrow and Hoffenberg, 1959]. [Fox and Rubin, 1964] began considering the loss function $\rho_\tau(u) = u(\tau - \mathbb{1}\{u < 0\})$ to investigate admissibility of quantile estimates until [Koenker and Bassett, 1978] introduced the regression setting and its asymptotic behavior. The ability to interpret quantile regression coefficients as treatment effects under a control population, in the case of binary [Lehmann, 1974, Doksum, 1974, Koenker, 2005] or multiple treatments [Koenker, 2005, Wang et al., 2018] gathered a lot of interest, as well as pointwise [Koenker, 2005] or uniform [Koenker, 2011, Belloni et al., 2014] confidence intervals. Thanks to its ability to characterize the whole conditional distribution, quantile regression has been linked to structural models [Matzkin, 2015]. The original model has been extended to nonlinear dependencies in parameters, incorporating censorship [Powell, 1986], Box-Cox transformations [Machado and Mata, 2000] or others [Koenker, 2005]. The capacity to relax linearity in covariates while preserving linearity in parameters has been extensively studied as nonparameteric quantile regression, from locally polynomial [Chaudhuri, 1991] or partially linear [Lee, 2003] to sparsity-oriented approaches able to control the parametric dimension of the models, with ℓ_1 or related total variation penalties, see for instance [Chen et al., 2001, Koenker et al., 1994]. This has also paved the way to a large literature on post selection inference [Belloni et al., 2014, Koenker, 2011]. Quantile regression has been extended to cope with time series models [Xiao and Koenker, 2002] or their frequency counterpart [Li, 2008], panel data [Wei et al., 2006, Arellano and Bonhomme, 2016], duration models [Koenker and Geling, 2001], missing data [Yang et al., 2018], causal models [Chesher, 2003] and instrumental variables [Chernozhukov and Hansen, 2004]. From the early simplex method, computational procedures have evolved to interior point methods [Portnoy and Koenker, 1997] with a later focus on sparse algebra [Koenker, 2011]. The increasing parametric dimension has shifted focus back to gradient descent, hence the adaptation of quantile regression through the alternating direction method of multipliers [Koenker, 2017].

Notions of multivariate quantiles. Unsurprisingly, extending quantile regression to the multivariate setting raises a number of questions, since the "inversion" of a functional $F : \mathbb{R}^d \to [0; 1]$ or the notion of multivariate median are not straightforward. Several bases to expand the notion of quantile to the multivariate case have been considered, relying on (i) orderings of multivariate

data, as exemplified by the use of depth functions [Tukey, 1975]; (*ii*) the extension of empirical quantile processes [Pyke, 1975]; (*iii*) the expansion of the distribution function to the multivariate case, such that its inverse can be thought of as a multivariate quantile [Chaudhuri, 1996, Koltchinskii, 1997]. [Kong and Mizera, 2012] propose a notion of directional quantile (quantile of projection), based on the minimization of $\mathbb{E}\left[\rho_\tau\left(Y_u - \alpha - \beta^\top Y_u^\perp\right)\right]$ over (α, β), where, for a vector $u \in \mathbb{R}^d$, $Y_u^\perp$ denotes the orthogonal of $Y_u \overset{\text{def.}}{=} u^\top Y$. The latter is stated in the absence of covariates for the sake of simplicity. Their envelopes coincide with halfspace depth contours [Tukey, 1975] so that computation is enabled through parametric linear programming [Hallin et al., 2010] and extensions to nonparametric formulations can be considered [Hallin et al., 2015]. A stream of works consider definitions based on M-estimators [Koltchinskii, 1997, Chaudhuri, 1996, Serfling, 2004]. [Wei, 2008] propose to define bivariate quantiles using Knothe-Rosenblatt transport, which is known to be linked to optimal transport [Carlier et al., 2008]. [Belloni and Winkler, 2011] suggest a notion of multivariate partial quantile, based on a partial order on $\mathbb{R}^d$. [Kato, 2012] considers the case of function-valued covariates and proposes estimates based on principal component analysis and corresponding plug-in estimators for quantiles. Among the above definitions, it is to be noted that some are set-valued. The Vector Quantile Regression (VQR) approach, which we focus on, developed by [Carlier et al., 2016b, 2017] defines multivariate quantiles as a transport from a fixed distribution (for instance uniform on a cube) to the conditional law, that maximizes their correlation. Thanks to polar factorization [Ryff, 1970, Brenier, 1991, McCann, 1995], it satisfies monotonicity of the quantile curves. Its current computational procedures rely on linear programming [Carlier et al., 2016b].

Contributions.

In this chapter, we propose to consider a regularized version of the correlation maximization problem introduced in [Carlier et al., 2016b, 2017], penalizing the entropy of the joint distribution, called the Regularized Vector Quantile Regression (RVQR) approach. Due to smoothness and regularity, the RVQR problem enjoys computational and analytical properties that are missing from the original VQR formulation. In particular, its dual problem is a smooth, unconstrained problem that can be solved efficiently using accelerated [Nesterov, 1983] gradient descent, which gives optimal convergence rates for first-order methods. Numerical illustrations are presented in the multivariate case, and classical quantile curves are retrieved in the one-dimensional case. Asymptotics in the finite-sample case are analyzed

in finite dimension, which allows to uncover a law of large numbers and a central limit theorem for the RVQR finite-sample potentials.

This chapter is organized as follows. Section 2 offers reminders on the notion of quantile; Section 3 will review the previous results of [Carlier et al., 2016b, 2017] on the "specified" case, and offers insight on the comparison with the shape-constrained classical quantile regression; and Section 4 reviews results on the multivariate case. Section 5 introduces the RVQR problem coupled with relevant results for that problem, as well as computational considerations and numerical results. Section 6 provides insight into asymptotics of the RVQR finite-dimensional potentials in the finite sample case.

2. Several Characterizations of Quantiles

Throughout this chapter, $(\Omega, \mathcal{F}, \mathbb{P})$ will be some fixed nonatomic space[3] probability. Given a random vector Z with values in $\mathbb{R}^k$ defined on this space we will denote by $\mathscr{L}(Z)$ the law of Z, given a probability measure θ on $\mathbb{R}^k$, we often write $Z \sim \theta$ to express that $\mathscr{L}(Z) = \theta$. Independence of two random variables Z_1 and Z_2 will be denoted as $Z_1 \perp\!\!\!\perp Z_2$.

2.1. Quantiles

Let Y be some univariate random variable defined on $(\Omega, \mathcal{F}, \mathbb{P})$. Denoting by F_Y the distribution function of Y:

$$F_Y(\alpha) \stackrel{\text{def.}}{=} \mathbb{P}(Y \leq \alpha), \ \forall \alpha \in \mathbb{R}$$

the *quantile* function of Y, $Q_Y = F_Y^{-1}$ is the generalized inverse of F_Y given by the formula:

$$Q_Y(t) \stackrel{\text{def.}}{=} \inf\{\alpha \in \mathbb{R} \ : \ F_Y(\alpha) > t\} \text{ for all } t \in (0,1). \tag{41}$$

Let us now recall two well-known facts about quantiles:

- $\alpha = Q_Y(t)$ is a solution of the convex minimization problem

$$\min_{\alpha} \{\mathbb{E}((Y-\alpha)^+) + \alpha(1-t)\} \tag{42}$$

[3] One way to define the nonatomicity of $(\Omega, \mathcal{F}, \mathbb{P})$ is by the existence of a uniformly distributed random variable on this space, this somehow ensures that the space is rich enough so that there exists random variables with prescribed law. If, on the contrary, the space is finite for instance only finitely supported probability measures can be realized as the law of such random variables.

- there exists a uniformly distributed random variable U such that $Y = Q_Y(U)$. Moreover, among uniformly distributed random variables, U is maximally correlated[4] to Y in the sense that it solves

$$\max\{\mathbb{E}(VY),\ V \sim \mu\} \tag{43}$$

where $\mu \stackrel{\text{def.}}{=} \mathcal{U}([0,1])$ is the uniform measure on $[0,1]$.

Of course, when $\mathscr{L}(Y)$ has no atom, i.e. when F_Y is continuous, U is unique and given by $U = F_Y(Y)$. Problem (43) is the easiest example of optimal transport problem one can think of. The decomposition of a random variable Y as the composed of a monotone nondecreasing function and a uniformly distributed random variable is called a *polar factorization* of Y. The existence of such decompositions goes back to [Ryff, 1970] and the extension to the multivariate case (by optimal transport) is due to [Brenier, 1991].

We therefore see that there are basically two different approaches to study or estimate quantiles:

- the *local* or "t by t" approach which consists, for a fixed probability level t, in using directly formula (41) or the minimization problem (42) (or some approximation of it), this can be done very efficiently in practice but has the disadvantage of forgetting the fundamental global property of the quantile function: it should be monotone in t,

- the global approach (or polar factorization approach), where quantiles of Y are defined as all nondecreasing functions Q for which one can write $Y = Q(U)$ with U uniformly distributed. In this approach, one rather tries to recover directly the whole monotone function Q (or the uniform variable U that is maximally correlated to Y). Therefore this is a global approach for which one should rather use the optimal transport problem (43).

2.2. Conditional Quantiles

Let us assume now that, in addition to the random variable Y, we are also given a random vector $X \in \mathbb{R}^N$ which we may think of as being a list of explanatory variables for Y. We are primarily interested in the dependence between Y and X and in particular the conditional quantiles of Y given

[4] In fact for (43) to make sense one needs some integrabilty of Y i.e. $\mathbb{E}(|Y|) < +\infty$.

$X = x$. Let us denote by ν the joint law of (X, Y) by ν the law of X, and by $\nu(.|x)$ the conditional law of Y given $X = x$:

$$\nu \stackrel{\text{def.}}{=} \mathscr{L}(X, Y),\ m \stackrel{\text{def.}}{=} \mathscr{L}(X),\ \nu(.|x) \stackrel{\text{def.}}{=} \mathscr{L}(Y|X = x) \tag{44}$$

which in particular yields

$$\mathrm{d}\nu(x, y) = \mathrm{d}\nu(y|x)\mathrm{d}m(x).$$

We then denote by $F(x, y) = F_{Y|X=x}(y)$ the conditional cdf:

$$F(x, y) \stackrel{\text{def.}}{=} \mathbb{P}(Y \leq y|X = x)$$

and $Q(x, t)$ the conditional quantile

$$Q(x, t) \stackrel{\text{def.}}{=} \inf\{\alpha \in \mathbb{R} \ :\ F(x, \alpha) > t\},\ \forall t \in (0, 1).$$

For the sake of simplicity, we assume that for $m = \mathscr{L}(X)$-almost every $x \in \mathbb{R}^N$ (m-a.e. x for short), one has

$$t \mapsto Q(x, t) \text{ is continuous and increasing} \tag{45}$$

so that for m-a.e. x, $F(x, Q(x, t)) = t$ for every $t \in (0, 1)$ and $Q(x, F(x, y)) = y$ for every y in the support of $\nu(.|x)$.

Let us now define the random variable

$$U \stackrel{\text{def.}}{=} F(X, Y), \tag{46}$$

then by construction:

$$\begin{aligned}\mathbb{P}(U < t|X = x) &= \mathbb{P}(F(x, Y) < t|X = x) = \mathbb{P}(Y < Q(x, t)|X = x)\\ &= F(x, Q(x, t)) = t.\end{aligned}$$

We deduce that U is uniformly distributed and independent from X (since its conditional cdf does not depend on x). Moreover since $U = F(X, Y) = F(X, Q(X, U))$ it follows from (45) that one has the representation

$$Y = Q(X, U)$$

in which U can naturally be interpreted as a latent factor.

This remark leads to a conditional polar factorization of Y through the pointwise relation $Y = Q(X, U)$ with $Q(X, .)$ nondecreasing and $U \sim \mu$, $U \perp\!\!\!\perp X$. We would like to emphasize now that there is a variational principle

behind this conditional decomposition. Let us indeed consider the variant of the optimal transport problem (43) where one further requires U to be independent from the vector of regressors X:

$$\max\{\mathbb{E}(VY),\ \mathscr{L}(V) = \mu,\ V \perp\!\!\!\perp X\}. \tag{47}$$

then we have

Proposition 9. *If $\mathbb{E}(|Y|) < +\infty$ and (45) holds, the random variable U defined in (46) solves (47).*

We refer to [Carlier et al., 2016a], Theorem 4.1 for a proof.

3. Quantile Regression

3.1. Specified Quantile Regression

Since the seminal work of [Koenker and Bassett, 1978], it has been widely accepted that a convenient way to estimate conditional quantiles is to stipulate an affine form with respect to x for the conditional quantile. Since a quantile function should be monotone in its second argument, this leads to the following definition:

Definition 6. *(Specified Quantile Regression) Quantile regression is specified if there exist $(\alpha, \beta) \in C([0,1], \mathbb{R}) \times C([0,1], \mathbb{R}^N)$ such that for m-a.e. x*

$$t \mapsto \alpha(t) + \beta(t)^\top x \text{ is increasing on } [0,1] \tag{48}$$

and

$$Q(x,t) = \alpha(t) + \beta(t)^\top x, \tag{49}$$

for m-a.e. x and every $t \in [0,1]$. If (48)-(49) hold, quantile regression is specified with regression coefficients (α, β).

Specification of quantile regression can be characterized by the validity of an affine in X representation of Y with a latent factor:

Proposition 10. *Let (α, β) be continuous and satisfy (48). Quantile regression is specified with regression coefficients (α, β) if and only if there exists U such that*

$$Y = \alpha(U) + \beta(U)^\top X \text{ almost surely, } \mathscr{L}(U) = \mu,\ U \perp\!\!\!\perp X. \tag{50}$$

We refer to [Carlier et al., 2016a], Proposition 4.3 for a proof.

3.2. Quasi-Specified Quantile Regression

Let us now assume that both X and Y are integrable

$$\mathbb{E}(\|X\| + |Y|) < +\infty \tag{51}$$

and normalize, without loss of generality, X in such a way that

$$\mathbb{E}(X) = 0. \tag{52}$$

Koenker and Bassett showed that, for a fixed probability level t, the regression coefficients (α, β) can be estimated by quantile regression i.e. the minimization problem

$$\inf_{(\alpha,\beta)\in\mathbb{R}^{1+N}} \mathbb{E}(\rho_t(Y - \alpha - \beta^\top X)) \tag{53}$$

where the penalty ρ_t is given by $\rho_t(z) \stackrel{\text{def.}}{=} tz^- + (1-t)z^+$ with z^- and z^+ denoting the negative and positive parts of z. For further use, note that (53) can be conveniently be rewritten as

$$\inf_{(\alpha,\beta)\in\mathbb{R}^{1+N}} \{\mathbb{E}((Y - \alpha - \beta^\top X)^+) + (1-t)\alpha\}. \tag{54}$$

As noticed by Koenker and Bassett, this convex program admits as dual formulation

$$\sup\{\mathbb{E}(V_t Y)) \ : \ V_t \in [0,1],\ \mathbb{E}(V_t) = (1-t),\ \mathbb{E}(V_t X) = 0\}. \tag{55}$$

An optimal (α, β) for (54) and an optimal V_t in (55) are related by the complementary slackness condition:

$$Y > \alpha + \beta^\top X \Rightarrow V_t = 1, \text{ and } Y < \alpha + \beta^\top X \Rightarrow V_t = 0. \tag{56}$$

Note that α appears naturally as a Lagrange multiplier associated to the constraint $\mathbb{E}(V_t) = (1-t)$ and β as a Lagrange multiplier associated to $\mathbb{E}(V_t X) = 0$.

To avoid mixing i.e. the possibility that V_t takes values in $(0,1)$, it will be convenient to assume that $\nu = \mathscr{L}(X,Y)$ gives zero mass to nonvertical hyperplanes i.e.

$$\mathbb{P}(Y = \alpha + \beta^\top X) = 0,\ \forall(\alpha,\beta) \in \mathbb{R}^{1+N}. \tag{57}$$

We also consider a nondegeneracy condition on the (centered) random vector X which says that its law is not supported by any hyperplane[5]:

$$\mathbb{P}(\beta^\top X = 0) < 1,\ \forall \beta \in \mathbb{R}^N \setminus \{0\}. \tag{58}$$

Thanks to (57), we may simply write

$$V_t = \mathbf{1}_{\{Y > \alpha + \beta^\top X\}} \tag{59}$$

and thus the constraints $\mathbb{E}(V_t) = (1-t)$, $\mathbb{E}(XV_t) = 0$ read

$$\mathbb{E}(\mathbf{1}_{\{Y>\alpha+\beta^\top X\}}) = \mathbb{P}(Y > \alpha + \beta^\top X) = (1-t),\ \mathbb{E}(X\mathbf{1}_{\{Y>\alpha+\beta^\top X\}}) = 0 \tag{60}$$

which simply are the first-order conditions for (54).

Any pair (α, β) which solves the optimality conditions (60) for the Koenker and Bassett approach will be denoted

$$\alpha = \alpha^{QR}(t), \beta = \beta^{QR}(t)$$

and the variable V_t solving (55) given by (59) will similarly be denoted V_t^{QR}

$$V_t^{QR} \overset{\text{def.}}{=} \mathbf{1}_{\{Y > \alpha^{QR}(t) + \beta^{QR}(t)^\top X\}}. \tag{61}$$

Note that in the previous considerations the probability level t is fixed, this is what we called the "t by t" approach. For this approach to be consistent with conditional quantile estimation, if we allow t to vary we should add an additional monotonicity requirement:

Definition 7. *Quantile regression is quasi-specified*[6] *if there exists for each t, a solution $(\alpha^{QR}(t), \beta^{QR}(t))$ of (60) (equivalently the minimization problem (53)) such that $t \in [0,1] \mapsto (\alpha^{QR}(t), \beta^{QR}(t))$ is continuous and, for m-a.e. x*

$$t \mapsto \alpha^{QR}(t) + \beta^{QR}(t)^\top x \text{ is increasing on } [0,1]. \tag{62}$$

A first consequence of quasi-specification is given by

Proposition 11. *Assume (45)-(51)-(52) and (57). If quantile regression is quasi-specified and if we define $U^{QR} \overset{\text{def.}}{=} \int_0^1 V_t^{QR} dt$ (recall that V_t^{QR} is given by (61)) then:*

[5] if $\mathbb{E}(\|X\|^2) < +\infty$ then (58) amounts to the standard requirement that $\mathbb{E}(XX^\top)$ is nonsingular.

[6] If quantile regression is specified and the pair of functions (α, β) is as in definition 6, then for every t, $(\alpha(t), \beta(t))$ solves the conditions (60). This shows that specification implies quasi-specification.

- U^{QR} *is uniformly distributed,*
- X *is mean-independent from* U^{QR} *i.e.* $\mathbb{E}(X|U^{QR}) = \mathbb{E}(X) = 0$*,*
- $Y = \alpha^{QR}(U^{QR}) + \beta^{QR}(U^{QR})^\top X$ *almost surely.*

Moreover U^{QR} *solves the correlation maximization problem with a mean-independence constraint:*

$$\max\{\mathbb{E}(VY),\ \mathscr{L}(V) = \mu,\ \mathbb{E}(X|V) = 0\}. \tag{63}$$

We refer to [Carlier et al., 2016a], Proposition 4.5 for a proof. Uniqueness is reached for the mean-independent decomposition given in proposition 11:

Proposition 12. *Assume (45)-(51)-(52)-(57) and (58). Let us assume that*

$$Y = \alpha(U) + \beta(U)^\top X = \overline{\alpha}(\overline{U}) + \overline{\beta}(\overline{U})^\top X$$

with:

- *both* U *and* $\overline{U}$ *uniformly distributed,*
- X *is mean-independent from* U *and* $\overline{U}$*:* $\mathbb{E}(X|U) = \mathbb{E}(X|\overline{U}) = 0$*,*
- $\alpha, \beta, \overline{\alpha}, \overline{\beta}$ *are continuous on* $[0,1]$*,*
- (α, β) *and* $(\overline{\alpha}, \overline{\beta})$ *satisfy the monotonicity condition (48),*

then

$$\alpha = \overline{\alpha},\ \beta = \overline{\beta},\ U = \overline{U}.$$

whose proof can be found in [Carlier et al., 2016a], Proposition 4.6. This argument allows for a strong representation in the quasi-specified case:

Corollary 3. *Assume (45)-(51)-(52)-(57) and (58). If quantile regression is quasi-specified, the regression coefficients* $(\alpha^{QR}, \beta^{QR})$ *are uniquely defined and if* Y *can be written as*

$$Y = \alpha(U) + \beta(U)^\top X$$

for U *uniformly distributed,* X *being mean independent from* U*,* (α, β) *continuous such that the monotonicity condition (48) holds then necessarily*

$$\alpha = \alpha^{QR},\ \beta = \beta^{QR}.$$

As said, quasi-specification is equivalent to the validity of the factor linear model:

$$Y = \alpha(U) + \beta(U)^\top X$$

for (α, β) continuous and satisfying the monotonicity condition (48) and U, uniformly distributed and such that X is mean-independent from U. This has to be compared with the decomposition of paragraph 2.2 where U is required to be independent from X but the dependence of Y with respect to U, given X, is given by a nondecreasing function of U which is not necessarily affine in X.

3.3. Quantile Regression without specification

Now we wish to address quantile regression in the case where neither specification nor quasi-specification can be taken for granted. In such a general situation, keeping in mind the remarks from the previous paragraphs, we can think of two natural approaches.

The first one consists in studying directly the correlation maximization with a mean-independence constraint (63). The second one consists in getting back to the Koenker and Bassett t by t problem (55) but adding as an additional global consistency constraint that V_t should be nonincreasing (which we abbreviate as $V_t \downarrow$) with respect to t:

$$\sup\{\mathbb{E}(\int_0^1 V_t Y dt) \ : \ V_t \downarrow, \ V_t \in [0,1], \ \mathbb{E}(V_t) = (1-t), \ \mathbb{E}(V_t X) = 0\} \quad (64)$$

Our aim is to compare these two approaches (and in particular to show that the maximization problems (63) and (64) have the same value) as well as their dual formulations. Before going further, let us remark that (63) can directly be considered in the multivariate case whereas the monotonicity constrained problem (64) makes sense only in the univariate case.

As proven in [Carlier et al., 2016b], (63) is dual to

$$\inf_{(\psi,\varphi,b)} \{\mathbb{E}(\psi(X,Y)) + \mathbb{E}(\varphi(U)) \ : \ \psi(x,y) + \varphi(u) \geq uy - b(u)^\top x\} \quad (65)$$

which can be reformulated as:

$$\inf_{(\varphi,b)} \int \max_{t\in[0,1]} (ty - \varphi(t) - b(t)^\top x)\nu(dx,dy) + \int_0^1 \varphi(t)dt \quad (66)$$

in the sense that[7]

$$\sup(63) = \inf(65) = \inf(66). \tag{67}$$

The existence of a solution to (65) is not straightforward and is established under appropriate assumptions in [Carlier et al., 2017] in the multivariate case. The following result, proved in [Carlier et al., 2016a], Lemma 4.8, shows that there is a t-dependent reformulation of (63):

Lemma 7. *The value of (63) coincides with*

$$\sup\{\mathbb{E}(\int_0^t V_t Y dt) \ : \ V_t \downarrow, \ V_t \in \{0,1\}, \ \mathbb{E}(V_t) = (1-t), \ \mathbb{E}(V_t X) = 0\}. \tag{68}$$

Let us now define

$$\mathcal{C} \stackrel{\text{def.}}{=} \{v \ : \ [0,1] \mapsto [0,1], \ \downarrow\}$$

Let $(V_t)_t$ be admissible for (64) and set

$$v_t(x,y) \stackrel{\text{def.}}{=} \mathbb{E}(V_t | X = x, Y = y), \ V_t \stackrel{\text{def.}}{=} v_t(X,Y)$$

it is obvious that $(V_t)_t$ is admissible for (64) and by construction $\mathbb{E}(V_t Y) = \mathbb{E}(V_t Y)$. Moreover the deterministic function $(t,x,y) \mapsto v_t(x,y)$ satisfies the following conditions:

$$\text{for fixed } (x,y),\ t \mapsto v_t(x,y) \text{ belongs to } \mathcal{C}, \tag{69}$$

and for a.e. $t \in [0,1]$,

$$\int v_t(x,y)\nu(dx,dy) = (1-t), \ \int v_t(x,y)x\nu(dx,dy) = 0. \tag{70}$$

Conversely, if $(t,x,y) \mapsto v_t(x,y)$ satisfies (69)-(70), $V_t \stackrel{\text{def.}}{=} v_t(X,Y)$ is admissible for (64) and $\mathbb{E}(V_t Y) = \int v_t(x,y) y \nu(dx,dy)$. All this proves that sup(64) coincides with

$$\sup_{(t,x,y)\mapsto v_t(x,y)} \int v_t(x,y) y \nu(dx,dy)dt \text{ subject to: } (69) - (70) \tag{71}$$

The main result of this section, proved in [Carlier et al., 2016a], Theorem 4.9, links the shape-constrained quantile regression problem to correlation maximization as follows:

[7] With a little abuse of notations when a reference number (A) refers to a maximization (minimization) problem, we will simply write $\sup(A)$ ($\inf(A)$) to the denote the value of this optimization problem.

Theorem 7. *The shape constrained quantile regression problem (64) is related to the correlation maximization with a mean independence constraint (63) by:*

$$\sup(63) = \sup(64).$$

4. Vector Quantile Regression

We now consider the case where Y is a random vector with values in $\mathbb{R}^d$ with $d \geqslant 2$. The notion of quantile does not have an obvious generalization in the multivariate setting however, the various correlation maximization problems we have encountered in the previous sections still make sense (provided Y is integrable say) in dimension d and are related to optimal transport theory. The aim of this section is to briefly summarize the optimal transport approach to quantile regression introduced in [Carlier et al., 2016b, 2017].

4.1. Brenier's map as a Vector Quantile

From now on we fix as a reference measure the uniform measure on the unit cube $[0,1]^d$ i.e.

$$\mu_d \stackrel{\text{def.}}{=} \mathcal{U}([0,1]^d) \tag{72}$$

Given Y, an integrable $\mathbb{R}^d$-valued random variable on $(\Omega, \mathcal{F}, \mathbb{P})$, a remarkable theorem due to [Brenier, 1991] and extended by [McCann, 1995] implies that there exists a unique $U \sim \mu_d$ and a unique (up to the addition of a constant) convex function defined on $[0,1]^d$ such that

$$Y = \nabla\varphi(U). \tag{73}$$

The map $\nabla\varphi$ is called the Brenier's map between μ_d and $\mathscr{L}(Y)$.

The convex function φ is not necessarily differentiable but being convex it is differentiable at Lebesgue-a.e. point of $[0,1]^d$ so that $\nabla\varphi(U)$ is well defined almost surely, it is worth at this point recalling that the Legendre transform of φ is the convex function:

$$\varphi^*(y) \stackrel{\text{def.}}{=} \sup_{u\in[0,1]^d} \{u^\top y - \varphi(u)\} \tag{74}$$

and that the subdifferentials of φ and φ^* are defined respectively by

$$\partial\varphi(u) \stackrel{\text{def.}}{=} \{y \in \mathbb{R}^d \ : \ \varphi(u) + \varphi^*(y) = u^\top y\}$$

and

$$\partial\varphi^*(y) \stackrel{\text{def.}}{=} \{u \in [0,1]^d \ : \ \varphi(u) + \varphi^*(y) = u^\top y\}$$

so that $\partial\varphi$ and $\partial\varphi^*$ are inverse to each other in the sense that

$$y \in \partial\varphi(u) \Leftrightarrow u \in \partial\varphi^*(y)$$

which is often refered to in convex analysis as the Fenchel reciprocity formula[8]. Note then that (73) implies that

$$U \in \partial\varphi^*(Y) \text{ almost surely.}$$

If both φ and φ^* are differentiable, their subgradients reduce to the singleton formed by their gradient and the Fenchel reciprocity formula simply gives $\nabla\varphi^{-1} = \nabla\varphi^*$. Recalling the subgradient of the convex function φ is monotone in the sense that whenever $y_1 \in \partial\varphi(u_1)$ and $y_2 \in \partial\varphi(u_2)$ one has

$$(y_1 - y_2)^\top (u_1 - u_2) \geqslant 0,$$

we see that gradients of convex functions are a genelarization to the multivariate case of monotone univariate maps. It is therefore natural in view of (73) to define the vector quantile of Y as:

Definition 8. *The vector quantile of Y is the Brenier's map between μ_d and $\mathscr{L}(Y)$.*

Now, it is worth noting that the Brenier's map (and the uniformly distributed random vector U in (73)) are not abstract objects, they have a variational characterization related to optimal transport[9]. Consider indeed

$$\sup\{\mathbb{E}(V^\top Y) \ : \ V \sim \mu_d\} \tag{75}$$

and its dual

$$\inf_{f,g}\{\int_{[0,1]^d} f \mathrm{d}\mu_d + \mathbb{E}(g(Y)) \ : \ f(u)+g(y) \geqslant u^\top y, \ \forall (u,y) \in [0,1]^d \times \mathbb{R}^d\} \tag{76}$$

then U in (73) is the unique solution of (75) and any solution (f,g) of the dual (76) satisfies $\nabla f = \nabla\varphi$ μ_d-a.e.

[8] Note the analogy with the fact that in the univariate case the cdf and the quantile of Y are generalized inverse to each other.

[9] In the case where $\mathbb{E}(\|Y\|^2) < +\infty$, (75) is equivalent to minimize $\mathbb{E}(\|V - Y\|^2)$ among uniformly distributed V's.

4.2. Conditional Vector Quantiles

Assume now as in paragraph 2.2 that we are also given a random vector $X \in \mathbb{R}^N$. As in (44), we denote by ν the law of (X, Y), by m the law of X and by $\nu(.|x)$ the conditional law of Y given $X = x$ (the only difference with (44) is that Y is $\mathbb{R}^d$-valued). Conditional vector quantile are then defined as

Definition 9. *For $m = \mathscr{L}(X)$-a.e. $x \in \mathbb{R}^N$, the vector conditional quantile of Y given $X = x$ is the Brenier's map between $\mu_d \overset{\text{def.}}{=} \mathcal{U}([0,1]^d)$ and $\nu(.|x) \overset{\text{def.}}{=} \mathscr{L}(Y|X = x)$. We denote this well defined map as $\nabla\varphi_x$ where φ_x is a convex function on $[0,1]^d$.*

If both φ_x and its Legendre transform

$$\varphi_x^*(y) \overset{\text{def.}}{=} \sup_{u\in[0,1]^d} \{u^\top y - \varphi_x(u)\}$$

are differentiable[10], one can define the random vector:

$$U \overset{\text{def.}}{=} \nabla\varphi_X^*(Y)$$

which is equivalent to

$$Y = \nabla\varphi_X(U). \tag{77}$$

One can check exactly as in the proof of Proposition 9 for the univariate case that if Y is integrable then

$$U \sim \mu_d,\ U \perp\!\!\!\perp X$$

and U solves

$$\max\{\mathbb{E}(V^\top Y),\ V \sim \mu_d,\ V \perp\!\!\!\perp X\}. \tag{78}$$

4.3. Vector Quantile Regression

When one assumes that the convex function φ_x is affine with respect to the explanatory variables x (specification):

$$\varphi_x(u) = \varphi(u) + b(u)^\top x$$

[10] A deep regularity theory initated by [Caffarelli, 1992] in the 1990's gives conditions on $\nu(.|x)$ such that this is in fact the case that the optimal transport map is smooth and/or invertible, we refer the interested reader to the textbook of [Figalli, 2017] for a detailed and recent account of this regularity theory.

with $\varphi : [0,1]^d \to \mathbb{R}$ and $b : [0,1]^d \to \mathbb{R}^N$ smooth, the conditional quantile is itself affine and the relation (77) takes the form

$$Y = \nabla\varphi_X(U) = \alpha(U) + \beta(U)X, \text{ for } \alpha = \nabla\varphi, \ \beta \overset{\text{def.}}{=} Db^\top. \tag{79}$$

This affine form moreover implies that not only U maximizes the correlation with Y among uniformly distributed random vectors independent from X but in the larger class of uniformly distributed random vectors for which[11]

$$\mathbb{E}(X|U) = \mathbb{E}(X) = 0.$$

This is the reason why the study of

$$\max\{\mathbb{E}(V^\top Y), \ V \sim \mu_d, \ \mathbb{E}(X|V) = 0\} \tag{80}$$

is the main tool in the approach of [Carlier et al., 2016b, 2017] to vector quantile regression. Let us now briefly summarize the main findings in these two papers. First observe that (80) can be recast as a linear program by setting $\pi \overset{\text{def.}}{=} \mathscr{L}(U, X, Y)$ and observing that U solves (80) if and only if π solves

$$\max_{\pi \in \mathrm{MI}(\mu_d, \nu)} \int_{[0,1]^d \times \mathbb{R}^N \times \mathbb{R}^d} u^\top y \mathrm{d}\pi(u, x, y) \tag{81}$$

where $\mathrm{MI}(\nu, \mu)$ is the set of probability measures which satisfy the linear constraints:

- the first marginal of π is μ_d, i.e., for every $\varphi \in C([0,1]^d, \mathbb{R})$:

$$\int_{[0,1]^d \times \mathbb{R}^N \times \mathbb{R}^d} \varphi(u) \mathrm{d}\pi(u, x, y) = \int_{[0,1]^d} \varphi(u) \mathrm{d}\mu_d(u),$$

- the second marginal of π is ν, i.e., for every $\psi \in C_b(\mathbb{R}^N \times \mathbb{R}^d, \mathbb{R})$:

$$\begin{aligned}\int_{[0,1]^d \times \mathbb{R}^N \times \mathbb{R}^d} \psi(x, y) \mathrm{d}\pi(u, x, y) &= \int_{\mathbb{R}^N \times \mathbb{R}^d} \psi(x, y) \mathrm{d}\nu(x, y) \\ &= \mathbb{E}(\psi(X, Y)),\end{aligned}$$

- the conditional expectation of x given u is 0, i.e., for every $b \in C([0,1]^d, \mathbb{R}^N)$:

$$\int_{[0,1]^d \times \mathbb{R}^N \times \mathbb{R}^d} b(u)^\top x \mathrm{d}\pi(u, x, y) = 0.$$

[11] here we assume that both X and Y are integrable

The dual of the linear program (80) then reads

$$\inf_{(\varphi,\psi,b)} \int_{[0,1]^d} \varphi \mathrm{d}\mu_d + \int_{\mathbb{R}^N \times \mathbb{R}^d} \psi(x,y) \mathrm{d}\nu(x,y) \tag{82}$$

subject to the pointwise constraint

$$\varphi(u) + b(u)^\top x + \psi(x,y) \geqslant u^\top y$$

given b and φ the lowest ψ fitting this constraint being the (convex in y) function

$$\psi(x,y) \stackrel{\text{def.}}{=} \sup_{u \in [0,1]^d} \{u^\top y - \varphi(u) - b(u)^\top x\}.$$

The existence of a solution (ψ, φ, b) to (82) is established in [Carlier et al., 2017] (under some assumptions on ν) and optimality for U in (80) is characterized by the pointwise complementary slackness condition

$$\varphi(U) + b(U)^\top X + \psi(X,Y) = U^\top Y \text{ almost surely.}$$

If φ and b were smooth we could deduce from the latter that

$$Y = \nabla\varphi(U) + Db(U)^\top U = \nabla\varphi_X(U), \quad \text{for } \varphi_x(u) \stackrel{\text{def.}}{=} \varphi(u) + b(u)^\top x$$

which is exactly (79). So specification of vector quantile regression is essentially the same as assuming this smoothness and the convexity of $u \mapsto \varphi_x(u) \stackrel{\text{def.}}{=} \varphi(u) + b(u)^\top x$. In general, these properties cannot be taken for granted and what can be deduced from complementary slackness is given by the weaker relations

$$\varphi_X(U) = \varphi_X^{**}(U), \; Y \in \partial\varphi_X^{**}(U) \text{ almost surely,}$$

were φ_x^{**} is the convex envelope of φ_x (i.e. the largest convex function below φ_x), we refer the reader to [Carlier et al., 2017] for details.

5. Numerical Vector Quantile Regression

5.1. Regularized Vector Quantile Regression

We now turn to a discrete setting for implementation purposes, and consider data $(X_j, Y_j)_{j=1..J}$ distributed according to the empirical measure $\nu = \sum_{j=1}^J \nu_j \delta_{(x_j,y_j)}$, and a $[0,1]^d$-uniform sample $(U_i)_{i=1,\dots,I}$ with empirical

measure $\mu = \sum_{i=1}^I \mu_i \delta_{u_i}$. In this setting, the vector quantile regression primal (81) writes

$$\max_{\pi \in \mathbb{R}_+^{I \times J}} \sum_{i=1}^I \sum_{j=1}^J u_i^\top y_j \pi_{ij}$$

subject to marginal constraints $\forall j, \sum_i \pi_{ij} = \nu_j$ and $\forall i, \sum_j \pi_{ij} = \mu_i$ and the mean-independence constraint between X and U: $\forall i, \sum_j x_j \pi_{ij} = 0$. Its dual formulation (82) reads

$$\inf_{(\varphi_i)_i, (\psi_j)_j, (b_i)_i} \sum_{j=1}^J \psi_j \nu_j + \sum_{i=1}^I \varphi_i \mu_i$$

subject to the constraint

$$\forall i, j, \varphi_i + b_i^\top x_j + \psi_j \geqslant u_i^\top y_j.$$

Using the optimality condition $\varphi_i = \max_j u_i^\top y_j - b_i^\top x_j - \psi_j$, we obtain the unconstrained formulation

$$\inf_{(\psi_j)_j, (b_i)_i} \sum_j \psi_j \nu_j + \sum_i \mu_i \left(\max_j u_i^\top y_j - b_i^\top x_j - \psi_j \right).$$

Replacing the maximum with its smoothed version[12], given a small regularization parameter ε, yields the smooth convex minimization problem (see [Peyré and Cuturi, 2019] for more details in connection with entropic regularization of optimal transport), which we call the *Regularized Vector Quantile Regression* (RVQR) problem

$$\inf_{\psi_j, b_i} J(\psi, b) \stackrel{\text{def.}}{=} \sum_j \psi_j \nu_j + \varepsilon \sum_i \mu_i \log \left[\sum_j \exp\left(\frac{1}{\varepsilon} [u_i^\top y_j - b_i^\top x_j - \psi_j] \right) \right] \tag{83}$$

We then have the following duality result[13]:

[12] Recall that the softmax with regularization parameter $\varepsilon > 0$ of $(\alpha_1, \ldots, \alpha_J)$ is given by $\text{Softmax}_\varepsilon(\alpha_1, \ldots, \alpha_J) \stackrel{\text{def.}}{=} \varepsilon \log\left(\sum_{j=1}^J e^{\frac{\alpha_j}{\varepsilon}} \right)$.

[13] Which can be proved either by using the Fenchel-Rockafellar duality theorem (see [Rockafellar, 1974], Theorems 19-20) or by hand. Indeed, in the primal, there are only finitely many linear constraints and nonnegativity constraints are not binding because of the entropy. The existence of Lagrange multipliers for the equality constraints is then straightforward.

Theorem 8. *The RVQR problem*

$$\begin{aligned}\max_{\pi_{ij}\geqslant 0} \quad & \sum_{ij}\pi_{ij}\left(u_i^\top y_j\right) - \varepsilon\sum_{ij}\pi_{ij}(\log\pi_{ij}-1)\\ & \sum_j \pi_{ij} = \mu_i\\ & \sum_i \pi_{ij} = \nu_j\\ & \sum_j \pi_{ij}x_j = \sum_j \nu_j x_j\end{aligned}$$

has dual (83), or equivalently

$$\min_{\varphi_i,\psi_j,b_i}\sum_i \mu_i\varphi_i + \sum_j \psi_j\nu_j + \varepsilon\sum_{ij}\exp\left(\frac{1}{\varepsilon}[u_i^\top y_j - \varphi_i - b_i^\top x_j - \psi_j]\right).$$

Note that the objective J in (83) remains invariant under the two transformations

- $(b,\psi)\leftarrow(b+c,\psi-c^\top x)$ with $c\in\mathbb{R}^N$ is a constant translation vector,
- $\psi\leftarrow\psi+\lambda$ where $\lambda\in\mathbb{R}$ is a constant.

These two invariances enable us to fix the value of $b_1 = 0$ and (for instance) to chose λ in such a way that $\sum_{i,j}\exp\left(\frac{1}{\varepsilon}[u_i^\top y_j - b_i^\top x_j - \psi_j]\right)) = 1$.

Remark 3. This formulation is eligible for stochastic optimization techniques when the number of (X,Y) observations is very large. Stochastic optimization w.r.t. ψ can be performed using the stochastic averaged gradient algorithm [Genevay et al., 2016], for instance by considering the objective

$$\inf_{\psi,\varphi,b}\sum_j h_\varepsilon(x_j,y_j,\psi,\varphi,b)\nu_j$$

with $h_\varepsilon(x_j,y_j,\psi,\varphi,b) = \psi_j + \sum_i \mu_i\varphi_i + \varepsilon\sum_i\exp\left(\frac{1}{\varepsilon}[u_i^\top y_j - b_i^\top x_j - \psi_j - \varphi_i]\right)$. Such techniques are not needed to compute b since the number of U samples (i.e. the size of b) is set by the user.

5.2. Numerical Resolution

As already noted the objective J in (83) is convex[14] and smooth. Its gradient has the explicit form

$$\frac{\partial J}{\partial \psi_j} \stackrel{\text{def.}}{=} \nu_j - \sum_{i=1}^{I} \mu_i \frac{e^{\theta_{ij}}}{\sum_{k=1}^{J} e^{\theta_{ik}}} \text{ where } \theta_{ij} = \frac{1}{\varepsilon}[u_i^\top y_j - b_i^\top x_j - \psi_j] \tag{84}$$

and

$$\frac{\partial J}{\partial b_i} \stackrel{\text{def.}}{=} -\mu_i \frac{\sum_{k=1}^{J} x_k e^{\theta_{ik}}}{\sum_{k=1}^{J} e^{\theta_{ik}}}. \tag{85}$$

To solve (83) numerically, we therefore can use a gradient descent mehod. An efficient way to do it is to use Nesterov accelerated gradient algorithm see [Nesterov, 1983] and [Beck and Teboulle, 2009]. Note that if ψ, b solves (83), the fact that the partial derivatives in (84)-(85) vanish imply that the coupling

$$\alpha_{ij}^{\varepsilon} \stackrel{\text{def.}}{=} \mu_i \frac{e^{\theta_{ij}}}{\sum_{k=1}^{J} e^{\theta_{ik}}}$$

satisfies the constraint of fixed marginals and mean-independence of the primal problem. Since the index j corresponds to observations it is convenient to introduce for every $x \in \mathcal{X} \stackrel{\text{def.}}{=} \{x_1, \ldots, x_J\}$ and $y \in \mathcal{Y} \stackrel{\text{def.}}{=} \{y_1, \ldots y_j\}$ the probability

$$\pi^{\varepsilon}(x, y, u_i) \stackrel{\text{def.}}{=} \sum_{j \,:\, x_j = x,\, y_j = y} \alpha_{ij}^{\varepsilon}.$$

5.3. Numerical results

Quantiles computation. The discrete probability π^{ε} is an approximation (because of the regularization ε) of $\mathscr{L}(U, X, Y)$ where U solves (80). The corresponding approximate quantile $Q_X^{\varepsilon}(U)$ is given by $\mathbb{E}_{\pi^{\varepsilon}}[Y|X, U]$. In the above discrete setting, this yields

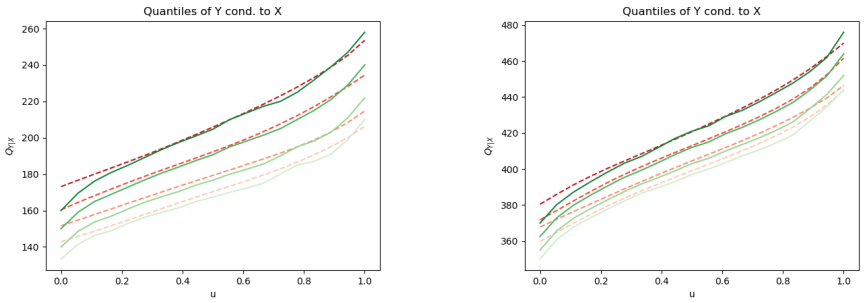
$$Q_x^{\varepsilon}(u_i) \stackrel{\text{def.}}{=} \mathbb{E}_{\pi^{\varepsilon}}[Y|X = x, U = u_i] = \sum_{y \in \mathcal{Y}} y \frac{\pi^{\varepsilon}(x, y, u_i)}{\sum_{y' \in \mathcal{Y}} \pi^{\varepsilon}(x, y', u_i)}.$$

Remark 4. To estimate the conditional distribution of Y given $U = u$ and $X = x$, we can use kernel methods. In the experiments, we compute approximate quantiles as means on neighborhoods of X values to make up for the lack of replicates. This amounts to considering $\mathbb{E}_{\pi^{\varepsilon}}[Y|X \in B_\eta(x), U = u_i]$ where $B_\eta(x)$ is a Euclidean ball of radius η centered on x.

[14] it is even strictly convex once we have chosen normalizations which take into account the two invariances of J explained above.

Empirical illstrations. We demonstrate the use of this approach on a series of health related experiments. We use the "ANSUR II" dataset (Anthropometric Survey of US Army Personnel), which can be found online[15]. This dataset is one of the most comprehensive publicly available data sets on body size and shape, containing 93 measurements for over 4,082 male adult US military personnel. It allows us to easily build multivariate dependent variables.

One-dimensional RVQR. We start by one-dimensional dependent variables ($d = 1$), namely *Weight* (Y_1) and *Thigh circumference* (Y_2), explained by X =(1, Height), to allow for comparison with classical quantile regression of [Koenker and Bassett, 1978]. Figure 11 displays results of our method compared to the classical approach, for different height quantiles (10%, 30%, 60%, 90%). Figure 11 is computed with a "soft" potential φ while Table 10 depicts the difference with its "hard" counterpart (see the beginning of section 5.1). Figure 12 and Table 11 detail the impact of regularization strength on these quantiles.



First dimension Second dimension

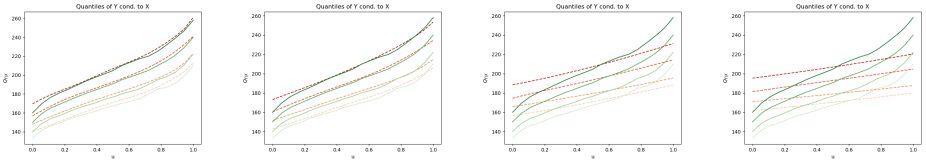
Figure 11: Comparison between one-dimensional RVQR (regularized dual in dashed red, with a "soft" φ) and classical approach (green) with (i) Y_1 =Weight (Left) or (ii) Y_2 =Thigh circumference and X =(1, Height). Quantiles are plotted for different height quantiles (10%, 30%, 60%, 90%). Regularization strengths are $\varepsilon = 0.1$. Chosen grid size is $n = 20$.

Multi-dimensional RVQR. In contrast, multivariate quantile regression explains the joint dependence $Y = (Y_1, Y_2)$ by X =(1,Height). Figures 14 and

[15]https://www.openlab.psu.edu/ansur2/

ε	0.05	0.1	0.5	1
$\frac{\|Q_{soft}-Q_{hard}\|_2}{\|Q_{soft}\|_2}$, $X = 10\%$	$3.8{\cdot}10^{-3}$	$1.5{\cdot}10^{-2}$	$6.7{\cdot}10^{-2}$	$9.2{\cdot}10^{-2}$
$\frac{\|Q_{soft}-Q_{hard}\|_2}{\|Q_{soft}\|_2}$, $X = 30\%$	$6.8{\cdot}10^{-3}$	$1.9{\cdot}10^{-2}$	$7.0{\cdot}10^{-2}$	$9.3{\cdot}10^{-2}$
$\frac{\|Q_{soft}-Q_{hard}\|_2}{\|Q_{soft}\|_2}$, $X = 60\%$	$1.2{\cdot}10^{-2}$	$2.0{\cdot}10^{-2}$	$6.9{\cdot}10^{-2}$	$9.5{\cdot}10^{-2}$
$\frac{\|Q_{soft}-Q_{hard}\|_2}{\|Q_{soft}\|_2}$, $X = 90\%$	$1.6{\cdot}10^{-2}$	$2.3{\cdot}10^{-2}$	$6.8{\cdot}10^{-2}$	$9.5{\cdot}10^{-2}$

Table 10: Relative error between one-dimensional RVQR with a "soft" computation of φ and its "hard" counterpart, with Y_1 =Weight and X =(1, Height) for different height quantiles (10%, 30%, 60%, 90%), depending on regularization strengths ε. Chosen grid size is $n = 20$.



$\varepsilon = 0.05$ $\varepsilon = 0.1$ $\varepsilon = 0.5$ $\varepsilon = 1$

Figure 12: One-dimensional RVQR, dual (dashed red) compared to classical QR (green) with Y_1 =Weight regressed on X =(1, Height), for varying regularization strengths ε. Quantiles are plotted for different height quantiles (10%, 30%, 60%, 90%). Chosen grid size is $n = 20$.

15 (each corresponding to an explained component, either Y_1 or Y_2) depicts how smoothing operates in higher dimension for different Height quantiles (10%, 50% and 90%), compared to a previous unregularized approach [Carlier et al., 2016b]. Figure 13 details computational times in 2D using an Intel(R) Core(TM) i7-7500U CPU 2.70GHz.

6. Statistical Analysis

In this section, we turn to the asymptotic analysis of the finite-dimensional RVQR dual potentials $v = (\psi, \varphi, b)$ in the finite-sample case, namely, whenever the data measure ν is accessed through an iid sample $X_1, \ldots, X_n \sim \nu$.

6.1. Regularity of Dual Potentials

For that purpose, regularity of the RVQR objective with respect to ν is first tackled. This section shows that the RVQR dual potentials $v = (\psi, \varphi, b)$

ε	0.05	0.1	0.5	1
$\frac{\|Q_{QR}-Q_{RVQR}\|_2}{\|Q_{QR}\|_2}$, $X = 10\%$	$9.8{\cdot}10^{-3}$	$9.8{\cdot}10^{-3}$	$2.8{\cdot}10^{-2}$	$3.8{\cdot}10^{-2}$
$\frac{\|Q_{QR}-Q_{RVQR}\|_2}{\|Q_{QR}\|_2}$, $X = 30\%$	$8.5{\cdot}10^{-3}$	$1.1{\cdot}10^{-2}$	$3.3{\cdot}10^{-2}$	$4.3{\cdot}10^{-2}$
$\frac{\|Q_{QR}-Q_{RVQR}\|_2}{\|Q_{QR}\|_2}$, $X = 60\%$	$7.7{\cdot}10^{-3}$	$9.3{\cdot}10^{-3}$	$3.1{\cdot}10^{-2}$	$4.4{\cdot}10^{-2}$
$\frac{\|Q_{QR}-Q_{RVQR}\|_2}{\|Q_{QR}\|_2}$, $X = 90\%$	$8.2{\cdot}10^{-3}$	$1.0{\cdot}10^{-2}$	$3.5{\cdot}10^{-2}$	$4.9{\cdot}10^{-2}$

Table 11: Relative error between one-dimensional RVQR and classical QR approach with Y_1 =Weight and X =(1, Height) for different height quantiles (10%, 30%, 60%, 90%), depending on regularization strengths ε. Chosen grid size is $n = 20$.

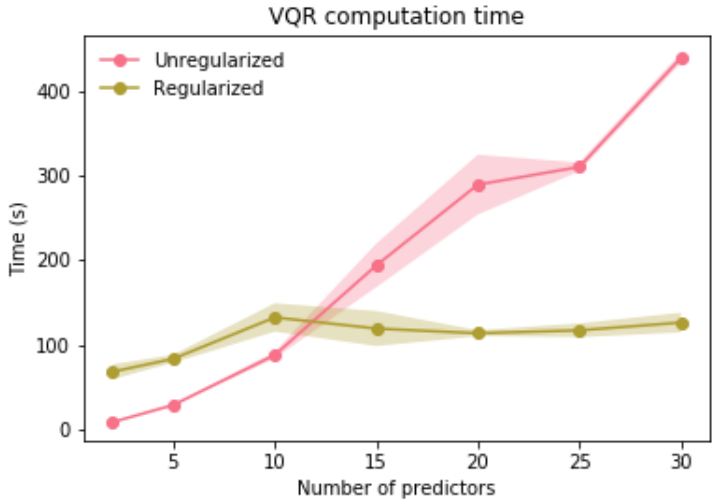


Figure 13: Comparison of computational times between the unregularized case (using Gurobi's barrier logging) and the regularized case, for a varying number of predictors in 2D. In the latter, this time represents the time to reach an error of 10^{-5} in $\|\cdot\|_2$ between two iterates of the transport plan for $\varepsilon = 0.1$. Chosen grid size is $n = 10$ (per axis).

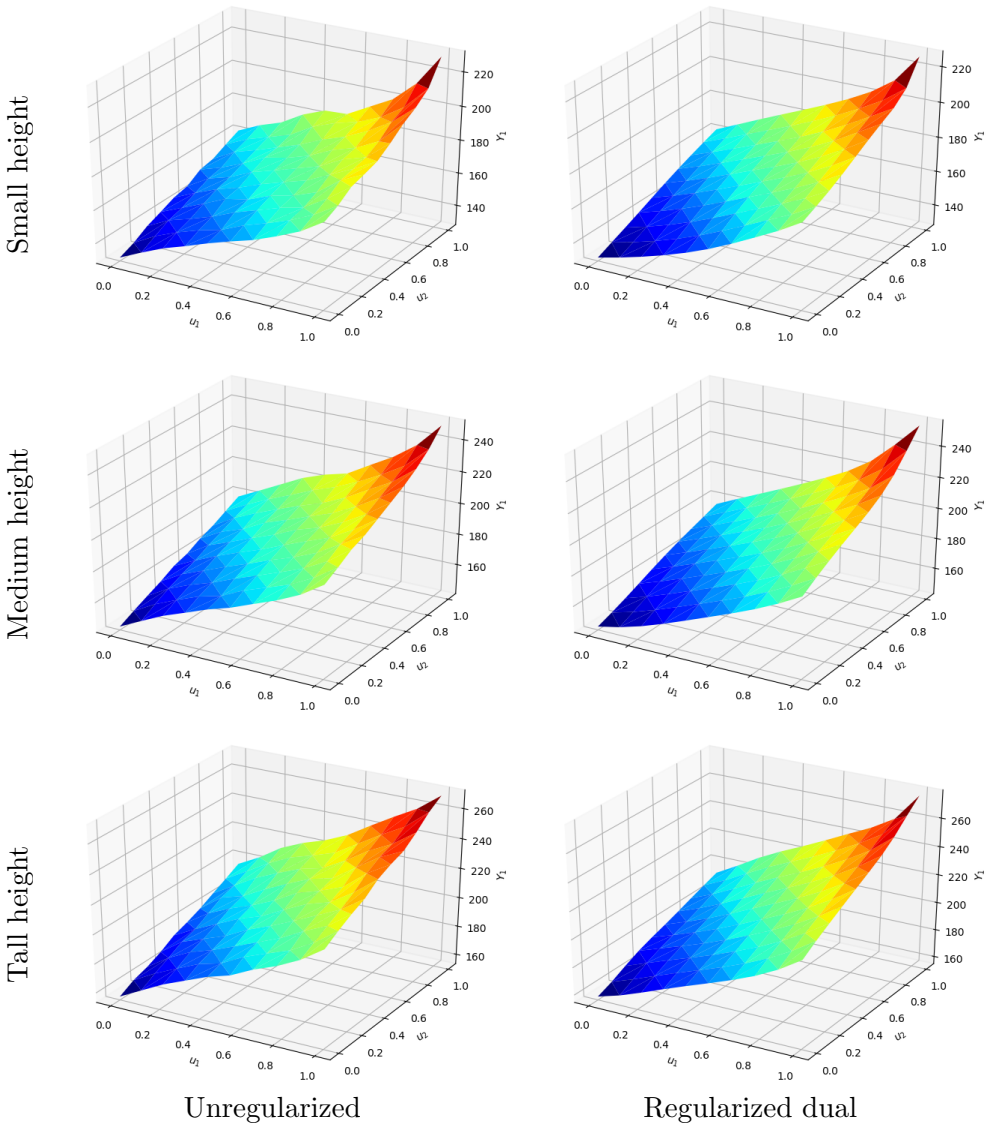


Figure 14: Two-dimensional RVQR of Y =(Weight, Thigh) explained by X =(1, Height). Quantiles of Y_1=Weight are plotted for different height quantiles: 10% (Bottom), 50% (Middle) and 90% (Top). Chosen grid size is $n = 10$ (per axis) and regularization strength $\varepsilon = 0.1$.

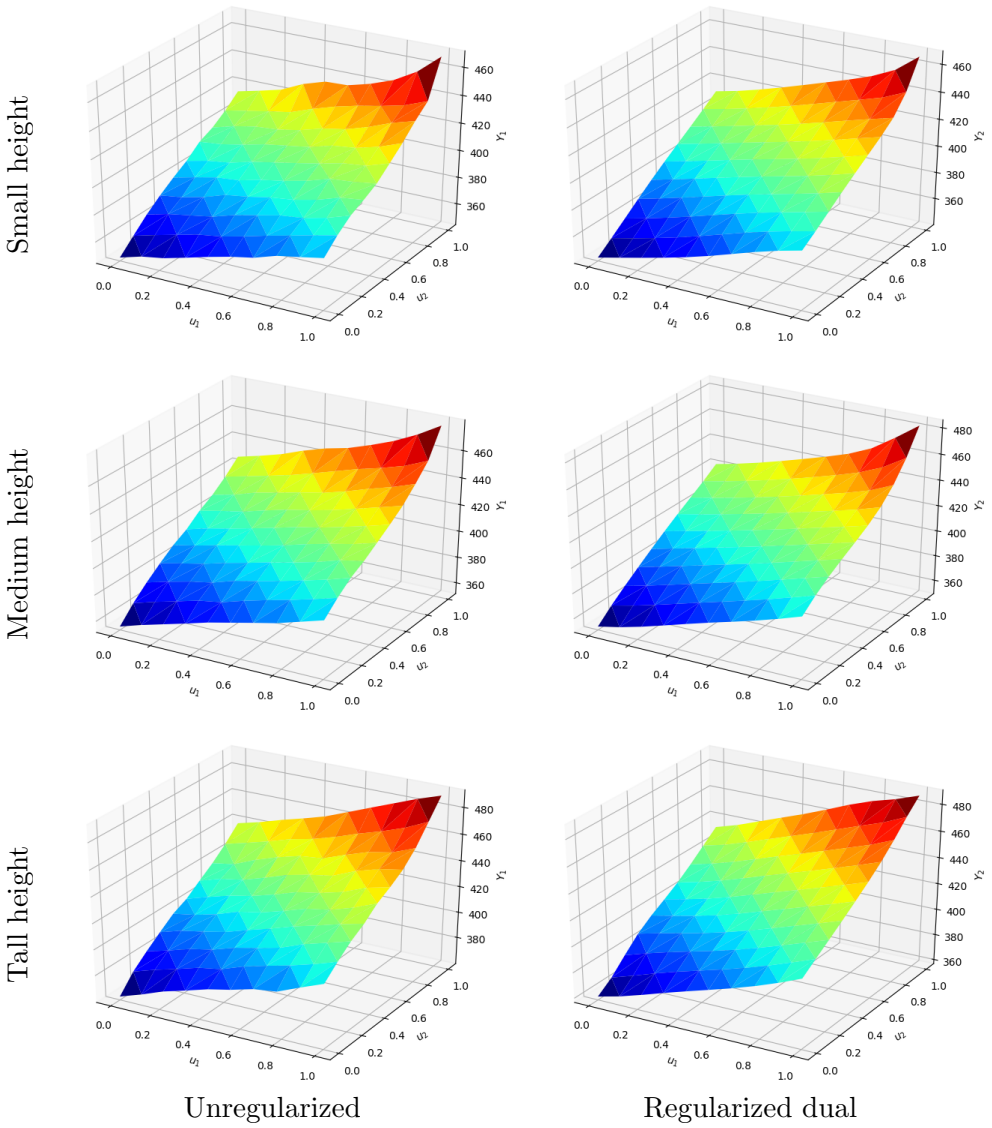


Figure 15: Two-dimensional RVQR of Y =(Weight, Thigh) explained by X =(1, Height). Quantiles of Y_2=Thigh are plotted for different height quantiles: 10% (Bottom), 50% (Middle) and 90% (Top). Chosen grid size is $n = 10$ (per axis) and regularization strength $\varepsilon = 0.1$.

are unique and regular (C^2) with respect to the data measure ν, as long as all states are observed. The main assumptions used in this section are the following ones:

- $(A1)$ Potentials are normalized: $\psi_1 = 0, b_1 = 0$
- $(A2)$ The data points $(x_j)_j$ generate $\mathbb{R}^N$: $\dim Vect(x_1, \ldots, x_J) = N$
- $(A3)$ All states are observed: $\forall i, j, \mu_i \nu_j > 0$
- $(A4)$ The data is centered: $\mathbb{E}(X) = \sum_j x_j \nu_j = 0$

We consider the second formulation of the RVQR problem, equivalent to (83), namely

$$\min_{v=(\psi,\varphi,b)} \sum_j \psi_j \nu_j + \sum_i \varphi_i \mu_i + \varepsilon \sum_{i,j} \exp\Big(\frac{1}{\varepsilon}(u_i^\top y_j - \varphi_i - \psi_j - b_i^\top x_j)\Big) \quad (86)$$

which, for simplicity, is also denoted $J(v)$.

Denoting $[K^\varepsilon]_{ij} = \varepsilon \exp\left(\frac{u_i^\top y_j}{\varepsilon}\right)$ and $\Lambda : v \in \mathbb{R}^{J+I+IN} \mapsto b_i^\top x_j + \varphi_i + \psi_j \in \mathbb{R}^{I \times J}$, (86) writes

$$\min_v \langle \Lambda(v), \mu \otimes \nu \rangle + \langle K^\varepsilon, e^{-\Lambda(v)} \rangle$$

as long as $(A4)$ holds. Using $F_\nu : M \in \mathbb{R}^{I \times J} \mapsto \langle M, \mu \otimes \nu \rangle + \langle K^\varepsilon, e^{-M} \rangle \in \mathbb{R}$, it yields

$$\min_v F_\nu \circ \Lambda(v)$$

Under some assumptions, Λ is injective:

Lemma 8. *Under assumptions (A1), (A2), (A4), function Λ is injective.*

Proof. If $\forall i, j, \varphi_i + \psi_j + b_i^\top x_j = 0$, then assumption $(A4)$ yields $\varphi_i = -\sum_j \psi_j \nu_j$. Moreover, by $(A1)$, normalization $b_1 = 0$ gives $\psi_j = -\varphi_1$, hence by $\psi_1 = 0, \psi_j = \varphi_i = 0$. Finally with $\forall i, j, b_i^\top x_j = 0$, assumption $(A2)$ yields $b_i = 0$. □

This allows to obtain the desired regularity:

Proposition 13. *Under assumptions (A1) to (A4), there exists a unique solution $v = (\psi, \varphi, b)$ to the dual problem (86), which is regular in ν: $v = h(\nu)$, where $h : D_h \overset{\text{def.}}{=} \{(\nu_j)_{j=1\ldots J} : \forall j, \nu_j > 0, \sum_j \nu_j = 1\} \to \mathbb{R}^{I+J+IN}$ is C^2.*

Proof. Problem (86) is strictly convex (its hessian is positive definite for all x as shown in the following) so it has at most one solution. J is moreover continuous and coercive hence it has a unique minimizer. First order conditions are the following $J + I + IN$ equations:

$$\nabla_v J(v) = \Lambda^\top \nabla F_\nu(\Lambda(v)) = 0 \tag{87}$$

which defines an implicit relation between v and ν, namely

$$g(\nu, v) = \Lambda^\top \nabla F_\nu(\Lambda(v)) = 0$$

Note that the hessian is invertible:

$$\nabla_{v,v} J(v) = \nabla_v g(\nu, v) = \Lambda^\top \nabla_{M,M} F_\nu(\Lambda(v))\Lambda \tag{88}$$

Indeed, the hessian $\nabla_{M,M} F_\nu(M)$ is diagonal with eigenvalues $K_{ij}^\varepsilon e^{-m_{ij}} > 0$; hence the hessian $\Lambda^\top \nabla_{M,M} F_\nu(\Lambda(v))\Lambda$ is also positive definite since Λ is injective (Lemma 8). Therefore $\nabla_v g(\nu, v)$ is invertible. From that, the implicit function theorem is applicable: there exists a unique C^1 function $h : D_h = \{(\nu_j)_{j=1\dots J} > 0, \sum_j \nu_j = 1\} \to \mathbb{R}^{I+J+IN}$, such that $h(\nu) = v$ and its partial derivatives are given by

$$J_h(\nu) = -[\nabla_v g(\nu, h(\nu))]^{-1} \nabla_\nu g(\nu, h(\nu)). \tag{89}$$

Since h is C^1 and g is C^2, by local inversion, J_h is also C^1. □

6.2. Law of Large Numbers

We consider the empirical measure $\hat{\nu}_n$ generated by an iid sample $X_1, \dots, X_n \sim \nu$. Denoting $\nu = \sum_{j=1}^J \nu_j \delta_{x_j}$, its empirical counterpart $\hat{\nu}_n$ writes $\hat{\nu}_n \overset{\text{def.}}{=} \left(\frac{1}{n}\sum_{i=1}^n 1\{X_i = x_j\}\right)_{j=1\dots J}$. The multinomial covariance matrix $\Sigma(\nu)$ writes

$$\Sigma(\nu) \overset{\text{def.}}{=} \begin{pmatrix} \nu_1(1-\nu_1) & -\nu_1\nu_2 & \cdots & -\nu_1\nu_J \\ -\nu_1\nu_2 & \nu_2(1-\nu_2) & \cdots & -\nu_2\nu_J \\ \vdots & \vdots & \ddots & \vdots \\ -\nu_1\nu_J & -\nu_2\nu_J & \cdots & \nu_J(1-\nu_J) \end{pmatrix}$$

The following RVQR Law of Large Numbers holds:

Proposition 14. *The sample-based potentials* $(\hat{\psi}_n, \hat{\varphi}_n, \hat{b}_n) \overset{\text{def.}}{=} h(\hat{\nu}_n)$ *converge almost surely to the true potentials, namely* $(\hat{\psi}_n, \hat{\varphi}_n, \hat{b}_n) \xrightarrow{a.s.} (\psi, \varphi, b)$.

Proof. The Strong Law of Large Numbers (see Theorem 5.18 from [Wasserman, 2004]) gives $\hat{\nu}_n \xrightarrow{a.s.} \nu$. Since $h : D_h = \{(\nu_j)_{j=1\dots J} : \forall j, \nu_j > 0, \sum_j \nu_j = 1\} \to \mathbb{R}^{I+J+IN}$ is C^1 and $\hat{\nu}_n \in D_h$ for n large enough, $h(\hat{\nu}_n)$ is well defined for n large enough. Since $\mathbb{P}(\nu \in D_h) = 1$ by $(A3)$, the continuous mapping theorem (see Theorem 2.3 from [van der Vaart, 2000]) yields that $h(\hat{\nu}_n) \xrightarrow{a.s.} h(\nu) = (\psi, \varphi, b)$. □

6.3. Central Limit Theorem

The following RVQR Central Limit Theorem holds:

Proposition 15. *The sample-based potentials $\hat{v}_n = (\hat{\psi}_n, \hat{\varphi}_n, \hat{b}_n)$ are asymptotically Gaussian, namely*

$$\sqrt{n}\left(\begin{pmatrix}\hat{\psi}_n\\ \hat{\varphi}_n\\ \hat{b}_n\end{pmatrix} - \begin{pmatrix}\psi\\ \varphi\\ b\end{pmatrix}\right) \xrightarrow{\mathcal{L}} Z, Z \sim \mathcal{N}(0, J_h(\nu)\Sigma(\nu)J_h(\nu)^\top)$$

where $\Sigma(\nu)$ is the (unknown) multinomial covariance matrix; $J_h(\nu)$ is the $(I + J + IN) \times J$ Jacobian matrix of h (see Proposition 13).

Proof. $n\hat{\nu}_n$ is an n-sized sample of a multinomial distribution with probability ν, hence Theorem 14.6 from [Wasserman, 2004] gives

$$\sqrt{n}(\hat{\nu}_n - \nu) \xrightarrow{\mathcal{L}} Y, Y \sim \mathcal{N}(0, \Sigma(\nu))$$

Since $h : D_h = \{(\nu_j)_{j=1\dots J} : \forall j, \nu_j > 0, \sum_j \nu_j = 1\} \to \mathbb{R}^{I+J+IN}$ is differentiable at ν, and $\hat{\nu}_n$ take their values in D_h for n large enough, the Delta Method (see Theorem 3.1 from [van der Vaart, 2000]) yields

$$\sqrt{n}(h(\hat{\nu}_n) - h(\nu)) \xrightarrow{\mathcal{L}} J_h(\nu)Y \overset{\text{def.}}{=} Z, Z \sim \mathcal{N}(0, J_h(\nu)\Sigma(\nu)J_h(\nu)^\top)$$

□

In practice, the covariance matrix $J_h(\nu)\Sigma(\nu)J_h(\nu)^\top$ being unknown, it has to be estimated, for instance using $J_h(\hat{\nu}_n)\Sigma(\hat{\nu}_n)J_h(\hat{\nu}_n)^\top$.

Conclusion

In this thesis, we have developed methods to perform machine learning and statistical estimation over the space of measures. Before detailing some avenues for future work, we would like to zoom on two salient features common to these approaches.

1. Summary of salient features

Learning from probability measures. Modeling data as probability distributions is the central topic of this thesis, but we would like to stress that they play different roles, whether it be

(i) *input objects to neural network architectures:* we show that considering input probability measures in their Lagrangian form in neural architectures (Chapter 1, Section 4) provides a geometric representation that can take into account invariance properties (Chapter 2, Section 5), and that alleviates the computational burden linked to Eulerian representations. This representation is characterized by an adaptability to a wide variety of applicative settings, from census, computer vision, biology and chemical data, to a suitable design of datasets, for instance in the context of automated machine learning. The resort to pairwise interaction functionals or their tensorized counterparts allows for the construction of universal approximators that are robust to input perturbations.

(ii) *output objects to neural network architectures:* the designed architectures are able to output probability measures as well, whether it be for generative or dynamic prediction purposes (Chapter 1, Section 4). This macroscopic representation is particularly well suited to contemporary challenges of limiting experimental costs or including privacy constraints. Such functionals require tailored layouts including measure-based loss functions to be learnt, such as the entropic Wasserstein distance.

(iii) *objects of interest for inference:* we demonstrate that the computation of the regularized transport plan eases processing and rendering of multivariate quantiles in the VQR framework (Chapter 3, Section 6). The obtained RVQR approach benefits from a computationally-friendly way to go beyond current pipelines while retrieving classical quantiles

[Koenker and Bassett, 1978] in the one dimensional setting, as well as unregularized quantiles in higher dimension [Carlier et al., 2016b].

Entropic optimal transport for high dimensional learning. Critical desirable components are unlocked by the use of entropic optimal transport in machine learning and statistics, from

(i) *scalability:* the structure of the regularized dual problem makes it a good candidate for learning, whether it be as a loss function for neural architectures with measure outputs (Chapter 1, Section 4) thanks to the GPU-friendliness of Sinhkorn's algorithm, or as an objective to retrieve conditional multivariate quantiles (Chapter 3, Section 6). As such, it allows processing probability measures in a parallelizable fashion as well as making RVQR amenable to high dimensional settings.

(ii) *differentiability:* the addition of an entropic term to the original problem allows to frame learning over distributions as differentiable programming, which can be performed using automatic differentiation (Chapter 1, Section 4 and Chater 2, Section 5) or accelerated gradient descent (Chapter 3, Section 6) in an optimal fashion.

(iii) *statistical properties:* the resort to the entropy enables to break the curse of dimensionality [Genevay et al., 2019, Mena and Niles-Weed, 2019] as well as to retrieve desired and expected asymptotic properties of multivariate quantiles in the finite sample case, yielding a law of large numbers and a central limit theorem for dual regularized potentials, that paves the way for hypothesis testing in the RVQR setting (Chapter 3, Section 6).

2. Perspectives for Future Work

Probability distribution-based neural networks. A natural extension of Chapter 1 lies in considering mass-varying measures in neural architectures, which falls under the scope of unbalanced optimal transport [Chizat, 2017]. It is motivated by a wide variety of applicative settings, from shapes and image processing, statistical learning, economic applications to evolution partial differential equations (PDEs). Such a development requires an alternative measure representation, since the modulation operation that allows for mass variation is not Lagrangian differentiable. Another important extension consists in investigating alternative representations such as Gaussian mixtures,

which would be useful in to build more expressive models with applications in biology or chemistry [Ficklin et al., 2017].

From a theoretical perspective, an important avenue to extend our contributions of Chapter 2 is to consider broader classes of invariants than products of permutations, which would be relevant for applications to shape or image processing or to deal with graph features. The extension to the equivariant case is also left for future work. The investigation of generalization bounds would also complement our universal approximation statements in quantifying the predictive performance of our networks. Computational perspectives include (i) tackling performance learning over broader sets of ML configurations; (ii) increasing expressiveness of the meta-features, for instance by going beyond their Euclidean nature; (iii) investigating their adaptability to new tasks, for which the probability distribution representation of tasks may be well suited [Finn et al., 2017, 2018].

More broadly, an important avenue is to explore the application of the methods developed in Chapter 1 and Chapter 2. In particular, promising applications include (i) domain adaptation [Courty et al., 2014], to transfer knowledge from a source domain to a target domain with possibly different marginal distributions and different tasks. Our method could be readily applied as it combines two already successful strategies used in the literature, namely reweighting strategies and gradual distortions to align distributions; (ii) a novel class of particle-based PDE solvers with applications to population dynamics. Our method extends already popular neural network-based approximate PDE solvers [Chen et al., 2018] to Lagrangian discretizations which are particularly well suited to populations dynamics.

Regularized multivariate quantile regression. The extension of multivariate quantile regression through the RVQR program developed in Chapter 3 yields several perspectives. On the statistical side, the preliminary study that has been presented opens the way to the design of hypothesis testing, the analysis of the infinite-dimensional setting, a more quantitative assessment of the error made on the dual potentials in the finite sample case, as well as investigations of both regularity with respect to the regularizing strength ε, and of the limit case $\varepsilon \to 0$, in which the asymptotic normality cannot be taken for granted. While we restricted the analysis to the case of an arbitrarily fixed ε, the idea of automatically selecting its value, for instance through an estimation procedure, would deepen the analysis. The idea has notably been investigated in the case of regularized Wasserstein

barycenters [Bigot et al., 2019b]. Provided that some additional data is gathered, for instance on the empirical joint measure $\hat{\pi}_{ij}$, estimation of ε could also be performed together with the dual potentials, for instance by maximum likelihood. Observation of $\hat{\pi}_{ij}$ is however not obvious, but could be done in application-dependent situations by setting arbitrary level curves using prior empirical knowledge (for instance, in healthcare applications). On the computational side, the regularized problem has been shown to be eligible for stochastic optimization techniques, suitable when the number of observations is too large, that are good candidates for GPU implementations. Comparison with the RVQR primal also yields several perspectives. The idea of solving the problem using alternating Bregman projections [Benamou et al., 2015] is appealing, however the Kullback-Leibler projection onto the mean-independence constraint is not in closed form. The issue can be circumvented by resorting to auxiliary variables, but it remains unclear how much these would be regularized compared to the entropy.

Beyond that, the idea of resorting to neural networks to model dual potentials in the RVQR setting is also promising. The objective of estimating a Monge map by a neural network and its ability to generalize beyond the original support [Seguy et al., 2018] may prove useful to make up for missing data or the presence of censorship, which could also help quantify quantile treatment effects. Moreover, neural network models based on recurrent-like mechanisms may be well suited to extend RVQR to cope with spatio-temporal data. The idea has begun to be investigated in the univariate case, for instance [Rodrigues and Pereira, 2020], which shares similarities in flavor to the recurrent mechanisms we introduced in our distribution-based networks.

Bibliography

P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas. Learning representations and generative models for 3D point clouds. *Proceedings of the 35th International Conference on Machine Learning*, 80:40–49, 10–15 Jul 2018.

M. Agueh and G. Carlier. Barycenters in the wasserstein space. *SIAM J. Math. Anal.*, 43(2):904–924, 2011.

M. Albooyeh, D. Bertolini, and S. Ravanbakhsh. Incidence networks for geometric deep learning. *Proceedings of the 37th International Conference on Machine Learning*, 2020.

J. Altschuler, J. Weed, and P. Rigollet. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 1961–1971, 2017.

D. Alvarez-Melis, T. Jaakkola, and S. Jegelka. Structured optimal transport. *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 84:1771–1780, 09–11 Apr 2018.

L. Ambrosio, N. Gigli, and G. Savare. Gradient flows: in metric spaces and in the space of probability measures. *Springer Science & Business Media*, 2008.

M. Arellano and S. Bonhomme. Nonlinear panel data estimation via quantile regressions. *Econometrics Journal*, 19(3):61–94, October 2016.

M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. *Proceedings of the 34th International Conference on Machine Learning*, 70:214–223, 06–11 Aug 2017.

K. Arrow and M. Hoffenberg. A time series analysis of interindustry demands. *North Holland Publishing Company*, 1959.

P. Austin, J. Tu, P. Daly, and D. Alter. The use of quantile regression in health care research: A case study examining gender differences in the timeliness of thrombolytic therapy. *Statistics in medicine*, 24:791–816, 03 2005.

S. Azagba and M. Sharaf. Fruit and vegetable consumption and body mass index: A quantile regression approach. *Journal of primary care & community health*, 3:210–220, 07 2012.

R. Bardenet, M. Brendel, B. Kégl, and M. Sebag. Collaborative hyperparameter tuning. *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, page II–199–II–207, 2013.

F. Bassetti, A. Bodini, and E. Regazzini. On minimum kantorovich distance estimators. *Statistics & probability letters*, 76(12):1298–1302, 2006.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Img. Sci.*, 2(1):183–202, Mar. 2009.

D. Belanger, Y. Ovadia, M. Bileschi, and B. Meade. Representation learning for seismic hawkes processes. 2018.

A. Belloni and R. L. Winkler. On multivariate quantiles under partial orders. *Ann. Statist.*, 39(2):1125–1179, 2011.

A. Belloni, V. Chernozhukov, and K. Kato. Uniform post-selection inference for least absolute deviation regression and other Z-estimation problems. *Biometrika*, 102(1):77–94, 12 2014. ISSN 0006-3444.

J.-D. Benamou, G. Carlier, M. Cuturi, L. Nenna, and G. Peyré. Iterative bregman projections for regularized transportation problems. *SIAM Journal on Scientific Computing*, 2015.

J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyperparameter optimization. *Advances in Neural Information Processing Systems 24*, pages 2546–2554, 2011.

E. Bernton, P. Jacob, M. Gerber, and C. Robert. Inference in generative models using the Wasserstein distance. working paper or preprint, May 2017.

D. P. Bertsekas. A new algorithm for the assignment problem. *Mathematical Programming*, 21(1):152–171, 1981.

D. Bertsimas and J. Tsitsiklis. Introduction to linear optimization. *Athena Scientific*, 1997.

J. Bigot and T. Klein. Characterization of barycenters in the wasserstein space by averaging optimal transport maps. *ESAIM: Probability and Statistics*, 22, 07 2015.

J. Bigot, E. Cazelles, and N. Papadakis. Central limit theorems for entropy-regularized optimal transport on finite spaces and statistical applications. *Electron. J. Statist.*, 13(2):5120–5150, 2019a.

J. Bigot, E. Cazelles, and N. Papadakis. Data-driven regularization of Wasserstein barycenters with an application to multivariate density registration. *Information and Inference: A Journal of the IMA*, 8(4):719–755, 11 2019b.

B. Bischl, G. Casalicchio, M. Feurer, F. Hutter, M. Lang, R. G. Mantovani, J. N. van Rijn, and J. Vanschoren. Openml benchmarking suites. *arXiv preprint arXiv:1708.03731*, 2019.

C. Borchardt and C. Jacobi. De investigando ordine systematis aequationum differentialium vulgarium cujuscunque. *Borchardt Journal für die reine und angewandte Mathematik*, 24:297–320, 1865.

P. Brazdil, C. Giraud-Carrier, C. Soares, and R. Vilalta. Metalearning: Applications to data mining. *Springer Publishing Company, Incorporated*, 2008.

Y. Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Comm. Pure Appl. Math.*, 44(4):375–417, 1991.

J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1872–1886, Aug. 2013. ISSN 0162-8828.

J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun. Spectral networks and locally connected networks on graphs. *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*, 2014.

M. Buchinsky. Changes in the U.S. Wage Structure 1963-1987: Application of Quantile Regression. *Econometrica*, 62(2):405–458, March 1994.

M. Buchinsky. The dynamics of changes in the female wage distribution in the USA: a quantile regression approach. *Journal of Applied Econometrics*, 13(1):1–30, 1998.

B. Cade and B. Noon. A gentle introduction to quantile regression for ecologists. *Frontiers in Ecology and the Environment*, 1:412–420, 10 2003.

L. Caffarelli. The regularity of mappings with a convex potential. *Journal of The American Mathematical Society - J AMER MATH SOC*, 5, 01 1992.

G. Canas and L. Rosasco. Learning probability measures with respect to optimal transport metrics. *Advances in Neural Information Processing Systems 25*, pages 2492–2500, 2012.

G. Carlier, A. Galichon, and F. Santambrogio. From Knothe's transport to Brenier's map and a continuation method for optimal transport. Oct. 2008.

G. Carlier, V. Chernozhukov, and A. Galichon. Vector quantile regression beyond correct specification. 2016a.

G. Carlier, V. Chernozhukov, and A. Galichon. Vector quantile regression: An optimal transport approach. *Annals of Statistics*, 44(3):1165–1192, June 2016b. ISSN 0090-5364.

G. Carlier, V. Chernozhukov, and A. Galichon. Vector quantile regression beyond the specified case. *J. Multivariate Anal.*, 161:96–102, 2017.

G. Carlier, V. Chernozhukov, G. D. Bie, and A. Galichon. Vector quantile regression and optimal transport, from theory to numerics. *Empirical Economics*, 2020.

G. Chamberlain. Quantile regression, censoring, and the structure of wages. *Advances in Econometrics: Sixth World Congress*, 1:171–210, 1994.

P. Chaudhuri. Nonparametric estimates of regression quantiles and their local bahadur representation. *The Annals of Statistics*, 19, 06 1991.

P. Chaudhuri. On a geometric notion of quantiles for multivariate data. *Journal of the American Statistical Association*, 91(434):862–872, 1996.

R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations. *Advances in Neural Information Processing Systems 31*, pages 6571–6583, 2018.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Rev.*, 43(1):129–159, Jan. 2001. ISSN 0036-1445.

X. Chen, X. Cheng, and S. Mallat. Unsupervised deep haar scattering on graphs. *Advances in Neural Information Processing Systems*, pages 1709–1717, 2014.

Z. Chen, S. Villar, L. Chen, and J. Bruna. On the equivalence between graph isomorphism testing and function approximation with gnns. *Advances in Neural Information Processing Systems 32*, pages 15894–15902, 2019.

X. Cheng, X. Chen, and S. Mallat. Deep haar scattering networks. *Information and Inference: A Journal of the IMA*, 5(2):105–133, 2016.

V. Chernozhukov and C. Hansen. The effects of 401(k) participation on the wealth distribution: An instrumental quantile regression analysis. *The Review of Economics and Statistics*, 86:735–751, 08 2004.

A. Chesher. Identification in nonseparable models. *Econometrica*, 71:1405–1441, 02 2003.

L. Chizat. Unbalanced optimal transport: Models, numerical methods, applications. *PhD thesis, PSL Research University*, 2017.

L. Chizat, P. Roussillon, F. Léger, F.-X. Vialard, and G. Peyré. Faster Wasserstein Distance Estimation with the Sinkhorn Divergence. In *Neural Information Processing Systems*, Advances in Neural Information Processing Systems, Vancouver, Canada, Dec. 2020.

T. Cohen and M. Welling. Group equivariant convolutional networks. *Proceedings of The 33rd International Conference on Machine Learning*, 48: 2990–2999, 20–22 Jun 2016.

N. Courty, R. Flamary, and D. Tuia. Domain adaptation with regularized optimal transport. *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, 2014.

N. Courty, R. Flamary, D. Tuia, and A. Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2017.

D. A. Cox, J. Little, and D. O'Shea. Ideals, varieties, and algorithms: An introduction to computational algebraic geometry and commutative algebra, 3/e (undergraduate texts in mathematics). *Springer-Verlag*, 2007.

I. Csiszar. i-divergence geometry of probability distributions and minimization problems. *Ann. Probab.*, 3(1):146–158, 02 1975.

F. Cucker and S. Smale. On the mathematics of emergence. *Japanese Journal of Mathematics*, 2(1):197–227, 2007.

M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems 26*, pages 2292–2300, 2013.

M. Cuturi and A. Doucet. Fast computation of wasserstein barycenters. volume 32 of *Proceedings of Machine Learning Research*, pages 685–693, Bejing, China, 22–24 Jun 2014. PMLR.

G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

G. De Bie, G. Peyré, and M. Cuturi. Stochastic deep networks. *International Conference on Machine Learning*, pages 1556–1565, 2019.

G. De Bie, H. Rakotoarison, G. Peyré, and M. Sebag. Distribution-based invariant deep networks for learning meta-features. (2006.13708), 2020.

M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, page 3844–3852, 2016.

E. del Barrio and J.-M. Loubes. Central limit theorems for empirical transportation cost in general dimension. *Ann. Probab.*, 47(2):926–951, 03 2019. doi: 10.1214/18-AOP1275.

E. del Barrio, J. A. Cuesta-Albertos, C. Matrán, and J. M. Rodríguez-Rodríguez. Tests of goodness of fit based on the l_2-wasserstein distance. *Ann. Statist.*, 27(4):1230–1239, 08 1999.

E. Del Barrio, E. Giné, and F. Utzet. Asymptotics for l 2 functionals of the empirical quantile process, with applications to tests of fit based on weighted wasserstein distances. *Bernoulli*, 11(1):131–189, 01 2005.

K. Doksum. Empirical probability plots and statistical inference for nonlinear models in the two-sample case. *The Annals of Statistics*, 2, 03 1974.

I. Drori, Y. Krishnamurthy, R. Rampin, R. Lourenco, J. One, K. Cho, C. Silva, and J. Freire. Alphad3m: Machine learning pipeline synthesis. *ICML International Workshop on Automated Machine Learning*, 2018.

N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song. Recurrent marked temporal point processes: Embedding event history to vector. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1555–1564, 2016.

D. Dua and C. Graff. UCI machine learning repository. 2017.

R. M. Dudley. The speed of mean glivenko-cantelli convergence. *Ann. Math. Statist.*, 40(1):40–50, 02 1969.

A. Dupuy and A. Galichon. Personality traits and the marriage market. *Journal of Political Economy*, 122(6):1271 – 1319, 2014.

P. Dvurechensky, A. Gasnikov, and A. Kroshnin. Computational optimal transport: Complexity by accelerated gradient descent is better than by sinkhorn's algorithm. *Proceedings of the 35th International Conference on Machine Learning*, 80:1367–1376, 10–15 Jul 2018.

F. Edgeworth. Xxii. on a new method of reducing observations relating to several quantities. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 25(154):184–191, 1888.

H. Edwards and A. J. Storkey. Towards a neural statistician. *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

E. Eide and M. H. Showalter. The effect of school quality on student performance: A quantile regression approach. *Economics Letters*, 58(3): 345–350, March 1998.

T. Elsken, J. H. Metzen, and F. Hutter. Neural architecture search: A survey. *J. Mach. Learn. Res.*, 20:55:1–55:21, 2019.

S. Erlander and N. F. Stewart. The gravity model in transportation analysis: theory and extensions. *Vsp*, 3, 1990.

S. N. Evans and F. A. Matsen. The phylogenetic kantorovich-rubinstein metric for environmental sequence samples. *T. Journal of the Royal Statistical Society. Series B, Statistical methodology*, 74:569–592, 2012.

H. Fan, H. Su, and L. J. Guibas. A point set generation network for 3d object reconstruction from a single image. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter. Efficient and robust automated machine learning. *Advances in Neural Information Processing Systems 28*, pages 2962–2970, 2015.

J. Feydy, T. Séjourné, F. Vialard, S. Amari, A. Trouvé, and G. Peyré. Interpolating between optimal transport and MMD using sinkhorn divergences. *The 22nd International Conference on Artificial Intelligence and Statistics,*

AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan, 89:2681–2690, 2019.

S. Ficklin, L. Dunwoodie, W. Poehlman, C. Watson, K. Roche, and F. Feltus. Discovering condition-specific gene co-expression patterns using gaussian mixture models: A cancer case study. *Nature Sci Rep*, 7, 2017.

A. Figalli. The monge-ampere equation and its applications. *Zurich Lectures in Advanced Mathematics. European Mathematical Society (EMS), Zurich.*, 2017.

C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

C. Finn, K. Xu, and S. Levine. Probabilistic model-agnostic meta-learning. *Advances in Neural Information Processing Systems 31*, pages 9516–9527, 2018.

R. Flamary and N. Courty. Pot python optimal transport library. 2017. URL `https://pythonot.github.io/`.

R. Flamary, C. Févotte, N. Courty, and V. Emyia. Optimal spectral transportation with application to music transcription. *Neural Information Processing Systems (NIPS)*, 2016.

M. Fox and H. Rubin. Admissibility of quantile estimates of a single location parameter. *Ann. Math. Statist.*, 35(3):1019–1030, 09 1964.

U. Frisch, S. Matarrese, R. Mohayaee, and A. Sobolevski. A reconstruction of the initial conditions of the universe by optimal mass transportation. *Nature*, 417:260–262, 2002.

C. Frogner, C. Zhang, H. Mobahi, M. Araya, and T. A. Poggio. Learning with a wasserstein loss. *Advances in Neural Information Processing Systems*, pages 2053–2061, 2015.

T. O. Gallouët and Q. Mérigot. A lagrangian scheme à la brenier for the incompressible euler equations. *Found. Comput. Math.*, 18(4):835–865, Aug. 2018. ISSN 1615-3375.

C. F. Gauss. Théorie du mouvement des corps sélestes parcourant des sections coniques autour du soleil. 1809.

C. F. Gauss. Anwendung der wahrscheinlichkeitsrechnung auf eine aufgabe der practischen geometrie. *Astronomische Nachrichten*, 1822.

A. Genevay, M. Cuturi, G. Peyré, and F. Bach. Stochastic optimization for large-scale optimal transport. *Advances in Neural Information Processing Systems*, pages 3440–3448, 2016.

A. Genevay, G. Peyré, and M. Cuturi. Learning generative models with sinkhorn divergences. *International Conference on Artificial Intelligence and Statistics*, pages 1608–1617, 2018.

A. Genevay, L. Chizat, F. Bach, M. Cuturi, and G. Peyré. Sample complexity of sinkhorn divergences. *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, 89:1574–1583, 2019.

R. Gens and P. M. Domingos. Deep symmetry networks. *Advances in Neural Information Processing Systems 27*, pages 2537–2545, 2014.

X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

M. Godin, A. K. Bryan, T. P. Burg, K. Babcock, and S. R. Manalis. Measuring the mass, density, and size of particles and cells using a suspended microchannel resonator. *Applied physics letters*, 91(12):123121, 2007.

A. V. Goldberg and R. E. Tarjan. Finding minimum-cost circulations by canceling negative cycles. *Journal of the ACM*, 36(4):873–886, 1989.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems 27*, pages 2672–2680, 2014.

P. Gordaliza, E. D. Barrio, G. Fabrice, and J.-M. Loubes. Obtaining fairness using optimal transport theory. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2357–2365, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

D. Graham, J. Wang, and S. Ravanbakhsh. Equivariant entity-relationship networks. 2019.

A. Gramfort, G. Peyré, and M. Cuturi. Fast Optimal Transport Averaging of Neuroimaging Data. *Information Processing in Medical Imaging (IPMI)*, June 2015.

A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. *Advances in Neural Information Processing Systems 19*, pages 513–520, 2007.

W. H. Grover, A. K. Bryan, M. Diez-Silva, S. Suresh, J. M. Higgins, and S. R. Manalis. Measuring single-cell density. *Proceedings of the National Academy of Sciences*, 108(27):10992–10996, 2011.

J. Guckenheimer, G. Oster, and A. Ipaktchi. The dynamics of density dependent population models. *Journal of Mathematical Biology*, 4(2): 101–147, 1977.

N. Guttenberg, N. Virgo, O. Witkowski, H. Aoki, and R. Kanai. Permutation-equivariant neural networks applied to dynamics prediction. *arXiv preprint arXiv:1612.04530*, 2016.

M. Hallin, D. Paindaveine, and M. Z'Siman. Multivariate quantiles and multiple-output regression quantiles: From l 1 optimization to halfspace depth. *Ann. Statist.*, 38(2):635–669, 04 2010.

M. Hallin, Z. Lu, D. Paindaveine, and M. Z'Siman. Local bilinear multiple-output quantile/depth regression. *Bernoulli*, 21(3):1435–1466, 08 2015.

J. Hartford, D. R. Graham, K. Leyton-Brown, and S. Ravanbakhsh. Deep models of interactions across sets. *Proceedings of the 35th International Conference on Machine Learning*, 2018.

T. Hashimoto, D. Gifford, and T. Jaakkola. Learning population-level diffusions with generative rnns. *Proceedings of The 33rd International Conference on Machine Learning*, 48:2417–2426, 20–22 Jun 2016.

M. Henaff, J. Bruna, and Y. LeCun. Deep convolutional networks on graph-structured data. *CoRR*, abs/1506.05163, 2015.

R. Herzig, M. Raboh, G. Chechik, J. Berant, and A. Globerson. Mapping images to scene graphs with permutation-invariant structured prediction. *Advances in Neural Information Processing Systems 31*, pages 7211–7221, 2018.

G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6):82–97, 2012.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

G. Huang, C. Guo, M. J. Kusner, Y. Sun, F. Sha, and K. Q. Weinberger. Supervised word mover's distance. *Advances in Neural Information Processing Systems*, pages 4862–4870, 2016.

F. Hutter, H. H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. *Proceedings of the 5th International Conference on Learning and Intelligent Optimization*, page 507–523, 2011.

F. Hutter, L. Kotthoff, and J. Vanschoren, editors. *Automated Machine Learning: Methods, Systems, Challenges*. Springer, 2018. In press, available at http://automl.org/book.

H. S. Jomaa, J. Grabocka, and L. Schmidt-Thieme. Dataset2vec: Learning dataset meta-features. 2019.

R. Jordan, D. Kinderlehrer, and F. Otto. The variational formulation of the fokker-planck equation. *SIAM journal on mathematical analysis*, 29:1–17, 1998.

L. Kantorovich. On the transfer of masses. *Doklady Akademii Nauk*, 37(2): 227–229, 1942.

K. Kato. Estimation in functional linear quantile regression. *Ann. Statist.*, 40(6):3108–3136, 12 2012.

N. Keriven and G. Peyré. Universal invariant and equivariant graph neural networks. *Advances in Neural Information Processing Systems 32*, pages 7090–7099, 2019.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

D. P. Kingma and M. Welling. Auto-encoding variational bayes. *ICLR*, 2014.

M. Klatt, C. Tameling, and A. Munk. Empirical regularized optimal transport: Statistical theory and applications. *SIAM J. Math. Data Sci.*, 2:419–443, 2020.

A. Klein, S. Falkner, S. Bartels, P. Hennig, and F. Hutter. Fast Bayesian Optimization of Machine Learning Hyperparameters on Large Datasets. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 54:528–536, 20–22 Apr 2017.

R. Koenker. Quantile regression. *Cambridge University Press*, 2005.

R. Koenker. Additive models for quantile regression: Model selection and confidence bandaids. *Braz. J. Probab. Stat.*, 25(3):239–262, 11 2011.

R. Koenker. Computational methods for quantile regression. *Handbook of Quantile Regression*, 2017.

R. Koenker and O. Geling. Reappraising medfly longevity. *Journal of the American Statistical Association*, 96(454):458–468, 2001. doi: 10.1198/016214501753168172. URL `https://doi.org/10.1198/016214501753168172`.

R. Koenker and K. F. Hallock. Quantile regression. *Journal of Economic Perspectives*, 15(4):143–156, December 2001.

R. Koenker, P. Ng, and S. Portnoy. Quantile smoothing splines. *Biometrika*, 81(4):673–680, 1994.

R. W. Koenker and G. Bassett. Regression quantiles. *Econometrica*, 46(1): 33–50, 1978.

V. I. Koltchinskii. M-estimation, convexity and quantiles. *Ann. Statist.*, 25 (2):435–477, 04 1997.

R. Kondor and S. Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. *Proceedings of the 35th International Conference on Machine Learning*, 2018.

R. Kondor, H. T. Son, H. Pan, B. Anderson, and S. Trivedi. Covariant compositional networks for learning graphs. 2018.

L. Kong and I. Mizera. Quantile tomography: using quantiles with multivariate data. *Statistica Sinica.*, 22:1589–1610, 2012.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, pages 1097–1105, 2012.

M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger. From word embeddings to document distances. *Proceedings of the 32nd International Conference on Machine Learning*, 37:957–966, 07–09 Jul 2015.

T. Le Gouic and J.-M. Loubes. Existence and consistency of wasserstein barycenters. *Probability Theory and Related Fields*, pages 1–17, 2016.

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, pages 2278–2324, 1998.

H. Lee, P. Pham, Y. Largman, and A. Y. Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. *Advances in Neural Information Processing Systems 22*, pages 1096–1104, 2009.

J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. *Proceedings of the 36th International Conference on Machine Learning*, 97: 3744–3753, 09–15 Jun 2019.

S. Lee. Efficient semiparametric estimation of a partially linear quantile regression model. *Econometric Theory*, 19:1–31, 02 2003.

A.-M. Legendre. Nouvelles méthodes pour la détermination des orbites des comètes. 1805.

E. Lehmann. Nonparametrics: statistical methods based on ranks. *Holden-Day Inc., San Francisco, Calif.*, 1974.

M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6):861–867, 1993.

B. Levy and E. Schwindt. Notions of optimal transport theory and how to implement them on a computer. *Computers & Graphics*, 72, 02 2018.

T.-H. Li. Laplace Periodogram for Time Series Analysis. *Journal of the American Statistical Association*, 103:757–768, June 2008.

J. A. F. Machado and J. Mata. Box-Cox quantile regression and the distribution of firm sizes. *Journal of Applied Econometrics*, 15(3):253–274, 2000.

H. Maron, H. Ben-Hamu, N. Shamir, and Y. Lipman. Invariant and equivariant graph networks. *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019a.

H. Maron, E. Fetaya, N. Segol, and Y. Lipman. On the universality of invariant networks. *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 4363–4371, 2019b.

H. Maron, O. Litany, G. Chechik, and E. Fetaya. On learning sets of symmetric elements. *Proceedings of the 37th International Conference on Machine Learning*, 2020.

R. L. Matzkin. Estimation of Nonparametric Models With Simultaneity. *Econometrica*, 83:1–66, January 2015.

R. J. McCann. Existence and uniqueness of monotone measure-preserving maps. *Duke Math. J.*, 80(2):309–323, 11 1995.

H. Mei and J. M. Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. *Advances in Neural Information Processing Systems*, pages 6754–6764, 2017.

B. Melly. Decomposition of differences in distribution using quantile regression. *Labour Economics*, 12(4):577–590, August 2005.

G. Mena and J. Niles-Weed. Statistical bounds for entropic optimal transport: sample complexity and the central limit theorem. *Advances in Neural Information Processing Systems 32*, pages 4541–4551, 2019.

A. Mensch and G. Peyré. Online sinkhorn: optimal transportation distances from sample streams. *Advances in Neural Information Processing Systems 33*, 2020.

M. A. Muñoz, L. Villanova, D. Baatar, and K. Smith-Miles. Instance spaces for machine learning classification. *Machine Learning*, 107(1):109–147, Jan. 2018. ISSN 0885-6125.

K. Muandet, K. Fukumizu, F. Dinuzzo, and B. Schölkopf. Learning from distributions via support measure machines. *Advances in Neural Information Processing Systems 25*, pages 10–18, 2012.

M. A. Muñoz, L. Villanova, D. Baatar, and K. Smith-Miles. Instance spaces for machine learning classification. *Machine Learning*, 107(1):109–147, 2018.

R. Murphy, B. Srinivasan, V. Rao, and B. Ribeiro. Janossy pooling: Learning deep permutation-invariant functions for variable-size inputs. *7th International Conference on Learning Representations, ICLR*, 2019.

F. Mémoli. Gromov-wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, 11(4):417–487, 2011.

Y. Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 3:543–547, 1983.

V. Perrone, R. Jenatton, M. W. Seeger, and C. Archambeau. Scalable hyperparameter transfer learning. *Advances in Neural Information Processing Systems 31*, pages 6845–6855, 2018.

T. Pevny and V. Kovarik. Approximation capability of neural networks on spaces of probability measures and tree-structured domains. *7th International Conference on Learning Representations*, 2019.

G. Peyré and M. Cuturi. Computational optimal transport. *Foundations and Trends in Machine Learning*, 11(5-6):355–607, 2019. ISSN 1935-8237.

G. Peyré, M. Cuturi, and J. Solomon. Gromov-wasserstein averaging of kernel and distance matrices. volume 48 of *Proceedings of Machine Learning Research*, pages 2664–2672. PMLR, 2016.

B. Pfahringer, H. Bensusan, and C. G. Giraud-Carrier. Meta-learning by landmarking various learning algorithms. *Proceedings of the Seventeenth International Conference on Machine Learning*, page 743–750, 2000.

F. Pitié, A. C. Kokaram, and R. Dahyot. Automated colour grading using colour distribution transfer. *Comput. Vis. Image Underst.*, 107(1-2):123–137, 2007.

B. Poczos, A. Singh, A. Rinaldo, and L. Wasserman. Distribution-free distribution regression. *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, 31:507–515, 29 Apr–01 May 2013.

S. Portnoy and R. Koenker. The gaussian hare and the laplacian tortoise: computability of squared-error versus absolute-error estimators. *Statist. Sci.*, 12(4):279–300, 11 1997.

J. L. Powell. Censored regression quantiles. *Journal of Econometrics*, 32(1): 143–155, 1986.

R. Pyke. Multidimensional empirical processes: some comments. *Stochastic Processes and Related Topics, Proceedings of the Summer Research Institute on Statistical Inference for Stochastic Processes*, 2:45–58, 1975.

C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017a.

C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 5105–5114, 2017b.

Q. I. Rahman and G. Schmeisser. Analytic theory of polynomials. *Oxford University Press*, 2002.

S. Rajaram, T. Graepel, and R. Herbrich. Poisson-networks: A model for structured point processes. *Proceedings of the 10th international workshop on artificial intelligence and statistics*, pages 277–284, 2005.

H. Rakotoarison, M. Schoenauer, and M. Sebag. Automated machine learning with monte-carlo tree search. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 3296–3303, 7 2019.

S. Ravanbakhsh, J. Schneider, and B. Poczos. Deep learning with sets and point clouds. *arXiv preprint arXiv:1611.04500*, 2016.

S. Ravanbakhsh, J. Schneider, and B. Póczos. Equivariance through parameter-sharing. *Proceedings of the 34th International Conference on Machine Learning*, 70:2892–2901, 2017.

J. R. Rice. The algorithm selection problem. *Advances in Computers*, 15: 65–118, 1976.

T. Rippl, A. Munk, and A. Sturm. Limit laws of the empirical wasserstein distance. *J. Multivar. Anal.*, 151(C):90–109, Oct. 2016.

R. T. Rockafellar. Conjugate duality and optimization. *SIAM*, 1974.

F. Rodrigues and F. C. Pereira. Beyond expectation: Deep joint mean and quantile regression for spatio-temporal problems. *IEEE transactions on neural networks and learning systems*, 2020.

A. Rolet, M. Cuturi, and G. Peyré. Fast dictionary learning with a smoothed wasserstein loss. *Artificial Intelligence and Statistics*, pages 630–638, 2016.

Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40 (2):99–121, 2000.

J. V. Ryff. Measure preserving transformations and rearrangements. *J. Math. Anal. Appl.*, 31:449–458, 1970.

F. Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 2015.

F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *Trans. Neur. Netw.*, 20(1):61–80, Jan. 2009. ISSN 1045-9227.

B. Schmitzer. A sparse multiscale algorithm for dense optimal transport. *J Math Imaging Vis*, 56:238–259, 2016.

D. Scieur, A. d'Aspremont, and F. Bach. Regularized nonlinear acceleration. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, page 712–720, 2016.

N. Segol and Y. Lipman. On universal equivariant set networks. *8th International Conference on Learning Representations*, 2020.

V. Seguy, B. B. Damodaran, R. Flamary, N. Courty, A. Rolet, and M. Blondel. Large-scale optimal transport and mapping estimation. *International Conference on Learning Representations (ICLR)*, 2018.

R. Serfling. Nonparametric multivariate descriptive measures based on spatial quantiles. *Journal of Statistical Planning and Inference*, 123:259–278, 07 2004.

J. Shawe-Taylor. Symmetries and discriminability in feedforward network architectures. *IEEE Transactions on Neural Networks*, 4(5):816–826, Sep. 1993. ISSN 1941-0093.

J. Shawe-Taylor. Symmetries and discriminability in feedforward network architectures. *IEEE Transactions on Neural Networks*, 4(5):816–826, 1993.

J. Solomon, F. de Goes, G. Peyré, M. Cuturi, A. Butscher, A. Nguyen, T. Du, and L. Guibas. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Trans. Graph.*, 34(4), July 2015. ISSN 0730-0301.

X. Song, G. Li, Z. Zhou, X. Wang, I. Ionita-Laza, and Y. Wei. QRank: a novel quantile regression tool for eQTL discovery. *Bioinformatics*, 33(14): 2123–2130, 03 2017.

J. T. Springenberg, A. Klein, S. Falkner, and F. Hutter. Bayesian optimization with robust bayesian neural networks. *Advances in Neural Information Processing Systems 29*, pages 4134–4142, 2016.

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, pages 3104–3112, 2014.

V. Tereshko. Reaction-diffusion model of a honeybee colony's foraging behaviour. *International Conference on Parallel Problem Solving from Nature*, pages 807–816, 2000.

C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown. Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 847–855, 2013.

J. Tukey. Mathematics and picturing data. *International congress of mathematics*, pages 523–531, 1975.

A. van der Vaart. *Asymptotic Statistics*. Asymptotic Statistics. Cambridge University Press, 2000. ISBN 9780521784504.

J. Vanschoren, J. N. van Rijn, B. Bischl, and L. Torgo. Openml: Networked science in machine learning. *SIGKDD Explorations*, 15(2):49–60, 2013.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems 30*, pages 5998–6008, 2017.

T. Vayer, L. Chapel, R. Flamary, R. Tavenard, and N. Courty. Fused gromov-wasserstein distance for structured objects. *Algorithms*, 13 (9): 212, 2020.

C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008. ISBN 9783540710509.

O. Vinyals, M. Fortunato, and N. Jaitly. Pointer networks. *Advances in Neural Information Processing Systems 28*, pages 2692–2700, 2015.

O. Vinyals, S. Bengio, and M. Kudlur. Order matters: Sequence to sequence for sets. *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

E. Wagstaff, F. Fuchs, M. Engelcke, I. Posner, and M. A. Osborne. On the limitations of representing functions on sets. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6487–6494, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

L. Wang, Y. Zhou, R. Song, and B. Sherwood. Quantile-optimal treatment regimes. *Journal of the American Statistical Association*, 113(523):1243–1254, 2018.

L. Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer Texts in Statistics. Springer, New York, 2004. ISBN 978-1-4419-2322-6. doi: 10.1007/978-0-387-21736-9.

J. Weed and F. Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. *Bernoulli*, 25(4A):2620–2648, 11 2019. doi: 10.3150/18-BEJ1065.

Y. Wei. An approach to multivariate covariate-dependent quantile contours with application to bivariate conditional growth charts. *Journal of the American Statistical Association*, 103(481):397–409, 2008.

Y. Wei, A. Ignatius, R. Koenker, and X. He. Quantile regression methods for reference growth charts. *Statistics in medicine*, 25:1369–82, 04 2006.

M. Weiler, M. Geiger, M. Welling, W. Boomsma, and T. S. Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems 31*, pages 10381–10392, 2018.

A. G. Wilson. The use of entropy maximising models, in the theory of trip distribution, mode split and route split. *Journal of transport economics and policy*, 1969.

D. H. Wolpert. The lack of A priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390, 1996.

J. Wood and J. Shawe-Taylor. Representation theory and invariant neural networks. *Discrete applied mathematics*, 69(1-2):33–60, 1996.

Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015a.

Z. Wu, S. Song, A. Khosla, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shape modeling. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015b.

S. Xiao, M. Farajtabar, X. Ye, J. Yan, L. Song, and H. Zha. Wasserstein learning of deep generative point process models. *Advances in Neural Information Processing Systems*, pages 3247–3257, 2017a.

S. Xiao, J. Yan, X. Yang, H. Zha, and S. M. Chu. Modeling the intensity function of point process via recurrent neural networks. *Proc. AAAI*, 17: 1597–1603, 2017b.

Z. Xiao and R. Koenker. Inference on the quantile regression process. *Econometrica*, 70:1583–1612, 02 2002.

K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? *7th International Conference on Learning Representations, ICLR*, pages 1–15, 2019.

X. Yang, N. N. Narisetty, and X. He. A new approach to censored quantile regression estimation. *Journal of Computational and Graphical Statistics*, 27(2):417–425, 2018.

L. Yi, W. Zhao, H. Wang, M. Sung, and L. J. Guibas. Gspn: Generative shape proposal network for 3d instance segmentation in point cloud. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

J. Yoon, T. Kim, O. Dia, S. Kim, Y. Bengio, and S. Ahn. Bayesian model-agnostic meta-learning. *Advances in Neural Information Processing Systems 31*, pages 7332–7342, 2018.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep sets. *Advances in Neural Information Processing Systems 30*, pages 3391–3401, 2017.

J. Zietz, E. Zietz, and G. Sirmans. Determinants of house prices: A quantile regression approach. *The Journal of Real Estate Finance and Economics*, 37:317–333, 02 2008.

RÉSUMÉ

Cette thèse propose des contributions théoriques et numériques pour effectuer des tâches d'apprentissage et de statistiques sur l'espace des mesures. Dans une première partie, nous introduisons une nouvelle classe de réseaux neuronaux qui traite les mesures de probabilité sous leur forme lagrangienne (obtenue par échantillonnage) à la fois comme entrées et sorties, qui se caractérise par sa robustesse et ses propriétés d'approximation universelle. Nous montrons que ce cadre peut être adapté pour effectuer des tâches de régression avec invariances additionnelles, dont les entrées sont des mesures de probabilité, en préservant sa robustesse et ses capacités d'approximation. Cette méthode permet de concevoir des résumés expressifs et adaptables de bases de données appelés « meta-features », dans le contexte de l'apprentissage automatisé. Dans une seconde partie, nous montrons que le recours à l'entropie facilite le calcul des quantiles conditionnels multivariés. Nous introduisons le problème de régression de quantile vectoriel régularisé, fournissons un algorithme efficace pour calculer les quantiles multivariés et montrons qu'il bénéficie de propriétés asymptotiques souhaitables.

MOTS CLÉS

Apprentissage statistique, Transport optimal, Réseaux neuronaux, Régression de quantile

ABSTRACT

This thesis proposes theoretical and numerical contributions to perform machine learning and statistics over the space of probability distributions. In a first part, we introduce a new class of neural network architectures to process probability measures in their Lagrangian form (obtained by sampling) as both inputs and outputs, which is characterized by robustness and universal approximation properties. We show that this framework can be adapted to perform regression on probability measure inputs, with customized invariance requirements, in a way that preserves its robustness and approximation capabilities. This method is proven to be of interest to design expressive, adaptable summaries of datasets referred to as “meta-features”, in the context of automated machine learning. In a second part, we demonstrate that the resort to entropy eases the computation of conditional multivariate quantiles. We introduce the regularized vector quantile regression problem, provide a scalable algorithm to compute multivariate quantiles and show that it benefits from desirable asymptotic properties.

KEYWORDS

Machine learning, Optimal transport, Neural networks, Quantile regression