Machine learning for structured data

Laetitia Chapel

Université Bretagne Sud
École doctorale MathSTIC, spécialité informatique

Habilitation à Diriger des Recherches

Apprentissage Machine pour Données Structurées

Laetitia Chapel

IRISA, UMR CNRS 6074

Soutenue le 22 mai 2022

Devant le jury composé de :

Rapporteurs

Gustau Camps-Valls	Professeur, Universitat de València
Stéphane Canu	Professeur, INSA Rouen
Marco Cuturi	Professeur, CREST-ENSAE

Examinateurs

Élisa Fromont	Professeur, Université de Rennes 1
Nicolas Papadakis	Directeur de recheche CNRS, Institut de Mathématiques de Bordeaux
Nicolas Courty	Professeur, Université Bretagne-Sud

Contents

Laetitia CHAPEL

Associate Professor, 40 years old
Université Bretagne Sud
Campus de Tohannic, 56000 Vannes

E-mail: laetitia.chapel@univ-ubs.fr
Web: https://people.irisa.fr/Laetitia.Chapel/

Experience

Since 2010	**Associate Professor**, *Université Bretagne Sud.* Research at *IRISA Lab, Obelix team* Teaching at *IUT de Vannes, Statistics and Business Intelligence department* "Prime d'encadrement doctoral et de recherche" (PEDR) rang A.
2008 - 2010	**Research fellow,** *National University of Ireland Maynooth.* Hamilton Institute.
2007 - 2008	**Research engineer**, *Cemagref now INRAE.* Laboratoire d'ingénierie des systèmes complexes.
2004 - 2007	**Ph.D. student**, *Cemagref now INRAE.* Laboratoire d'ingénierie des systèmes complexes.

Education

2004 - 2007	**PhD in Computer Science**, *Université de Clermont-Ferrand.* PhD advisor: **Guillaume Deffuant**. PhD title (in French): *Maintenir la viabilité ou la résilience des systèmes : les machines à vecteurs de support pour rompre la malédiction de la dimensionnalité ?* Committee: • *Reviewers*: **Rémi Munos** and **Patrick Saint-Pierre** • *Examiners*: **Philippe Mahey**, **Luc Doyen** and **Christian Mullon**
2003 - 2004	**Master's degree in knowledge discovery in databases** and **Master's degree in statistics,** *Université Lyon 2.*

Teaching and administrative activities

Teaching

Below is a selection of recent courses I have taught at IUT de Vannes, Statistics and BI department, and Université Bretagne Sud. From my appointment as an assistant professor at Université Bretagne Sud in 2010, my average teaching duty has been around 250 hours per year (heures équivalent TD). In 2019, I was awarded a "Congé Recherche and Conversion Thématique" (CRCT) of 6 months and in 2021, a CRCT of 4 months.

2020 - ...	**Introduction to Machine Learning for Earth Observation**, *13.5h.* Copernicus Master in Digital Earth - Master 2.
2014 - 2020	**Introduction to Machine Learning,** *18h.* IUT de Vannes, BSc.
2014 - ...	**Logistic regression**, *54h.* IUT de Vannes, BSc.
2019 - ...	**Statistical programming with Python**, *54h.* IUT de Vannes, BSc.
2014 - ...	**Supervised classification**, *39h.* IUT de Vannes, BSc.

Administrative responsibilities

Selection board

2021	**Assistant Professor in computer science**, *Irisa Lab*, IUT de Vannes.
2020	**Assistant Professor in computer science**, *Irisa Lab*, École St-Cyr, UBS.
2019	**Assistant Professor in computer science**, *Irisa Lab*, UBS.
2019	**Assistant Professor in computer science**, *Irisa Lab*, IUT de Vannes.
2018	**Assistant Professor in computer science**, *Bordeaux Computer Science Laboratory*, Bordeaux INP.

Administration

2020 - ...	**Member of "Commission de la recherche" and of "Conseil académique"**, *elected*, Université Bretagne Sud.
2018 - ...	**Member of "Conseil d'institu"**, *elected*, IUT de Vannes.
2015 - ...	**Work placement coordinator**, *Statistics and BI department*, IUT de Vannes.

Research activities

Student's supervision

Post-doctoral fellows

- **Pierre Gloagen**, *Anomaly detection in trajectory data*, co-supervised with Chloé Friguet & Romain Tavenard. 2017-2019. Publications: [J2], [C5].
  Now assistant professor at AgroParisTech.

Current PhD students

- **Guillaume Mahey**, *Optimal Transport for novelty and out-of-distribution detection*, with Gilles Gasso, 2021-....
- **Manal Hamzaoui**, *Structured classification of structured data*, with Minh-Tan Pham & Sébastien Lefèvre, 2019-.... Publications: [O1]
- **François Painblanc**, *Time series classification for remote sensing*, with Chloé Friguet & Romain Tavenard, 2019-....

Past PhD students

- **Titouan Vayer**, *A contribution to Optimal Transport on incomparable spaces*, with Romain Tavenard & Nicolas Courty, 2017-2020. Publications: [P1], [J3], [C3], [C4].
  Now post-doctoral fellow at ENS Lyon (Dante team).
- **Adeline Bailly**, *Time Series Classification Algorithms with Applications in Remote Sensing* , with Romain Tavenard, 2015-2018. Publications: [J4], [C6], [C9], [C10], [CH1].
  Now research engineer at Direction Générale de l'Armement.
- **Yanwei Cui**, *Kernel-based learning on hierarchical image representations: applications to remote sensing data classification*, with Sébastien Lefèvre, 2014-2017. Publications: [J5], [C7], [C8], [C11].
  Now ML specialist at Amazon.

Reviewing activities

- **Journal Reviewer** for Data Mining and Knowledge Discovery (DAMI), Machine Learning, International Journal of Remote Sensing, Remote Sensing (RS), IEEE Geoscience and Remote Sensing Letters (GRSL), IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS).
- **Conference Reviewer** for Neural Information Processing Systems (NeurIPS), International Conference in Machine Learning (ICML), International conference on Learning Representations (ICLR), International Conference on Artificial Intelligence and Statistics (AISTATS), International Geoscience and Remote Sensing Symposium (IGARSS).
- **Workshops and national conferences** Optimal Transport for Machine Learning workshop (OTML, NeurIPS 2019 and 2021), Conférence nationale d'Apprentissage automatique (CAp)

Involvement in academic projects

2019 - 2023 **MULTI-variate, -temporal, -resolution and -SourCe remote sensing image Analysis and LEarning (MULTISCALE)**, *Principal investigator.*
Agence Nationale de la recherche & Tubitak Agency (Turkey).

2020 - 2024 **A road toward safe artificial intelligence in mobility (RAIMO)**, *Collaborator.*
Agence Nationale de la recherche, IA chair. PI: Stéphane Canu.

2019 - 2023 **Machine learning tools for time series (MATS)**, *Collaborator.*
Agence Nationale de la recherche JCJC. PI: Romain Tavenard.

2017 - 2021 **Bringing Optimal Transport and Machine Learning Together (OATMIL)**, *Collaborator.* Agence Nationale de la recherche. PI: Nicolas Courty.

2017 - 2020 **Analysis of boat trajectory data (SESAME)**, *Collaborator.*
Agence Nationale de la recherche, Astrid Call. PI: Ronan Fablet.

2013 - 2016 **Spatio-Temporal Analysis by Recognition within Complex Images for Remote Sensing of Environment (ASTERIX)**, *Collaborator.*
Agence Nationale de la recherche, JCJC. PI: Sébastien Lefèvre.

Organization of scientific events

2022 **Multiscale - learning and analysis of multi* remote sensing images**, *Special session at ICPRAI 2022,* with Minh-Tan Pham, Sébastien Lefèvre and Erchan Aptoula.

2022 **Conférence nationale d'Apprentissage automatique (CAp)**, *Vannes,* with Elisa Froment, Nicolas Courty, Chloé Friguet, Charlotte Pelletier, among others.

2021 **Optimal Transport for Machine Learning Workshop** *at NeurIPS 2021,* with Jason Altschuler, Charlotte Bunne, Marco Cuturi, Rémi Flamary, Gabriel Peyré, Alex Suvorikova.

2016 **Statlearn Workshop**, *Vannes,* with Charles Bouveyron, Mathieu Emily, Pierre Latouche, Julien Jacques, Chloé Friguet, Nicolas Béchet, Nicolas Courty.

Committees

PhD committee member

2021 **François-Pierre Paty**, *Examiner.* Advisor: Marco Cuturi.

2021 **Kilian Fatras**, *Examiner.* Advisors: Nicolas Courty and Rémi Flamary.

2021 **Kimia Nadjahi**, *Examiner.* Advisors: Umut Simsekli, Alain Durmus and Roland Badeau.

ANR scientific selection committee

2021 **Member of "comité d'évaluation scientifique"**, *AI track.*

2021 **Member of "comité d'évaluation scientifique"**, *Call ANR - JST CREST AI (bilateral call with Japan).*

Publications

Pre-prints

[P1] T. Vayer, L. Chapel, N. Courty, R. Flamary, Y. Soullard, and R. Tavenard, *Time Series Alignment with Global Invariances*, arXiv preprint arXiv:2002.03848, 2020.

Journal articles

[J1] R. Flamary, N. Courty, A. Gramfort, M.Z. Alaya, A. Boisbunon, S. Chambon, L. Chapel, A. Corenflos, K. Fatras, N. Fournier, and others, *POT: Python Optimal Transport*, Journal of Machine Learning Research, 22(78), pp. 1–8, 2021.

[J2] P. Gloaguen, L. Chapel, C. Friguet and R. Tavenard, *Scalable clustering of segmented trajectories within a continuous time framework. Application to maritime traffic data*, Machine Learning, pp. 1–27, 2021.

[J3] T. Vayer, L. Chapel, R. Flamary, R. Tavenard, and N. Courty, *Fused Gromov-Wasserstein distance for structured objects*, Algorithms, 13(9), 2020.

[J4] A. Bailly, L. Chapel, R. Tavenard and G. Camps-Valls, *Nonlinear Time-Series Adaptation for Land Cover Classification*, IEEE Geoscience and Remote Sensing Letters 14(6), pp. 896–900, 2017.

[J5] Y. Cui, L. Chapel, and S. Lefèvre *Scalable Bag of Subpaths Kernel for Learning on Hierarchical Image Representations and Multi-Source Remote Sensing Data Classification*, Remote Sensing, 9(3), pp. 196, 2017.

[J6] L. Chapel, T. Burger, N. Courty and S. Lefèvre, *PerTurbo manifold learning algorithm for weakly labeled hyperspectral image classification*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(4), pp. 1070–1078, 2014.

[J7] L. Chapel, X. Castelló, C. Bernard, G. Deffuant, V.M. Eguíluz, S. Martin and M. San Miguel *Viability and resilience of languages in competition*, Plos one, 5(1), pp. e8681, 2010.

[J8] E. Sauquet, M.H. Ramos, L. Chapel, P. Bernardara, *Streamflow scaling properties: investigating characteristic scales from different statistical approaches*, Hydrological Processes: An International Journal, 22(7), pp. 3462–3475, 2008.

[J9] L. Chapel, G. Deffuant, S. Martin, and C. Mullon, *Defining yield policies in a viability approach*, Ecological Modelling, 212(1-2), pp. 10–15, 2008.

[J10] G. Deffuant, L. Chapel, and S. Martin, *Approximating viability kernels with support vector machines*, IEEE transactions on automatic control, 52(5), pp. 933–937, 2007.

International conferences or workshops with proceedings

[C1] L. Chapel*, R. Flamary*, H. Wu, C. Févotte, and G. Gasso, *Unbalanced Optimal Transport through Non-negative Penalized Linear Regression*, Advances in Neural Information Processing Systems (NeurIPS), 2021. (* indicates equal contribution)

[C2] L. Chapel, M.Z. Alaya, and G. Gasso, *Partial Optimal Tranport with applications on Positive-Unlabeled Learning*, Advances in Neural Information Processing Systems (NeurIPS), 33, 2020.

[C3] T. Vayer, R. Flamary, R. Tavenard, L. Chapel, N. Courty, *Sliced Gromov-Wasserstein*, Advances in Neural Information Processing Systems (NeurIPS), 2019.

[C4] T. Vayer, L. Chapel, R. Flamary, R. Tavenard and N. Courty, *Optimal Transport for structured data with application on graphs*, International Conference on Machine Learning (ICML), 2019.

[C5] R. Fablet, N. Bellec, L. Chapel, C. Friguet, R. Garello, P. Gloaguen, G. Hajduch, S. Lefèvre, Sébastien, F. Merciol and P. Morillon, *Next Step for Big Data Infrastructure and Analytics for the Surveillance of the Maritime Traffic from AIS & Sentinel Satellite Data Streams*, BiDS'2017-Conference on Big Data from Space, 2017.

[C6] R. Tavenard, S. Malinowski, L. Chapel, A. Bailly, H. Sanchez, and B. Bustos, *Efficient Temporal Kernels between Feature Sets for Time Series Classification*, Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD), 2017.

[C7] Y. Cui, L. Chapel, and S. Lefèvre, *Combining multiscale features for classification of hyperspectral images: A sequence-based kernel approach*, IEEE Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2016.

[C8] Y. Cui, S. Lefèvre, L. Chapel, and A. Puissant, *Combining multiple resolutions into hierarchical representations for kernel-based image classification*, International Conference on Geographic Object-Based Image Analysis (GEOBIA), 2016.

[C9] A. Bailly, D. Arvor, L. Chapel and R. Tavenard *Classification of MODIS time series with Dense Bag-of-Temporal-SIFT-Words: Application to cropland mapping in the Brazilian Amazon*, IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2016 (13 citations).

[C10] A. Bailly, S. Malinowski, R. Tavenard, T. Guyet and L. Chapel, *Bag-of-Temporal-SIFT-Words for Time Series Classification*, ECML/PKDD International Workshop on Advanced Analytics and Learning on Temporal Data, 2015.

[C11] Y. Cui, L. Chapel and S. Lefèvre, *A subpath kernel for learning hierarchical image representations*, International Workshop on Graph-Based Representations in Pattern Recognition (GBR), 2015.

[C12] L. Chapel, and C. Friguet, *Anomaly detection with score functions based on the reconstruction error of the kernel PCA*, Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD), 2014.

[C13] S. Lefèvre, L. Chapel, F. Merciol, *Hyperspectral image classification from multiscale description with constrained connectivity and metric learning*, IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2014.

[C14] F. Merciol, L. Chapel, and S. Lefèvre, *Hyperspectral image representation through alpha-trees*, 9th Conference on Image Information Mining, 2014.

[C15] L. Chapel, T. Burger, N. Courty and S. Lefèvre, *Hyperspectral image representation through alpha-trees*, IEEE International conference on Geoscience and Remote Sensing Symposium (IGARSS), 2012.

[C16] L. Chapel, and G. Deffuant, *Inner and Outer Capture Basin Approximation with Support Vector Machines*, 8th International Conference on Informatics in Control, Automation and Robotics, 2011.

[C17] J. Keeney, O. Conlan, V. Holub, M. Wang, L. Chapel, M. Serrano, and S. Van Der Meer *A semantic monitoring and management framework for end-to-end services*, IEEE International Symposium on Integrated Network Management, 2011.

[C18] L. Chapel, and D.J. Leith, *Sparse input matrix and state estimation for linear systems*, IEEE Conference on Decision and Control (CDC), 2010.

[C19] L. Chapel, D. Botvich, and D. Malone, *Probabilistic approaches to cheating detection in online games*, IEEE Conference on Computational Intelligence and Games, 2010.

[C20] L. Chapel, and G. Deffuant, *SVM viability controller active learning: Application to bike control*, IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, 2007.

[C21] L. Chapel, S. Martin, and G. Deffuant, *Lake eutrophication: using resilience evaluation to compute sustainable policies*, International Conference on environmental systems and technology, 2007.

[C22] L. Chapel and G. Deffuant, *SVM viability controller active learning*, Kernel machines for reinforcement learning workshop, 2006.

[C23] L. Chapel, G. Deffuant, S. Martin, and C. Mullon, *Defining yield policies in a viability theory approach*, European Conference on Ecological Modeling, 2005.

Book chapters

[CH1] A. Bailly, S. Malinowski, R. Tavenard, L. Chapel and T. Guyet, *Dense Bag-of-Temporal-SIFT-Words for Time Series Classification*, LNAI special volume of Advanced Analytics and Learning on Temporal Data (AALTD), 9785, pp. 17–30, 2016.

[CH2] L. Chapel and G. Deffuant, *Approximating Viability Kernels and Resilience Values: Algorithms and Practical Issues Illustrated with KAVIAR Software*, Viability and Resilience of Complex Systems, Springer, pp. 161–192, 2011.

[CH3] X. Castelló, F. Vazquez, V.M. Eguíluz, L. Loureiro-Porto, M. San Miguel, L. Chapel and G. Deffuant, *Viability and resilience in the dynamics of language competition*, Viability and Resilience of Complex Systems, Springer, pp. 39–73, 2011.

[CH4] L. Chapel and G. Deffuant, *SVM Approximation of Value Function Contours in Target Hitting Problems*, Lecture Notes in Electrical Engineering, pp. 37–48, 2013.

Others

[O1] M. Hamzaoui, L. Chapel, M.-T. Pham, S. Lefèvre, *Hyperbolic Variational Auto-Encoder for Remote Sensing Scene Classification*, ORASIS, journées francophones des jeunes chercheurs en vision par ordinateur, 2021.

[O2] L. Chapel, *Maintenir la viabilité ou la résilience d'un système : les machines à vecteurs de support pour rompre la malédiction de la dimensionnalité ?*, PhD thesis, Université Blaise Pascal, Clermont-Ferrand, 2007.

Introduction

Context of my research

This document is a summary of my research conducted within Obelix team[1] of IRISA lab since 2012 as an assistant professor. My research has mainly focused on the development of machine learning tools for structured data, originally applied to Remote Sensing (RS) data, and more recently directed toward methodological work in Optimal Transport (OT) for Machine Learning (ML).

My PhD thesis [O2] took place at IRSTEA in Clermont-Ferrand and dealt with the problem of defining control policies to maintain the viability or the resilience of a dynamical system. In opposite to optimal control that seeks the dynamical system to reach an optimal value, viability theory deals with the problem of finding action policies that keeps indefinitely the dynamical system within an acceptable range of constraints. Note that all the initial states may not be able to reach this objective; the set of initial states for which such a set of control policies exists is called the *viability kernel*. In this thesis, I proposed to use Support Vector Machines to construct the viability kernel, which provides a kind of barrier function useful to use optimization techniques to find the appropriate action policies, avoiding the computationally intensive discretization of the control variables, allowing one to alleviate the dimensionality curse related to the dimension of the control variables.

I then joined the National University of Ireland Maynooth as a research fellow. My main work there was still focused on defining new solutions for dynamical systems relying on ML tools. In particular, I addressed the problem of sparse identification of the input matrix parameter in linear systems. Taking advantage of the connections between Kalman Filters and least square estimation, I formulated a filter that combines state and sparse input matrix estimation as a ℓ_1 regularized least square optimization problem [C18].

I was recruited at Université Bretagne Sud as an assistant professor at the end of 2010. Making the analysis that the research themes dealt with my original research team did not enthral me, I took place in 2012 in the creation of a new Irisa team *Obelix*, whose research themes are centered on the understanding of environmental systems through observations. Here again, ML is a key to learn from (possibly series of) large amount of structured data available from, among other, RS images. I saw, in the problematics tackled by the team, a lot of common features with my previous works, but also new challenges mainly related to the objects of interest. Indeed, these are highly structured data in the sense that we often deal with time series of geographically structured pixels that can be represented as trees or graphs. My involvement on the ANR project Asterix (2013-2016), standing for "Spatio-Temporal Analysis by Recognition within Complex Images for Remote Sensing of Environment", was a great opportunity to focus on these new research themes.

Remote sensing data processing has long been conducted at pixel level but the significant developments on the spatial resolution front have led to the emergence of object-based image analysis, that no

[1] Obelix stands for *environment observation with complex imagery*.

longer handles every pixel independently, but rather in contextual groups, thus increasing significantly the involved information extraction capacity. In particular, multiscale models such as hierarchical representations (or trees) have been proposed and widely acknowledged as the appropriate solution since they enable modeling efficiently the relations between different image objects at multiple detail levels. Then, it seemed to me that it was an important and challenging problem to be able to tackle those hierarchical representations efficiently in order to perform RS images classification to, e.g., produce land cover land use maps. Defining new kernel-based learning methods able to deal with data structured as trees was the theme of the PhD of Yanwei Cui [Cui 2017a], that I co-supervised with Sébastien Lefèvre, who started in 2014. He defined new kernels able to deal with hierarchical data described by continuous attributes and showcased their interest in the RS data classification context. More recently, hyperbolic spaces attracted my attention. They are geometrical spaces that inherently encode very efficiently tree-structured data. This motivated the launch of the PhD thesis of Manal Hamzaoui (started in 2019, co-supervised with Sébastien Lefèvre and Minh-Tan Pham) whose aim is to take benefit of the hyperbolic geometry in the context of RS image classification.

At the same time, and driven by my past interests with dynamical systems, I started to get interested in processing observed time series of data. Indeed, the launch of numerous space programs as well as Earth Observation satellites, such as Copernicus, has led to an abundance of series of satellite images with a given temporal resolution (Sentinel-2 possesses a revisit cycle between 5 to 10 days). Consequently, processing paradigms become mandatory in order to manage this mass of timely structured data. This motivated the PhD of Adeline Bailly [Bailly 2018] that I co-supervised with Romain Tavenard, and the post-doc of Pierre Gloaguen (2017-2019, co-supervised with Romain Tavenard and Chloé Friguet) within the context of the ANR Sesame (2017-2020). Dealing with time series is a challenging problem as one should take into account the temporal correlation that exists between the observations, the possibly high dimensionality and the noise that is often present in RS series of images. Hundreds of dedicated machine learning tools for dealing with this specific object exist, whose interest depends on the view over what phenomenon drives the data and how the temporal behavior should be (or not) included in the data. Adeline focused on defining accurate representations of the data, allowing better classification performances in a wide range of scenarii. Pierre's targeted the problem of clustering a huge amount of trajectories using continuous time models. We also studied how incorporating a temporal localization information efficiently. Finally, I supervise the PhD of François Painblanc (started in 2019, co-supervised with Romain Tavenard and Chloé Friguet) whose aim is to develop dedicated machine learning tools for domain adaptation dedicated to time series (*e.g.* when series of data are labelled in one year, and that a classification should be performed the following year, with a difference in the evolution of the phenomenon that is under study). These works are funded by the ANR MATS, standing for "Machine Learning for Time Series".

Remote sensing images can also be represented as graphs, representing the relations that may exists between the features and/or the localization of superpixels/regions. It drove me to the study of graphs as objects of interest, and to study the problem of how classifying them efficiently. This was the topic of the PhD of Titouan Vayer [Vayer 2020a], co-supervised with Nicolas Courty and Romain Tavenard, that took place within the ANR OATMIL ("Bringing Optimal Transport and Machine Learning Together") project. In this context, we built new metrics to compare graphs (among other structured data), relying on tools from optimal transport. We introduced the Fused Gromov-Wasserstein distance which has been shown to be effective in comparing benchmarked graphs, studied some theoretical properties of this distance, and also proposed the Sliced Gromov-Wasserstein distance, allowing Gromov-Wasserstein to scale well with the number of points.

Strong from this experience in dealing RS structured data, I launched in 2019 an international ANR project entitled MULTISCALE (MULTI-variate, -temporal, -resolution and -SourCe remote sensing image Analysis and LEarning) together with researchers in Turkey (PI: myself from the French side, Erchan Aptoula from the Turkish side). This research project aims at providing a complete and integrated framework for multiscale image analysis and learning with hierarchical representations of series of complex remote sensing images.

Almost independently from the RS integration of my research, I got interested in one major drawback (to my viewpoint) of optimal transport for machine learning: *all* the information has to be transferred from one source domain to a target one. This hypothesis does not make sense in the machine learning context as the observations are often polluted by noise, can contain outliers or may suffer mislabeling issues. It gave rise to my interest for *partial* or *unbalanced* optimal transport, that I tackled from a methodological point of view, with the aim to develop new solutions to solve the problem. These works are the starting point of the thesis of Guillaume Mahey (co-supervised with Gilles Gasso), who started in November this year, whose aim is to define optimal transport tools for out-of-distribution detection, within the context of the ANR RAIMO ("Vers une intelligence artificielle sûre pour la mobilité").

Learning from structured data

What are structured data? In many domains, the data can be decomposed into a set of *entities* that possess an intricate internal structure usually in the form of one or more *relations* between them [Battaglia 2018]. In comparison to usual "flat" data represented in a sample×features table (or tabular format), structured data not only involve entities and their description, but also describe the connectivity between them. Typical instances of structured data are images, sequences, trees or graphs and are ubiquitous in many learning tasks. For example, machine translation consists in automatically converting a sequence of symbols in some language into a sequence of symbols in another language. Time series forecasting aims at predicting future events through a sequence of timely dependent observations. Link prediction in networks has ubiquitous applications in biological, social or transportation networks. Multiscale image segmentation consists in dividing an image into homogeneous regions that are included one into an other depending on the scale we consider.

Graphs are often used as models of systems of entities and relations. A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a collection of nodes $\mathcal{V}$ describing the entities (that can be endowed with features) and of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ (directed or undirected) that represent the relations. Trees are connected acyclic undirected graphs. When the order matters, sequence and times series are instances of directed graphs with fixed ordering (see figure 1). Graphs give rise to a property that may be satisfied: the permutation invariance property or *isomorphism*. Two graphs are isomorphic when there exists an edge-preserving bijection between them, that is to say when they are identical up to a reordering of their nodes. This is a key property when dealing with structured data as the entities, except time serie or sequences, are not provided in any particular order.

Main learning streams for structured data. Learning from structured data presents specific challenges as i) the commonly used feature-attribute representation may not hold and different instances may have different lengths ii) the i.i.d. assumption may no longer be applicable iii) the relational property should be taken into account. One consequence is that the usual techniques for classifying numerical data are not directly applicable and it calls for the need of a rethinking of the main questions of machine learning (see figure 2): i) how to accurately represent the data such that the resulting embedding describes

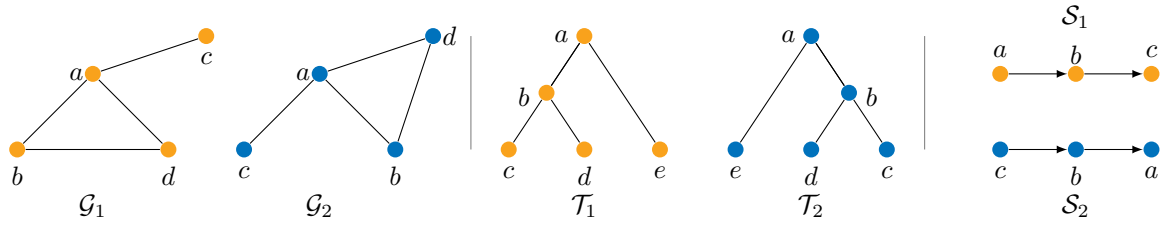


Figure 1: The two labeled graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are isomorphic as there exists a vertex bijection which is both edge-preserving and label-preserving. The two trees $\mathcal{T}_1$ and $\mathcal{T}_2$ are connected acyclic undirected graph that are isomorphic. Time series are a special instance of directed graphs that has a fixed neighborhood structure with a fixed order. Time series $\mathcal{S}_1$ and $\mathcal{S}_2$ are not identical here as the ordering of the nodes conveys temporal information that should be taken into account.

efficiently both the feature and the structure information? ii) how to define a appropriate measure of distance between two structured data? One key question is also how to deal with noise and/or mislabeled samples.

One main stream for dealing with structured data is to look for an "appropriate" embedding in order to come down to a propositional (i.e. based on feature-vector representations) learning algorithm. The problem is then to construct a new set of features that captures the relational properties of the data. Such a transformation is sometimes referred to as propositionalization [Kramer 2001]. This is the aim of the bag-of-word technique [Sivic 2008] which extracts features from text documents and structured data in general; efficient solutions when dealing with sequences of words are now Word2vec [Mikolov 2013] or Glove [Pennington 2014] and take as input a large corpus of text and produces a vector space, in which words that share common contexts in the corpus are located close to one another in the space. Graph embedding aims at representing graphs as low dimensional vectors such that the graph structures are preserved [Cai 2018]. Struc2vec [Ribeiro 2017] learns latent representations of the nodes of the graph that encodes structural similarities and context. Rather than considering Euclidean spaces that may be unfit for representing structured data, some other lines of research embed the data into spherical [Davidson 2018], hyperbolic [Nickel 2017] or even a combination of those spaces [Gu 2018], allowing a better latent representation in lower dimensional spaces. When it comes to time series, dictionary-based classifiers describe the series as representative words: shapelets [Lines 2012] are time series subsequences which are representative of a class; SAX [Lin 2007] is a symbolic representation of time series; dynamic topic models [Wang 2008] are generative models that describe the evolution of topics (i.e. a distribution over features) of a collection of documents (i.e. bags of extracted features) over time. Deep learning has been widely employed to address specifically the problem of time series forecasting or classification, the most popular models probably being recurrent neural networks RNN and long short-term memory LSTM. RNN model the temporal dependency by connecting each time step with the previous one; LSTM [Hochreiter 1997] can be seen as an instance of RNN that learns the temporal aspect in larger horizon by keeping track of short-term patterns; more recently, another type of recurrent unit, gated recurrent unit (GRU) [Cho 2014], has been shown to perform well on tasks with long term dependencies in this context.

There has been a long history of designing dedicated distance or similarity measures between structured data as many machine learning algorithms require assessing how (dis)similar two objects are. Here we name a very few of them in a structured data context. Dedicated distances exist for sequences, e.g. the Levenshtein distance [Levenshtein 1966] which is defined as the number of edit operations that one needs

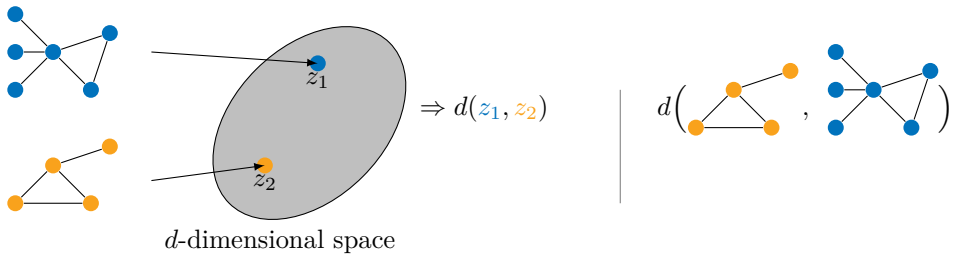


Figure 2: (Left) The two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are embedded in a d-dimensional space in which they can be easily compared. (Right) The two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are compared thanks to a dedicated function that operates directly on the structured data.

to perform on one sequence in order to obtain the second; dynamic time wrapping (DTW) [Sakoe 1978] deals with time series. When it comes to probability distributions, the Kullback–Leibler divergence [Kullback 1951] or the Earth Mover distance [Rubner 2000] are state-of-the-art metrics. The graph isomorphism problem [Babai 2018] deals with the problem of determining whether two finite graphs are isomorphic, i.e. looks if there exists a bijection between those graphs; graph [Vishwanathan 2010] (resp. tree) kernels may be instances of convolution kernels that combines distances computed on substructures [Vishwanathan 2004]. The global alignment kernel [Cuturi 2011] is a weighted combination of the set of all possible alignments between time series.

The case of incomparable spaces. In most of the previous approaches, the main challenge is to compare data that live in *incomparable spaces*, that is to say data that have their own information that may be not shared with the others. Propositional data are usually structured as common feature vector representation that is handy to cope with. On the opposite, relational or structured data may not share the same features but also carry extra information about the structure that has to be incorporated into the learning process. When dealing with a dataset composed of several graphs or trees, the latter may not have the same number of nodes neither the same set of attributes or even edges. This also encompasses the case when the data are collected under distinct environments, representing different times of collection, contexts or measurement modalities (e.g. when different sensors are used to measure related quantities). Dedicated paradigms such as heterogeneous domain adaptation [Yeh 2014], directly take into account this problem in the learning process. This is a situation that is very often encountered in remote sensing for example, when the same area of interest has be caught at different times with different captors [Voreiter 2020]. One other approach is to learn, based on some prior knowledge w.r.t. some invariance, classes in order to be more robust to irrelevant feature transformations [Battaglia 2018].

Contributions and articulation of this document

Over the last 10 years in the Obelix team, I have focused on designing ML methods and algorithms that deal with structured data. In this document, I have sum up some of my contributions; the “we” or “our” formulations reflecting that they have always be performed in collaboration with some colleagues. The manuscript is organized in three parts, mostly chronologically:

Contributions to machine learning for remote sensing images. The first part of this document sums up our oldest works (**2014-2017**) that deal with remote sensing images as spatially structured objects of interest. Indeed, the significant advances on the spatial resolution front have led to the emergence of object-based analysis, in which images are described as contextual groups, that may be organized hierarchically, depending at the scale we are dealing with. I sum up some of our contributions in designing dedicated tree-kernel based methods in this context. I also describe some works that studied how a manifold learning algorithm is interesting to perform hyperspectral image classification. Albeit older, these works are the base of more prospective current works that are described in the last chapter of the manuscript.

Contributions to machine learning for time series. The second part deals with our works (**2015-2021**) related specifically with time series as objects of interest. It starts with a brief description of some particular challenges brought by time series in the ML community. The description of our works is divided into two chapters: the first one relates with solutions that rely on efficient and effective time series embeddings, such as bags-of-words or continuous time descriptions of the time series, while the second one focus on contributions in designing dedicated similarity measures between time series, among which a dedicated temporal kernel and a DTW-based metric that rely on aligned time series.

The aforementionned works mostly rely on "classical" and well-established machine learning tools such as kernels or bags-of-words. At some point, the question that arised was if there were not some more powerful tools for dealing with structured data. It then led me to consider optimal transport that has been recently successfully introduced in the machine learning community in a wide range of learning applications (see the discusion about optimal transport and machine learning in section 6.4), but at this time, mostly on vectorial data.

Contributions to Optimal Transport for Machine Learning, with a focus on graphs. The last chapter is then dedicated to our more recent works (**from 2019 to 2021**) that aim at exploiting the geometrical properties of optimal transport (OT) to design a new metric to compare graphs, namely the fused Gromov-Wasserstein distance (FGW). Optimal transport has been quite recently introduced in the machine learning community and is now an integral component of many core ML methods. This has brought new challenges to the OT community, with thus a need to design new theories and efficient frameworks to make OT and ML more compliant.This drove us to also propose contributions in designing OT solutions that allow i) dealing with outliers or noise ii) designing solutions that are more large scale compliant. This part starts with an introduction of the main ingredients of OT.

Global summary of my research and perspectives. Finally, this document ends with some still open challenges and some research perspectives.

Part I

Contributions to Machine Learning for Remote Sensing Images

Chapter 1

Machine Learning for Object-Based Image Analysis in Remote Sensing

Contents

The aim of this chapter is to set the scene of machine learning for complex remote sensing data. We start by giving the main features and challenges related to complex and structured remote sensing data, then we focus on the special case in which the data are represented as hierarchical representations. GEOgraphic-Object-Based Image Analysis framework [Blaschke 2014] has gained increasing interest and is today a paradigm for remote sensing image processing beyond conventional pixel-based analysis. In this framework, hierarchical image representation is the key concept –meaningful objects are obtained from multiscale image segmentations– , and spatial relationships among objects are encoded in a tree structure.

1.1 Remote Sensing data are complex and inherently structured

1.1.1 The era of complex remote sensing big data

Besides the significant developments in terms of spatial and spectral image resolutions, the past decade has also witnessed an impressive increase in the availability and rate of acquisition of remote sensing images. While their cost of acquisition was one the main reasons behind their scarcity in the past, the launch of the Copernicus program has led nowadays to the provision of free data. In particular, remote sensing data are now characterized by its volume (multiple petabytes per year) and variety: we can now easily access and obtain multiple (LiDAR, SAR and optical) images at various spatial (e.g. SPOT4 at 10m, SPOT5 at 10m, Pleiades at 2m), spectral (multi- and even hyperspectral images) and temporal (Sentinel-2 possesses a revisit cycle between 5 to 10 days) resolutions for a given geographical region of study. In parallel to this development, there is a significant rise in the number of both public and

private satellites (Pléiades, WorldView 2-3 etc.) capable of providing very high spatial resolution images (<2m) as well as of aerial missions able to enrich these data through LiDAR and hyperspectral images. Moreover, the description of a given region of interest can be further reinforced through geographic databases or even expert knowledge. Consequently, original processing paradigms become mandatory in order to manage this mass of heterogeneous data, exploit their inherent complementarity and accomplish various tasks such as land-use monitoring and natural disaster management.

1.1.2 Some challenges related to structured and complex remote sensing data

The general challenge of machine learning for Earth Observation is to transform this impressing amount of data into knowledge. Due to the inherent structure of the data, several challenges have then emerged in the past years.

High resolution: correlation within the spectral and within the spatial domain. High resolution comes from both spectral and spatial domains. In the spectral domain, the hyperspectral image sensors allow the acquisition of the signal in hundreds of spectral wavelengths for each image pixel, with correlated dimensions. It induces problems related to the high dimensionality of data, especially in the case of limited availability of training samples. It is then customary to reduce the dimensionality into a representation that still account for the structure of the data. (Non-)Linear dimensionality reduction such as PCA or LDA methods have been found to be very efficient and effective methods in that context. Manifold learning algorithms assume that the original high dimensional data actually lie on an embedded lower dimensional manifold. The mapping of the data from high to low dimensional spaces can also be learnt thanks to learning algorithms such as ISOMAP [Tenenbaum 2000] or local linear embedding [Roweis 2000], whose aim is to provide a low-dimensional space that preserves the local neighboring relationships of the high-dimensional data. For a long time, kernel methods have been state-of-the-art methods for hyperspectral image analysis [Camps-Valls 2006] but more recently, deep models have demonstrated their potential to extract discriminative features between different classes in the remote sensing context [Li 2019, Yuan 2015]. Learning the manifold structure or extracting appropriate feature for remote sensing images are still under active study by the community, such as the very recent works of [Taskin 2021] or [Baisantry 2021].
Another challenge is related to high spatial resolution. The availability of Very High Spatial Resolution (VHSR) images provides submetric resolution, allowing the exploitation of the fine details of the observed scene, where additional information such as texture, shape of complex objects or even structure of the object composition can be better revealed. We thus observe a high spatial autocorrelation of remote sensing imagery, with every pixel that are more likely to "look like" their neighboring pixels (see figure 1.1). Beyond the large volume of data that has to be tackled, the spatial and the autocorrelation structure has then to be taken into account. Contextual local visual features such as SIFT have then been defined to compute a holistic image representation [Yang 2008]. One other option is to rely on homogeneous superpixels (see section 1.2) that allow adaptive partition of remote sensing imagery into meaningful objects with coherent spatial properties [Lv 2019]. Dedicated network architectures are also used in this context, e.g. capsule networks that uses groups of neurons that can encode spatial information of features [Zhang 2019].

Multi-modality. Remote sensing data often come from several sensors, possibly with different spatial and spectral resolutions, different acquisition conditions, different views, different nature or even addi-

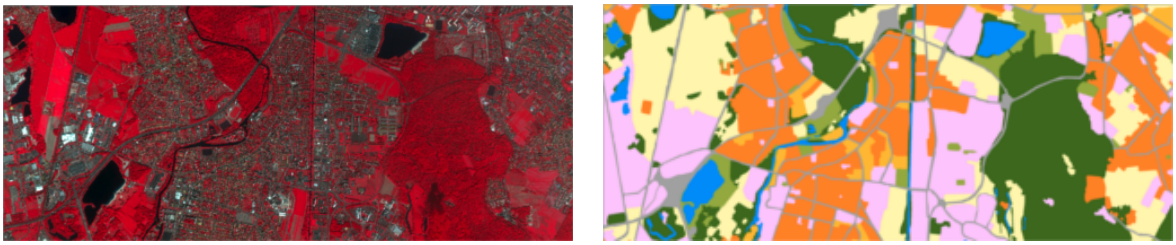


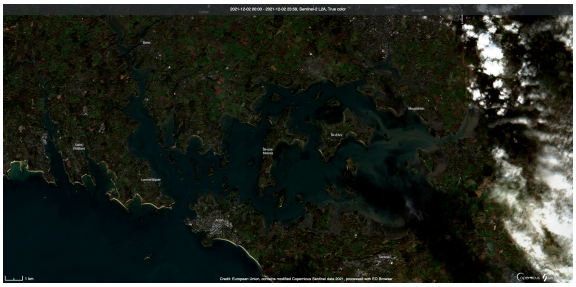
Figure 1.1: Urban scene taken over South of Strasbourg, France. (Left) false color image of Pleiades (© CNES 2012, distribution Airbus DS / Spot Image) with 50 cm resolution, and the associated ground truth (© LIVE UMR 7362) whose labels form large homogeneous areas within the image[2].

tional expert knowledge (see figure 1.2). It then provides a richer description of the same scene but the challenge is then to be able to exploit the complementary information carried by different modalities. The data fusion contest is held annually by IEEE Geoscience and Remote Sensing Society in order to encourage the development of new methods. The fusion can occur in different processing levels, that is to say at the raw data level, the feature level and the decision level [Schmitt 2016]. When one aim at transferring knowledge from one (or several) images to one another, we face a *heterogeneous* domain adaptation problem. It is one of the hottest scenario in the community; most of the existing solutions are devoted to learning a feature mapping function to map both the source and the target into a common feature space using kernels [Tuia 2016], deep methods [Voreiter 2020] or try to transfer knowledge across deep architectures [Pires de Lima 2020].

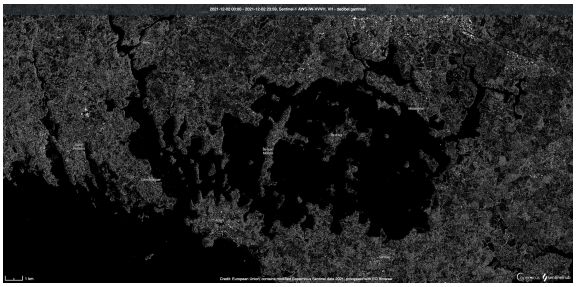
Large volume and scarcity of labels. The dynamic of remote sensing programs leads to abundance of geospatial image data: as an example, the Sentinel Copernicus program aims at delivering around 4 TeraBytes of data each day in the next few years. Machine learning algorithms have then to accommodate with the large scale nature of the data. While this amount of data is amenable to deep learning techniques, the scarcity of labels for (newly collected) data makes the problem trickier to solve. This lack of labeled is at the origin of the growing interest for semi-supervised learning in the remote sensing community as it provides a powerful framework for leveraging on unlabeled data [Oliver 2018]. For example, semi-supervised learning with large convolutional networks is able to leverage large collections of unlabeled images (up to 1 billion) [Yalniz 2019]. Out of the scope of this manuscript, scalability issues related with memory and computational requirements can be addressed through the development of dedicated computing structures, high performance computing and decentralized algorithms. More related to the scarcity of labels for newly collected data, open-set tasks including few-shot learning, whose aim is to *adapt* the classifier to unseen categories based on few labeled samples, have been recently considered in the remote sensing community [Zhang 2021].

Multi-temporality. Change detection refers to the task of analyzing two (or more) images of the same scene with the aim to flag changes between the acquisitions. This is an active research area with a broad range of applications, and many methods have been developped (see [Si Salah 2020] for a categorization of them). The difficulty in that context mainly depends on the acquisition conditions of the images, with more challenging scenario when they come from different sensors and are taken with different viewpoints. When it comes with a series of images (figure 1.3), one approach proceeds by analyzing the evolution of

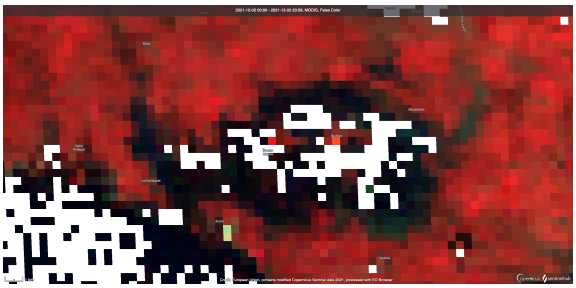
[2]The data have been provided by Anne Puissant, LIVE UMR 7362, Strasbourg.



(a) Sentinel02 (13 bands - colored in RGB) with a resolution from 10 meters to 60m.



(b) Sentinal01 (SAR image) with a resolution of 10m or 40m.



(c) MODIS (36 bands - false color) with a resolution of 250m to 1000m.

(d) LANDSAT 8 (11 bands) with a resolution of 15m to 30m.

Figure 1.2: Images of the Gulf of Morbihan taken in November 2021 by different sensors at different resolutions. Images have been collected on `https://apps.sentinel-hub.com/eo-browser`.

pixels through time [Bagnall 2017]. In the case of satellite image time series classification, the TempCNN architecture [Pelletier 2019] uses convolutions applied in the temporal domain. In section 4.1, we propose a continuous time algorithm for dealing with GPS trajectories that are composed of shared sub-trajectories. Classical CNNs, such as ResNet, have also been adapted for time series [Ismail Fawaz 2019]. One of the main challenge in this context is when the time distortion that may be unknown or different among classes (see the related discussion in section 3.3) and when the images come from different domains [Bailly 2017] (not described in this manuscript).

1.2 Object-Based Image Analysis and hierarchical image representation in Remote Sensing

1.2.1 Remote sensing hierarchical representations

Remote sensing data processing has long been conducted at pixel level, since times when objects of interest where way smaller or at most comparable to a pixel's size. The significant developments on the spatial resolution front has led to the emergence of object-based image analysis, that no longer handles every pixel independently, but in contextual groups, thus increasing significantly the involved information extraction capacity. Since then, it has become possible to analyze image content not only based on its spectral values, but also on their morphological (e.g. size, shape) and contextual (e.g. adjacency relations with other objects, neighborhood relations) properties as well. In particular, multiscale models such as hierarchical representations (or trees) have been proposed and widely acknowledged as the appropriate

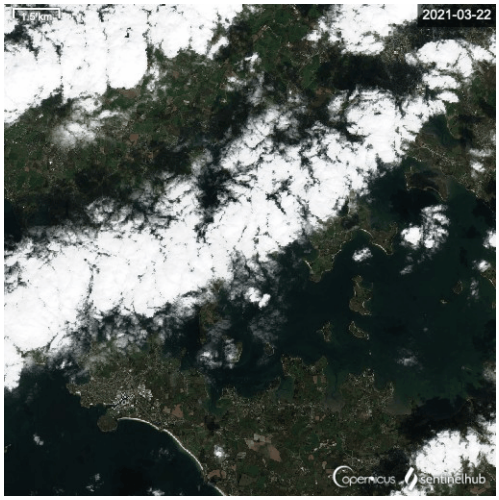


Figure 1.3: Time series of images taken in 2021 of the Gulf of Morhbihan, the city of Vannes being at the top center of the image. The images have been taken from Sentinel02 satellite. They are multispectral images (13 bands) and are colored only with RGB bands. Images with cloud cover more than 10% have been discarded. The animation can only be seen some pdf viewers such as Adobe Acrobat reader. Images have been collected on https://apps.sentinel-hub.com/eo-browser.

solution since they enable modeling efficiently the relations between different image objects at multiple detail levels, hence the structure present within the image; just for the sake of reference, suffice it to say that it is currently possible to process tens of millions of pixels within a single spectral band in just a single second.

Besides their computational and data modeling advantages, hierarchical representations have additionally paved the way to novel content description approaches. Specifically, attribute filters are capable of processing entire connected components, and hence enable object-based image analysis. Despite being long known, it has only been thanks to the advent of efficient hierarchical representations that they became available as powerful pixel and content descriptors in the form of both attribute profiles and attribute spectra. One other advantage of these representations is that they are most suitable for handling objects of interest that can manifest themselves at multiple scales. In fact, depending on the required level of analysis, it is no longer uncommon for image regions to require multiple labels; e.g. a region might be identified as a road at a fine scale, as a residential area at an intermediate scale, or even as town or city at a coarse scale. These labels might be known a priori (supervised classification) or not (unsupervised classification or segmentation).

Providing a complete and integrated framework for multiscale image analysis and learning with hierarchical representations of complex remote sensing images is the aim of the ANR research projet Multiscale[3] (see figure 1.4).

1.2.2 Some challenges related to learning (with) hierarchical image representations

Learning hierarchical representations. Tree-based approaches are widely used and acclaimed methods for remote sensing image representations. Yet, so far their application has been mostly limited to panchromatic optical images, and multivariate images where they are mostly calculated on a per-band

[3] https://people.irisa.fr/Laetitia.Chapel/multiscale/

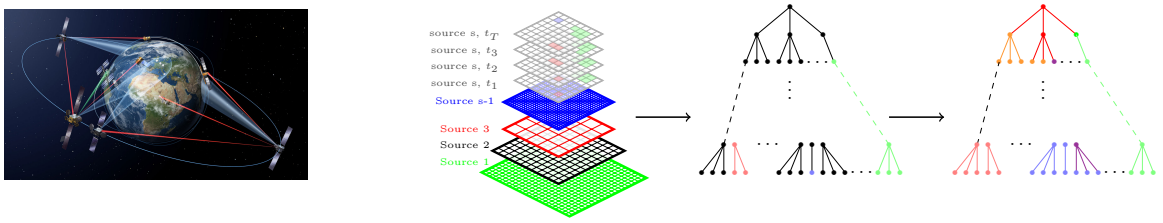


Figure 1.4: Hierarchical representations constructed from multivariate, multi-source, multi-resolution and multi-temporal data allow learning at multiple scales. (Left) Sentinel online - ESA; (Middle Left) set of images coming from possibly different sources, at different resolutions and different time instants; (Middle right) hierarchical representation built or learnt from this set of data; (Right) learning on the hierarchical representation. Colors represent here classes or optimal cuts at different and adaptive levels.

basis with just a handful of exceptions; thus without taking into account inter-band information. When it comes to multi-modal data, tree construction from a temporal set of images has little been considered, expect in the specific case of SAR images [Alonso González 2014] or in the work of [Tuna 2020] who adapted tree-based representations to satellite image time series. In addition, even if there exists numerous works about dealing with multi-source data (e.g. [Dechesne 2017]), there exists almost no hierarchical representation that allows taking into account multi-source data. One can nevertheless cite [Tochon 2015] who extended hierarchical representations to multi-modal data, by separating however the spectral, spatial and temporal dimensions. On the top of that, data often come with additional information as expert knowledge, geographical databases or labels, and this information is rarely introduced in the construction phase of the tree. In [Lefèvre 2014], we aim at integrating some labels within the construction phase, by encouraging regions with same labels to be aggregated together thanks to metric learning. In a different context than remote sensing, there has been some recent works dedicated to learning hierarchical representations [Nickel 2017], mostly on symbolic data, in some well-suited spaces (i.e. hyperbolic spaces) for data that exhibit latent hierarchical structure. Adapting those concepts to (multi-modal and multi-temporal) hierarchical representations is, to my opinion, one of the most promising challenge in the remote sensing community.

Learning on hierarchical representations. Hierarchical representations constitute rarely an end by themselves. Instead, they often serve as tools for numerous tasks, especially for image content description for classification or object recognition. Solutions based specifically on multiscale representations have been proposed: in [Valero 2011] for instance, a binary partition tree is processed so as to preserve only the most pertinent levels. The tree representations have been intensively used in order to extract effective content descriptions with attribute profiles long being the state-of-the-art method [Ghamisi 2014, Santana Maia 2021]. Despite extensive work, attribute profiles still remain however based on tree-cuts according to either manually or automatically chosen cut thresholds [Cavallaro 2017]. Deep learning methods have also been used to take into account the multiscale property, mainly by using pooling layers [Audebert 2016]. In the same time, in the machine learning community have emerged new tools to deal with hierarchical representations but they have been exploited quite recently in the remote sensing community. Classification of structured data like graphs or trees is a well-established field [Niepert 2016], the hierarchical representation is still often used to extract features rather than performing directly the classification. In addition, while supervised classification of "flat" labels (i.e. labels with no dependence between each other) is a marked up research field, hierarchical classification deals with labels organized

in a hierarchy. This paradigm suits particularly well to multi-scale remote sensing data as it would allow one to fully take advantage of the inherent hierarchical nature of the representation. Indeed, pixels or segments can belong to several classes, depending of the considered scale, e.g. from the building scale to a district and town scale in an urban context. Taking fully advantage of the hierarchical representations with segmentation and classification at multiple scales (thought a hierarchy of labels) is one challenge that is still to be solved.

1.3 Part outline and contributions

Chapter classification of remote sensing data with kernels and manifolds first sums up some of the contribution of the PhD thesis of Yanwei Cui [Cui 2017a]. It addresses some of the challenges related to multi-modal images at different resolutions thanks to the definition of a dedicated kernel that allows working directly on hierarchies. Then, I present the use of a manifold learning algorithm in the context of hyperspectral classification that is particularly efficient for weakly labeled datasets of images [Chapel 2014].

Chapter 2

Classification of Remote Sensing Data with Kernels and Manifolds

Contents

In this chapter, we investigate kernel-based strategies that make possible taking as input data with a tree-structured shape and capturing the topological patterns inside each structure through designing structured kernels. We apply the designed kernel in a multi-source and multi-resolution remote sensing image classification context. These works relate to the PhD on Yanwei Cui [Cui 2017a] and have been independantly published in [Cui 2015, Cui 2016a, Cui 2016b, Cui 2017b]. We also describe a manifold algorithm for hyperspectral image classification that we show to be very effective and efficient in a weakly labeled classification context. The results have been published in [Chapel 2014].

2.1 A kernel for learning on hierarchical image representations

Designing a kernel that can learn on hierarchical image representations, several critical aspects have to be taken into account:

- Unordered tree: we concentrate on kernels that can handle unordered trees (through which hierarchical image representations are often modeled). Such kernels should be able to capture the hierarchical relationships (i.e. parent-children relation) among the nodes.
- Numerical features: another important property in hierarchical image representations is that each node represents a region and that attributes of a region, e.g. color or size, are in general numerical features (at least at the time of the publication). This actually differed from many other domains where nodes are labeled by a fixed number of symbols.

- Robustness to structure distortion and noise: hierarchical representations heavily rely on the adopted construction techniques. These techniques build the tree in an unsupervised way, which tree structure might vary due to complexity of image contents, presence of the noise, or undesired regions grouped together. Thus, the resulting structures are less strict than the one in other domains such as chemoinformatics. This should be taken into account when designing the kernel.
- Complexity: the adopted kernel should be efficient and scalable. Unlike other domains, such as chemoinformatics, or nature language processing, where the data structures are relatively small, hierarchical representations often have a large number of nodes, and attribute of each node might be up to thousands of dimension.

To address all the aspects mentioned above, we propose a structured kernel based on the concept of subpath. It works on vertical hierarchical relationships among nodes in the structured data, with nodes equipped with numerical features. For its computation, we propose an iterative approach with a quadratic complexity w.r.t. the size (i.e. number of nodes) of structured data, and a quadratic complexity w.r.t. number of training samples. It is efficient when dealing with small structure size and limited number of training samples, and we call it BoSK (for Bag of Subpaths Kernel).

2.1.1 An instance of a convolutional kernel

On of the most standard way to construct valid kernels is to follow the convolution kernel framework [Haussler 1999]. It states that computing a kernel on a complex structure can be achieved by summing up kernels on its substructures. Following this framework, a large number of structured kernels have been proposed under different decompositions [Vishwanathan 2010]. In the case of unordered trees, the most popular solution is to rely on the concept of subpath [Kimura 2011] that has been identified as an appropriate substructure to ensure satisfying levels of expressiveness and effectiveness. We decompose either a tree $\mathcal{T}$ or a path $\mathcal{P}$ as a set of subpaths, that are paths connecting a node to one of its descendants (resp. ancestors) in $\mathcal{T}$ (resp. $\mathcal{P}$); the set of subpaths also includes individual nodes (see figure 2.1). Let us denote a subpath by $s_p = (n_{(1)}, n_{(2)} \ ..., n_{(p)})$, $s_p \in \mathcal{G}$ with $\mathcal{G}$ a tree or a path of length p. The kernel between two subpaths s_p and s'_p is defined as the product of atomic kernels computed on pairs of nodes $k(n_{(t)}, n'_{(t)})$ of the subpaths:

$$K(s_p, s'_p) = \prod_{t=1}^{p} k(n_{(t)}, n'_{(t)}) \tag{2.1}$$

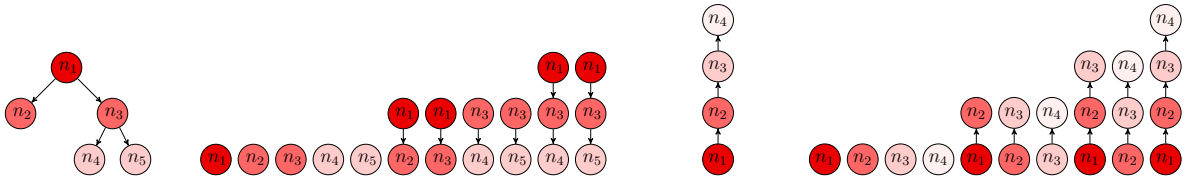
and the definition of BoSK between $\mathcal{G}$ and $\mathcal{G}'$ is written as, with μ_p a weight parameter:

$$K(\mathcal{G}, \mathcal{G}') = \sum_{p=1}^{P} \mu_p \sum_{s_p \in \mathcal{G}} \sum_{s'_p \in \mathcal{G}'} K(s_p, s'_p). \tag{2.2}$$

2.1.2 Efficient computation of BoSK

In the case of the (conventional) Gaussian atomic kernel $k(\cdot)$, $K(s_p, s'_p)$ can be written as:

$$K(s_p, s'_p) = \prod_{t=1}^{p} \exp(-\gamma ||\boldsymbol{x}_{n_{(t)}} - \boldsymbol{x}_{n'_{(t)}}||^2) = \exp(-\gamma ||\boldsymbol{x}_{s_p} - \boldsymbol{x}_{s'_p}||^2) = \ \langle \phi(\boldsymbol{x}_{s_p}) \ , \ \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}} \approx z(\boldsymbol{x}_{s_p})^T z(\boldsymbol{x}_{s'_p}) \ , \tag{2.3}$$



(a) A tree $\mathcal{T}$ and all its subpaths s_p.

(b) A path $\mathcal{P}$ and all its subpaths s_p.

Figure 2.1: Examples of structured data that can be extracted from hierarchical image representations, a tree $\mathcal{T}$, a path $\mathcal{P}$ and their subpaths.

where $\boldsymbol{x}_{s_p} = [\boldsymbol{x}^T_{n_{(1)}}, \boldsymbol{x}^T_{n_{(2)}}, \cdots, \boldsymbol{x}^T_{n_{(p)}}]^T \in \mathbb{R}^{pd}$ is the numerical feature of subpath s_p, being the concatenation the features of the nodes and $z(\boldsymbol{x}_{s_p})$ is the Random Fourier Feature approximation of dimension D. By using the explicit mapping function for the Gaussian kernel, BoSK can be rewritten as follows:

$$K(\mathcal{G}, \mathcal{G}') = \sum_{p=1}^{P} \sum_{s_p \in \mathcal{G}} \sum_{s'_p \in \mathcal{G}'} \langle \phi(\boldsymbol{x}_{s_p}), \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}} = \sum_{p=1}^{P} \langle \sum_{s_p \in \mathcal{G}} \phi(\boldsymbol{x}_{s_p}), \sum_{s'_p \in \mathcal{G}'} \phi(\boldsymbol{x}_{s'_p}) \rangle_{\mathcal{H}} = \tau(\boldsymbol{s})^T \tau(\boldsymbol{s}') \,, \quad (2.4)$$

where the set of vectors encoded into the feature space for each subpath s_p are aggregated inside a single vector $\tau(\boldsymbol{s}) = [\sum_{s_1 \in \mathcal{G}} z(\boldsymbol{x}_{s_1})^T, \cdots, \sum_{s_P \in \mathcal{G}} z(\boldsymbol{x}_{s_P})^T]^T$. The proposed approximation, SBoSK yields a linear complexity of $O(n|\mathcal{G}|dD)$, while the exact computation maintains a quadratic complexity of $O(n^2|\mathcal{G}|d)$.

2.2 BoSK for multi-source and multi-resolution image classification

To perform image classification from a hierarchical representation, we propose to combine BoSK computed on paths $\mathcal{P}$ and trees $\mathcal{T}$ respectively.

2.2.1 BoSK on path for multiscale contextual information

In pixel-wise classification, each pixel is represented by its spectral information, for instance r-g-b color information, or hyperspectral information in the hyperspectral remote sensing imagery. The spectral feature can be written as a d-dimensional vector and fed directly into a classifier. In such way, each pixel is treated independently, thus the spatial relationships among them are not preserved. However, spatially closed pixels often share similar spectral information and are more likely to belong to the same class. Without taking this image domain specification into account in the classification scheme, the resulting classification maps are often noisy and suffer from the "salt-and-pepper" effect. The hierarchy from a pixel to the whole image can be modeled by a path structure, where the nodes encode the feature of the regions and the edges model the hierarchical relationships among them. SBoSK applied on path structure allows explicitly taking into account the hierarchical relationships among ancestral regions from different scales, providing a powerful tools for multiscale context-based pixel-wise image classification. As illustrated in figure. 2.2, through the hierarchy, the ancestral regions encode the evolution of pixel from finer to coarser level, thus contextual information can be revealed.

2.2.2 BoSK on object spatial decomposition

A paradigm for tile image classification advocates the idea of relying on hierarchical representations, which are built using series of nested partitions or segmentations, rather than on the usual flat representation.

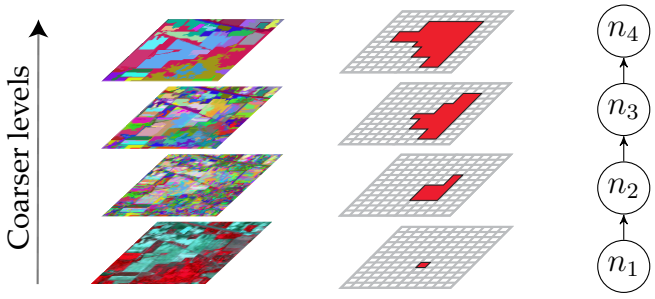


Figure 2.2: Contextual information extracted from hierarchical image representation. Each pixel (leaf of hierarchical representation) is considered as data instance to be classified, and described by features on the set of ancestral regions on the path $\mathcal{P}$ linking it to the root.

Regions at different scales are generated using multiscale segmentation tools and represented through a tree structure, where the root node represents the whole image and the leaves stand for the finest scale of segmentation. Regions are the nodes of the tree described by a set of features as the nodes attributes, and the relationships among regions are modeled through the edges. The hierarchical image representations can be constructed either by iteratively segmenting the image in 4 regions at successive scales (quad-tree representation as in figure 2.4a), or by multiscale segmentation algorithms (as shown in figure 2.3 and figure 2.4b).

(a) level 0 (b) level 1 (c) level 2 (d) level 3 (e) level 4

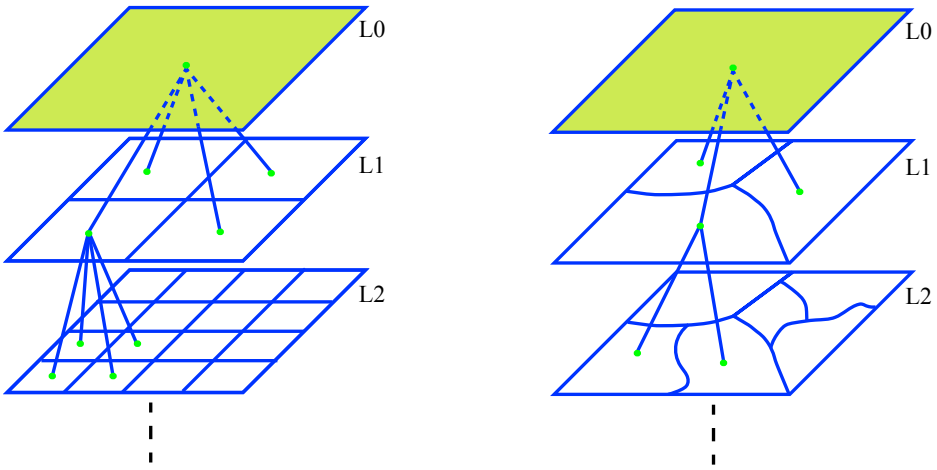
Figure 2.3: Illustration of a multiscale image segmentation from level 0 (whole image) to level 4 (the image is taken for the Merced dataset).

2.2.3 BoSK for multi-source image classification

We present here a novel multi-source and multi-resolution classification approach relying on BoSK and operating on a hierarchical image representation built from two images at different resolutions, possibly with different modalities. Both images capture the same scene with different sensors and are joined together through the hierarchical representation, where, for instance, coarser levels are built from a Low Spatial Resolution (LSR) or Medium Spatial Resolution (MSR) image while finer levels are generated from a High Spatial Resolution (HSR) or Very High Spatial Resolution (VHSR) image.

We combine BoSK computed on path $\mathcal{P}$ and trees $\mathcal{T}$ respectively. The final kernel between two data instances $K(\mathcal{G}, \mathcal{G}')$ is computed using a linear combination of the two BoSK:

$$\begin{aligned} K(\mathcal{G},\mathcal{G}') &= \rho \times K(\mathcal{P},\mathcal{P}') + (1-\rho) \times K(\mathcal{T},\mathcal{T}') \\ &= \rho \times \tau(\boldsymbol{s} \in \mathcal{P})^T \tau(\boldsymbol{s}' \in \mathcal{P}') + (1-\rho) \times \tau(\boldsymbol{s} \in \mathcal{T})^T \tau(\boldsymbol{s}' \in \mathcal{T}') \\ &= \left[\sqrt{\rho} \times \tau(\boldsymbol{s} \in \mathcal{P})^T, \sqrt{1-\rho} \times \tau(\boldsymbol{s} \in \mathcal{T})^T\right]^T \left[\sqrt{\rho} \times \tau(\boldsymbol{s}' \in \mathcal{P}')^T, \sqrt{1-\rho} \times \tau(\boldsymbol{s}' \in \mathcal{T}')^T\right], \end{aligned} \tag{2.5}$$



(a) Quad-tree representation used in Spatial Pyramid Matching Model

(b) Hierarchical representation generated with a multiscale segmentation algorithm

Figure 2.4: Illustration of a quad-tree representation and an arbitrary hierarchical representation.

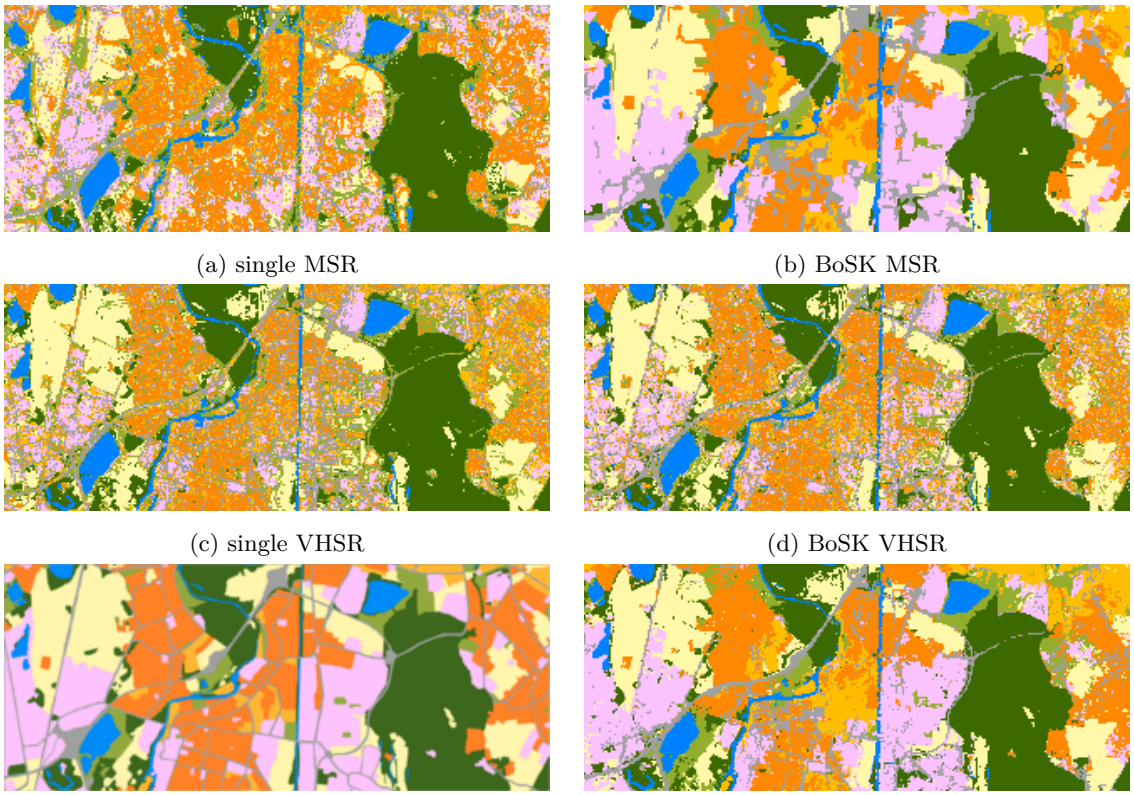
where $K(\mathcal{P}, \mathcal{P}')$ is BoSK on paths, and $K(\mathcal{T}, \mathcal{T}')$ is BoSK on trees, $\tau(\boldsymbol{s} \in \mathcal{P})$ and $\tau(\boldsymbol{s} \in \mathcal{T})$ are RFF embedding of $\mathcal{P}$ and $\mathcal{T}$ respectively, with a parameter $\rho \in [0, 1]$ that controls the importance ratio between the two kernels. Such embedding allows computing the fused kernel through inner product of concatenated feature vectors. It computes each data instance independently, yielding a linear complexity w.r.t. training sample size and maintaining the scalability of the proposed classification approach.

We evaluate the proposed approach focusing on urban land-use classification in the South of Strasbourg city, France. Two images are considered, both capturing the same geographical area with different sources: i) a MSR image captured by a Spot-4 sensor, containing 326×135 pixels at a 20 m spatial resolution, described by 4 spectral bands; ii) a VHSR image, captured by a Pleiades satellite, containing 13040×5400 pixels at a 0.5 m spatial resolution described by 4 different spectral bands. The classification results show that combining contextual and decomposition information leads to a significant improvement. Indeed we observe, for various training sample sizes, more than 4% improvement over BoSK on a single MSR image, and more than 10% improvement over BoSK on a single VHSR image. We can see in figure 2.5 that the prediction achieves a spatial regularization for the large regions, while providing precision for the small structures such as road networks.

2.3 A manifold learning algorithm for weakly labelled hyperspectral image classification

We now turn into describing the problem of leaning on high dimensional data when only few labeled pixels are available.

Although hyperspectral data live in a high dimensional space, the spectral correlation between bands is high and it is very unlikely that they occupy the whole space in an anisotropic manner: a manifold assumption then makes sense. Manifold learning algorithms assume that the original high dimensional data actually lie on an embedded lower dimensional manifold. Another path of interest to improve the classification performances is to focus on generative algorithms, that naturally entail a description of each class. Of course, a cornerstone of the statistical learning theory is that learning a boundary between classes (as



(a) single MSR (b) BoSK MSR

(c) single VHSR (d) BoSK VHSR

(e) Ground truth image (f) Combined

Figure 2.5: Classification maps for methods using single and multiple levels of a hierarchical image representation. Scenario 1: single level on Spot-4 image **(a)** vs. multiple levels contextual information on Spot-4 image **(b)**; scenario 2: single level on Pleiades image **(c)** vs. multiple levels spatial decomposition information on Pleiades image **(d)**; scenario 3: combination of contextual and spatial decomposition information **(f)**. Ground truth image **(e)** is also given as reference.

it is done with discriminative classifiers) is a simpler problem than training a model for each of them. However, from a theoretical point of view, generative learning is often more efficient than discriminative learning when the number of features is large compared to the number of training samples [Ng 2001] while discriminative models are often better asymptotically. In the context of hyperspectral image classification, generative classifiers are thus particularly interesting since, as pointed out in [Bioucas-Dias 2013], "supervised classification faces challenges related with the unbalance between high dimensionality and limited availability of training samples". We focus on the PerTurbo algorithm [Courty 2011] for hyperspectral image classification.

2.3.1 Manifold class description and classification algorithm

Algorithm. The idea behind PerTurbo is to build the predictive function in an implicit manner, within two steps. In the first step, the geometry of the set of training examples for each class ℓ is characterised. Then, a similarity metric adapted to these sets of geometric models is derived, so that the predictive function reads as the minimum distance of a test sample η to those models.

Let us denote $\mathcal{S}_\ell$ the set of all the N_ℓ training examples with label ℓ. We assume that each set $\mathcal{S}_\ell$ is embedded into a dedicated Riemannian manifold $\mathcal{M}_\ell$, whose geometric structure can be expressed in terms of the Laplace-Beltrami operator. This operator can be efficiently approximated by the heat kernel (modeling the propagation of a variation of temperature along a manifold), this latter being in turn approximated by the spectrum of the Gaussian kernel $K(\mathcal{S}_\ell)$, the Gram matrix of the training set $\mathcal{S}_\ell$, whose (i^{th}, j^{th}) term is given by $K_{ij}(\mathcal{S}_\ell) = k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^\top \cdot \phi(\boldsymbol{x}_j)$. When a test sample η is considered, a similarity measure between a class and η can be derived from the extent to which its inclusion to the class changes the geometric characterisation of the associated manifold $\mathcal{M}_\ell$. More formally, the projection of sample η onto the subspace spanned by $\phi(\mathcal{S}_\ell)$ is given by:

$$r(\eta \to \mathcal{M}_\ell) = K(\mathcal{S}_\ell)^{-1/2} k(\mathcal{S}_\ell, \eta). \tag{2.6}$$

The perturbation measure of class ℓ by a sample then reads:

$$\tau(\eta, \mathcal{M}_\ell) = \|\phi(\eta)\|^2 - \|r(\eta \to \mathcal{M}_\ell)\|^2 = 1 - k(\mathcal{S}_\ell, \eta)^\top K(\mathcal{S}_\ell)^{-1} k(\mathcal{S}_\ell, \eta). \tag{2.7}$$

Test sample is then classified into the class that provides the smallest perturbation, i.e. $\arg\min_\ell \tau(\eta, \mathcal{M}_\ell)$.

Interpretability of the model: similarity between classes The geometric characterisation of each class by the kernel matrix $K(\mathcal{S}_\ell)$ carries extra information that can be used to describe classes, allowing for example measuring the separability of the different classes. The projection of sample on a manifold can be generalised in order to project all the samples $\mathcal{S}_{\ell_1}$ of a given class ℓ_1 into the subspace spanned by $\phi(\mathcal{S}_{\ell_2})$: $r(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}) = K(\mathcal{S}_{\ell_1})^{-1/2} k(\mathcal{S}_{\ell_2}, \mathcal{S}_{\ell_1})$. Hence, the Gram matrix associated to this projection is

$$K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}) = r(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2})^\top \cdot r(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}) = K(\mathcal{S}_{\ell_2}, \mathcal{S}_{\ell_1})^\top K(\mathcal{S}_{\ell_2})^{-1} K(\mathcal{S}_{\ell_2}, \mathcal{S}_{\ell_1}). \tag{2.8}$$

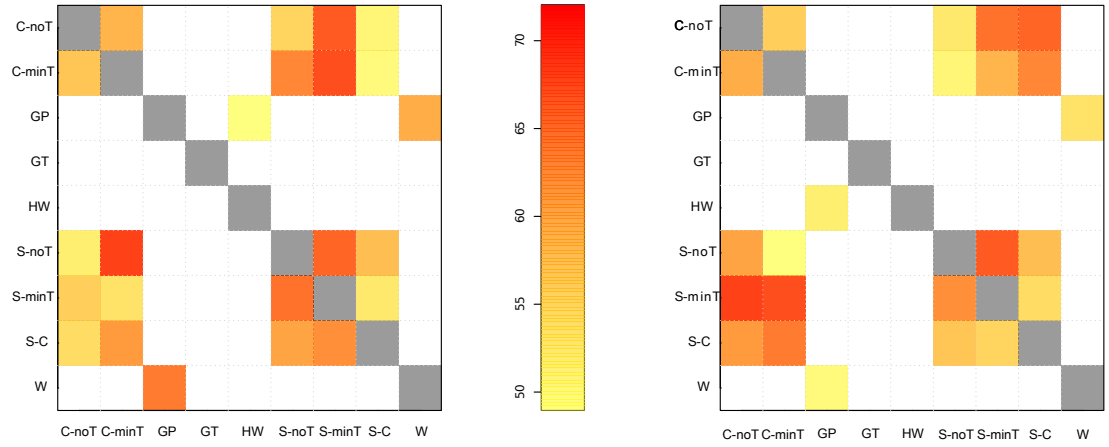
$K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2})$ is a surrogate kernel of $K(\mathcal{S}_{\ell_1})$: both kernels are constructed on the same set of eigenvectors and eigenvalues, but they are evaluated on different sets. As such, classes with similar geometry will have "similar" kernel matrices while those with manifolds lying in different spaces will have "different" kernel matrices. The degree of agreement between two matrices can be evaluated thanks to the empirical alignment of the kernel matrix $K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2})$ with the kernel matrix $K(\mathcal{S}_{\ell_1})$ with respect to the sample $\mathcal{S}_{\ell_1}$:

$$A(\mathcal{S}_{\ell_1}, K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}), K(\mathcal{S}_{\ell_1})) = \frac{\langle K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}), K(\mathcal{S}_{\ell_1}) \rangle_F}{\sqrt{\langle K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}), K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}) \rangle_F \langle K(\mathcal{S}_{\ell_1}), K(\mathcal{S}_{\ell_1}) \rangle_F}} \tag{2.9}$$

where $\langle C, D \rangle_F$ stands for the inner product between matrices C and D.

2.3.2 Behavior on low-sized training sets

When comparing the performances of the algorithms on several hyperspectral datasets (Indian Pines, Pavia University, Pavia Center, Washington DC Mall), we observe that it provides significantly better accuracy rates than SVM when the number of training samples is low, while being faster to compute. Calculating the similarity matrix between each class and its surrogates gives a table layout; a function of the obtained values, as well as a function of the intensity of the values of the confusion matrix, are represented in figure 2.6. One can notice the structural similarity of the two matrices, confirming the idea that an a priori on the classification results can be obtained from the computation of the similarity matrix.



(a) Similarity matrix (b) Ranks (c) Confusion matrix

Figure 2.6: Measures of separability of the different classes of the *Indian Pines* dataset. Cells represent, in Figure (a) a function of the measure of similarity $A(\mathcal{S}_{\ell_1}, K(\mathcal{S}_{\ell_1} \to \mathcal{M}_{\ell_2}), K(\mathcal{S}_{\ell_1}))$ between the classes; in figure (c), a function of the percentage of pixels of class in column that have been classified as the class in row. The function used is the rank; colour intensity in Figure (b) represents the rank of each cell: the most similar pairs of classes and those with highest error rates being coloured in dark red (cells with the lowest ranks are not coloured and diagonal cells are coloured in grey in both cases).

Part II

Contributions to Machine Learning for Time Series

Chapter 3

Machine Learning Algorithms for Time Series

Contents

Time series are ubiquitous and their analysis or learning requires the use of dedicated methods, able to take into account the additional temporal dimension of the data. There has been much effort devoted to the field in the two last decades, the key being to be able to either design effective dissimilarity measures that can take time series as input or designing algorithms that rely on a suitable representation of the data. In this chapter, I give a brief overview of the classical methods for machine learning for time series on which our works on the topic rely on. Some related challenges are then described at the end of the chapter.

Time series are sequences of features $\boldsymbol{x} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_{n-1})$ with associated time stamps $(t_0, t_1, \cdots, t_{n-1})$ that lie in a p-dimensional space. When $p = 1$, times series are said *univariate*, otherwise *multivariate*. The temporal dimension can be included in different manners: i) only the ordering of the observations matters (as in the DTW similarity for instance) ii) the time stamps are taken into account. Some approaches like the convolutional ones suppose that the time is regularly spaced while some others allow dealing with irregularly sampled time series.

3.1 Dissimilarity measures for time series

3.1.1 DTW and its variants

DTW. The state-of-the-art algorithm for assessing dissimilarity between time series is without a doubt Dynamic Time Warping (DTW, [Sakoe 1978]), whose extensions to multivariate time series have been proposed in [Ten Holt 2007]. DTW takes into account temporal shifts and distortions, but can be sensitive to noise or outliers (as all the time instants are taken into account) and does not take explicitly the time into account (which may be a drawback in some applications).

In its standard form, given two time series $\boldsymbol{x}$ and $\boldsymbol{y}$ of the same dimensionality p, DTW is defined as:

$$\mathrm{DTW}(\mathbf{x},\mathbf{y}) = \min_{\pi\in\Pi(\mathbf{x},\mathbf{y})} \sum_{(i,j)\in\pi} d(\mathbf{x}_i,\mathbf{y}_j) = \min_{\pi\in\Pi(\mathbf{x},\mathbf{y})} \langle C(\boldsymbol{x},\boldsymbol{y}),\pi\rangle \tag{3.1}$$

where the entry $C_{i,j} = (d(\boldsymbol{x}_i,\boldsymbol{y}_j))_{i,j}$ of $\boldsymbol{C}(\boldsymbol{x},\boldsymbol{y})$ represents the distance between $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$, where d is the ground cost (in most cases, the squared Euclidean distance $d(\mathbf{x}_i,\mathbf{y}_j) = \|\mathbf{x}_i - \mathbf{y}_j\|^2$). An alignment π is a sequence of pairs of time frames which is considered to be admissible iff it belongs to the following set of constraints:

$$\begin{aligned}\boldsymbol{\Pi}(\mathbf{x},\mathbf{y}) = \{\,\pi \in \mathbb{R}_+^{n\times m} | \pi(0,0) = 1, \pi(n,m) = 1,\\ \text{for each } \pi(i,j) = 1, \text{ we have either } \pi(i+1,j) = 1 \text{ or } \pi(i,j+1) = 1 \text{ or } \pi(i+1,j+1) = 1\}\,.\end{aligned} \tag{3.2}$$

As such, it defines a connected path between the beginning and the end of the time series (figure 3.1). Efficient computation of the above-defined similarity measure can be performed in quadratic time using dynamic programming thanks to a recurrence formula.

Many variants of this similarity measure have been introduced. For example, the set of admissible alignment paths can be restricted to those lying around the diagonal using the so-called Itakura parallelogram or Sakoe-Chiba band, or a maximum path length can be enforced [Zhang 2017].

softDTW. A differentiable variant of DTW, coined softDTW, has been introduced in [Cuturi 2017] and is based on previous works on alignment kernels [Cuturi 2007]. It replaces the min operation in Equation (3.1) by a soft-min operator $\min^\gamma$ whose smoothness is controlled by a parameter $\gamma > 0$, resulting in the DTW_γ distance:

$$\mathrm{DTW}_\gamma(\mathbf{x},\mathbf{y}) = \min_{\pi\in\Pi(\mathbf{x},\mathbf{y})}{}^{\gamma} \sum_{(i,j)\in\pi} d(\mathbf{x}_i,\mathbf{y}_j) = -\gamma \log\left(\sum_{\pi\in\Pi(\mathbf{x},\mathbf{y})} e^{-\sum_{(i,j)\in\pi} d(\mathbf{x}_i,\mathbf{y}_j)/\gamma}\right). \tag{3.3}$$

In the limit case $\gamma = 0$, $\min^\gamma$ reduces to a hard min operator and DTW_γ is defined as equivalent to the DTW algorithm.

Global alignment kernel. Instead of considering the minimum over the set of possible alignments in eq. (3.1), [Cuturi 2007] argue that integrating over the whole set of alignments provides a richer statistic, and propose the global alignment kernel as the exponentiated soft-minimum of all alignment distances:

$$k_{\mathrm{GA}}(\mathbf{x},\mathbf{y}) = \sum_{\pi\in\Pi(\mathbf{x},\mathbf{y})} \exp^{-\langle C(\boldsymbol{x},\boldsymbol{y}),\pi\rangle}. \tag{3.4}$$

They show that the computational effort to compute k_{GA} is $O(nm)$, similar to DTW, and [Cuturi 2011] also proposes variants of the kernel by considering a smaller subset of admissible alignments.

Gromov-DTW. When it comes to heterogeneous time series, [Cohen 2021] define GDTW that extends the Gromov-Wasserstein distance (eq. (6.8) in section 6) to time series:

$$\mathrm{GDTW}(\boldsymbol{x},\boldsymbol{y}) = \min_{\pi\in\boldsymbol{\Pi}(n,m)} \sum_{i,j,k,l} L(d_X(\boldsymbol{x}_i,\boldsymbol{x}_k), d_Y(\boldsymbol{y}_j,\boldsymbol{y}_l))\pi_{i,j}\pi_{k,l} \tag{3.5}$$

in which L is a loss function that measures the alignment of the pairwise distances and the constraint set $\boldsymbol{\Pi}(\boldsymbol{x},\boldsymbol{y})$ is the set of all the admissible DTW alignment matrices (eq. (3.2)). They provide an

"approximate" Frank-Wolfe algorithm [Frank 1956] with a fixed step size to solve the problem rather than optimizing it (see Section 7.1.2 for a description of the Frank-Wolfe algorithm), ensuring that, at each iteration of the algorithm, the solution is an alignment matrix. The price to pay of this approximation is that there is no proof of convergence of the algorithm. Authors claim that the algorithm works well in practice, but in [Vayer 2020b], we show that it may have difficulties to reach a global optimum (see also figure 5.4 in section 5.2), which may come from this approximation. Together with the authors of [Vayer 2020b], we conjecture that projecting within the set of admissible alignment matrices at the end of the Frank-Wolfe algorithm rather than at each iteration may be a better alternative as, in practice, we observe that few optimal steps differ from the fixed one, and that the final solution would be "not far away" from a true alignment matrix. It would have the advantage of ensuring the convergence of the algorithm, which is not the case for [Cohen 2021]. We did not investigate further the impacts of this alternative solution though.

3.1.2 Other dissimilarity measures

There exist many dissimilarity measures for time series, whose use depends on the context at stake. For example, the set of L_p distances can be used when the time series are registered with no temporal distortion. Dedicated measures such as temporal instances of edit distances have also been instantiated, e.g. the basic idea of the Longest Common SubSequences (LCSS) problem is to find the longest common subsequence between two sequences, in which some elements can be unmatched or left out.

FGW distance on time series. The fused Gromov-Wasserstein distance defined in section 7.1 can be used for time series (see an example in figure 7.5). In that case, they are viewed as special instances of non-directed graphs rather than directed ones (see figure 1). As a consequence, the knowledge of the beginning and the end of the time series is lost, introducing a non-desirable invariance. Figure 3.1 presents one example of such behavior on a toy example: while the two time series are different, FGW is unable to distinguish them.

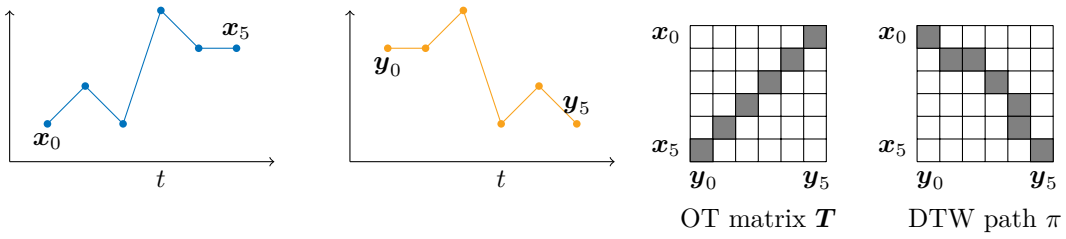


Figure 3.1: The isomorphism property is not sought when dealing with time series: the two time series have a FGW distance of 0. The DTW alignement produces a path from $(\boldsymbol{x}_0, \boldsymbol{y}_0)$ to $(\boldsymbol{x}_5, \boldsymbol{y}_5)$.

3.2 Embedding time series

There exists a wide number of algorithms that aim at embedding time series in a common Euclidean space that captures the main features of the data and in which they can be easily compared. The objective is to find a latent space that best captures the main features of the data or that allows the best solution of the learning task at stake. From the historical Fourier and wavelet transformations, the most prominent

families of methods that are developed nowadays are based on deep learning techniques: a query on Google scholar for "Time series" + "deep learning" for the last 5 years returns almost 100 000 results. In this section, I review some of the solutions in which our works place themselves into.

DTW-based Embedding of time series in a common latent space. Canonical Time Warping (CTW) [Zhou 2009] aims at aligning two sequences in a common latent space (of reduced dimensionality). It combines Canonical Correlation Analysis with DTW for temporal alignment, allows for local spatial deformations and can handle heterogeneous times series (i.e. with different modalities or dimensions). When a non-linear projection is sought, its extension Deep CTW [Trigeorgis 2016] relies on deep architectures to transform the sequences. [Deng 2020] define an invariant subspace in which the DTW distance between samples and their reconstruction is minimized.

Codebook-based representations. The bag-of-words (BoW) model consists in representing an instance using a histogram of word occurrences. Albeit the temporal order is lost, it is able to capture structural information by defining both local and global characteristics. The quantization that is involved might lead to information loss, nonetheless, it also allows robustness to noise. Many codebook-based representations use k-means to select k-codewords. The selected codewords correspond to cluster centers, and each local feature is assigned to the nearest cluster center. From the Bag-of-Patterns to Bag-of-SFA-Symbols, there exist many implementations of the BoW model for representing time series; a review in the context of time series classification can be found in [Bagnall 2017]. In [Bailly 2015b] and [Bailly 2015a], we propose a representation based on SIFT-words for time series, together with an application on a remote sensing context [Bailly 2016]. SIFT features are straightforward 1D adaptations of the Scale-Invariant Feature Transform (SIFT [Lowe 1999]) framework introduced in Computer Vision. Section 4.2 gives more details about the algorithm and its performances.

Shapelet-based representations. Shapelets are discriminative time series subsequences (of length $L < n$). The original time series are then described as a vector of distances between each of the shapelet and the optimal match localization:

$$d(\boldsymbol{x}, \mathbf{s}_k) = \min_{j=1,\cdots,J} \sum_{\ell=1}^{L} (x_{j+\ell-1} - \mathbf{s}_{k,\ell})^2$$

in which $\mathbf{s}_k$ is the kth. Shapelets-based methods also provide interpretability of the results since it is possible to extract the most discriminative shapelet(s). [Grabocka 2014] propose to learn the shapelets instead of searching for the best of them. In [Bailly 2018], we propose an enhancement of the learning time series shapelets (LTS) algorithm based on the insertion of adversarial time series in the training procedure. Basically, we show that LTS is a special case of CNN for 1d data and we propose to introduce an adversarial regularization to improve the learning. Empirical evaluation on the UCR dataset shows that it improves the overall classification accuracy when compared to the LTS algorithm alone.

Dynamic topic models. When it comes to discovering specific patterns in time series, topic models are state-of-the-art techniques. They are mixture models originally designed for discovering *topics* into text *documents*. Those documents are then represented as bag of features and the topics are then a distribution over features. They have been extensively studied in the text mining community: intuitively, given the occurrence of some words, one can defined topics as clusters of similar words. The patterns of words are estimated through hierarchical probabilistic models and a dynamic version, that captures

the evolution of topics in a sequentially organized corpus of documents, has been defined in [Blei 2006]. In another line, [Wang 2006] propose a LDA-style topic model that captures how the structure changes over time. Efficient approximate posterior inference techniques for determining the evolving topics are usually used to estimate the mixture components and proportions. For these methods, time series are hence seen as bags of timestamped features.

3.3 Some challenges for machine learning for time series

Domain adaptation and/or heterogeneous setting. Learning under distribution shift is one paradigm of major interest in the machine learning literature. Indeed, the training and the test data are often subject to collection bias or can be collected under heterogeneous conditions, because of different times of measurement, contexts or even measurement modalities (e.g. when different sensors are used to measure related quantities). This first challenge questions our ability to transfer knowledge across different domains; the problematic of *domain adaptation* or *transfer learning* is even more significant when dealing with time series as temporal shifts can be encountered in addition to the feature distribution shift. As an example, let us consider the automatic land use and land cover classification from time series of satellite images problem, which is of paramount relevance to assess vegetation and crop status. The bio-geo-physical variables exhibit distinct temporal structures due to the difference in the geographical area of collection, the modality of the measurements, the time of collection etc. The lack of annotated data (due to the high cost and human resources needed to characterize and classify land cover through field campaigns) makes the problem of combining the knowledge from heterogeneous data even more significant. In that context, we propose in [Bailly 2017] to use a kernel manifold alignment [Tuia 2016] method for domain adaptation of remote sensing time series, allowing one to realign the time series in the same domain in which a classification is then performed.

When it comes to domain adaptation for time series that live in the same space, existing methods mainly rely on domain adversarial neural networks, such as [Wilson 2020, Liu 2021] to name a few. When they live on different spaces, most methods rely on DTW, flagship examples being Canonical Time Warping [Zhou 2009] or Gromov-DTW [Cohen 2021].

Learning the importance of the time within the algorithms. Learning on time series may take very diverse forms, depending on the algorithms that are implied. Their choice depends on the invariants and the nature of the data at stake. Real world time series data are easily affected by complex temporal transformations, that may differ from one domain or class to another. As an example, when dealing with remote sensing time series, some classes may be affected by long-term changes (e.g. those related to urbanization) or cyclic changes (e.g. agriculture classes); some of them may also be temporally distorted, e.g. with a shift harvest/collect from one parcel to another or different maturation speed. A great deal of effort has been put into studying how to align the time series data, but less into studying which a priori should be put onto the analysis. The main option at this moment is the use of ensemble techniques that combine output from base models [Lines 2018, Ismail Fawaz 2019]. Designing methods able to learn the type of invariance or the distortion present in the data is then of prime importance.

From infrequent measurements to continuous monitoring. From infrequent and parsimonious measurements of signals (as represented in the UCR archive [Chen 2015] that contains 85 datasets of univariate time series of average length of 460 time stamps equally spaced in time, an average training set size of 530 series), rapid advances in technologies are generating a rapid growth both in the size,

but also in the complexity of the time series data. Introducing or relying on continuous time models is then of paramount importance: even if the time series are discrete, the underlying process is often continuous and leads to unequally spaced measurements. It has been shown that we should be very careful when interpreting results obtained with a discrete-time model when analyzing continuous processes [Loossens 2021]. It also necessitates the development of scalable algorithms able to deal with the related amount of data. The recent success of the ordinary differential equations (ODE) in various networks [Chen 2018] is a flagship example of this trend. It has led to the definition of continuous-depth models able to handle irregularly-sampled time series data. Among related works, we can cite [Rubanova 2019] who show that ODE-based models outperform their RNN-based counterparts on irregularly-sampled data; in the case of partially-observed irregularly-sampled multivariate time series, [Kidger 2020] propose Neural Controlled Differential Equations. Nevertheless, during the internship of François Painblanc [Painblanc 2019], we found out that neural-ODE were costly to train and prone to overfitting. In another line of research, the goal of continuous time modeling approaches is to formalize the laws that govern the evolution of the observed processes into a mathematical framework. Continuous-time models, like the Ornstein–Uhlenbeck model [Uhlenbeck 1930] (that can be seen as continuous autoregressive model), rely on differential equations and describe a system that is evolving through time. The aim is then to approximate the parameters of the underlying process that fit the most to the observed data. While the treatment of discrete-time model is well established in the community, less attention has been paid to the continuous-time ones and that, to my opinion, is one of the challenges when dealing with time series. Leaning toward this aim, and for the sake of efficiency, we have proposed in [Gloaguen 2021] a 2-step scheme to cluster trajectories i) non-temporal pre-clustering of the data ii) in a refinement step, model of the series segments using a continuous-time model (see section 4.1).

3.4 Part outline and contributions

Here is a short description of the 2 following chapters of the manuscript that give my main contributions in the field of learning for time series.

Chapter New representations for time series deals with building new embedding schemes for learning on time series. We first propose a clustering algorithm that relies on Ornstein–Uhlenbeck processes to take into account the continuous nature of trajectories. We next describe our work for designing a new time series representation based on temporal SIFT descriptors, allowing taking into account the local structure present in the data.

Chapter Building sensible metrics for time series provides new algorithms for comparing time series. The first one defines a time-sensitive local kernel between time series that is able to take into account irregularly sampled time series. A second contribution specifically deals with the case of time series that are *incomparable*, that is to say that are described by unregistered or different features. The proposed framework combines a latent global transformation of the feature space with the widely used Dynamic Time Warping (DTW). The latent global transformation captures the feature invariance while the DTW deals with the temporal shifts.

Chapter 4

Machine Learning Relying on Sensible Time Series Representations

Contents

In this section, I sum up some works that aim at defining new sensible representations for time series. The first work [Gloaguen 2021] has been performed during the post-doc of Pierre Gloaguen and defines a generic framework for the clustering of large trajectory data sets. A trajectory is described by both the successive positions and velocities. Each trajectory is seen as a succession of movement modes in which an Ornstein Uhlenbeck process is used to model the velocity. A cluster of trajectories is then characterized by its distribution over movement modes. This framework is then specified in continuous time and space, which makes its formulation insensitive to GPS sampling, relaxing assumptions of previous models of the literature that are based on space quantization It relies on a global latent representation of the feature space in which the time series can be compared. I then described the Bag-of-Temporal-SIFT-Words method that has been defined during the PhD of Adeline Bailly [Bailly 2018]. The method extracts SIFT-based keypoints [Lowe 1999], then describes those keypoints through gradient magnitude and then represents the time series as a bag of words of the quantized version of the keypoints. In that case, and contrary to the first method, the temporal consistency all along the time series is lost but it allows one to handle temporal shifts. In section 5.1, a solution is proposed to take into account the temporal dimension.

4.1 Inference for Sequences of Ornstein Uhlenbeck Processes for time series clustering

We propose in [Gloaguen 2021] a new modelling framework to cluster sequences of a large amount of trajectories recorded at potentially irregular frequencies. The model is specified within a continuous time

framework, being robust to irregular sampling in records and accounts for possible heterogeneous movement patterns within a single trajectory. It partitions a trajectory into sub-trajectories, or movement modes, allowing a clustering of both individuals' movement patterns and trajectories. The modelling framework aims to account for two levels of heterogeneity possibly present in trajectory data: i) heterogeneity of an individual's movement within a single trajectory, and ii) heterogeneity between observed trajectories of several individuals. Time series are then represented as successive Ornstein Uhlenbeck Processes. The following three problems must be solved: i) characterizing different movement modes present in the dataset; ii) for each observation, estimating in which movement modes it belongs; iii) clustering together trajectories that have the same distribution over movement modes. We first start by defining a parametric framework to model trajectory data.

4.1.1 Continuous time model for the movement modes

A movement mode is assumed to be characterized by a specific correlated velocity model, defined in a continuous-time framework. Formally, during a time segment $[\tau_1;\tau_2]$, if an individual adopts the movement mode k, then its velocity process $(V_t)_{\tau_1\leq t\leq\tau_2}$ is assumed to be the solution of the following Stochastic Differential Equation (SDE):

$$V_t = -\Gamma_k\left(V_t-\mu_k\right)t + \Sigma_k W_t, \quad \tau_1 \leq t \leq \tau_2 \text{ and } V_{\tau_1} = v_{\tau_1} \tag{4.1}$$

where $\mu_k \in \mathbb{R}^2$ is the asymptotic mean velocity of the k-th movement mode; Γ_k is an autocorrelation parameter, Σ_k is a diffusion term, W_t is a standard Brownian motion. The solution to eq. (4.1) is a well known continuous time stochastic process, the Ornstein Uhlenbeck Process (OUP) [Uhlenbeck 1930] (see figure 4.1). The resulting process $(X_t)_{\tau_1\leq t\leq\tau_2}$ is known as an Integrated Ornstein Ulhenbeck Process (IOUP), which remains a Gaussian Process.

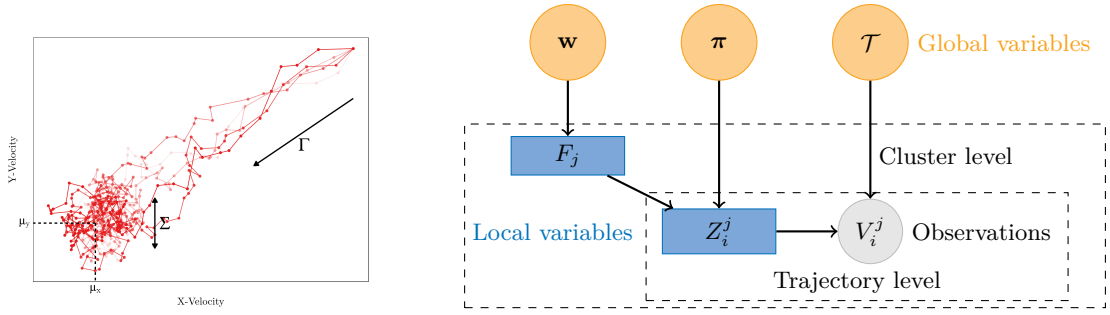


Figure 4.1: (Left) Five realizations of bivariate Ornstein Ulhenbeck Processes starting at position $v_0 = (0,0)$ (top right corner) (Right) Graphical representation of the hierarchical structure of the model, in which (global variables) $\mathbf{w}$ represents the trajectory cluster weights, $\boldsymbol{\pi}$ is the set of all unknown movement modes weights, $\mathcal{T}$ represents the set of unknown movement parameters, (local variables) F_j is the cluster of the jth trajectory, $Z_i^j = k$ if the velocity belongs to the kth movement mode.

4.1.2 A 2-step clustering approach

In order to perform scalable parameter inference and clustering of both trajectories and GPS observations (into movement modes), we adopt a pragmatic two-step approach that takes advantage of the inherent properties of the OUP and are briefly described hereafter.

Step-1. A first dual clustering is performed based on a simpler independent Gaussian mixture model, in order to estimate potential movement modes and trajectory clusters: it allows getting rid of within mode autocorrelation in the inference, and therefore eases the computations. The Gaussian hypothesis in this case is rather natural, as the OUP stationary distribution is Gaussian. For this first step, we define a hierarchical model and perform a Bayesian estimation of the parameters using stochastic variational inference, making the algorithm scalable (unlike traditional Gibbs sampling procedures). Figure 4.1 (right) gives the associated graphical model.

Step-2. Among the estimated movement modes, only those meeting a temporal consistency constraint are kept. Parameters of these consistent movement modes are then estimated and used to reassign observations that were assigned to inconsistent movement modes. It ensures that only trajectory segments for which this stationary distribution was reached are kept to estimate movement modes.

4.1.3 Experiments on GPS trajectories

Experiments on simulated data. A data set of 40 trajectories containing overall 8 000 observations is simulated, according to a model with 2 trajectory clusters, the two clusters being composed of respectively 2 and 3 movement modes. Simulated data are shown in figure 4.2 (left). However, two extra movement modes are estimated, corresponding to the transition phases towards the light blue movement mode and leaving from it (figure 4.2 (Step 1)). Including the second step, movement modes for inconsistent sequences are re-estimated (figure 4.2 (Step 2)). The trajectory cluster assignation is 100% right.

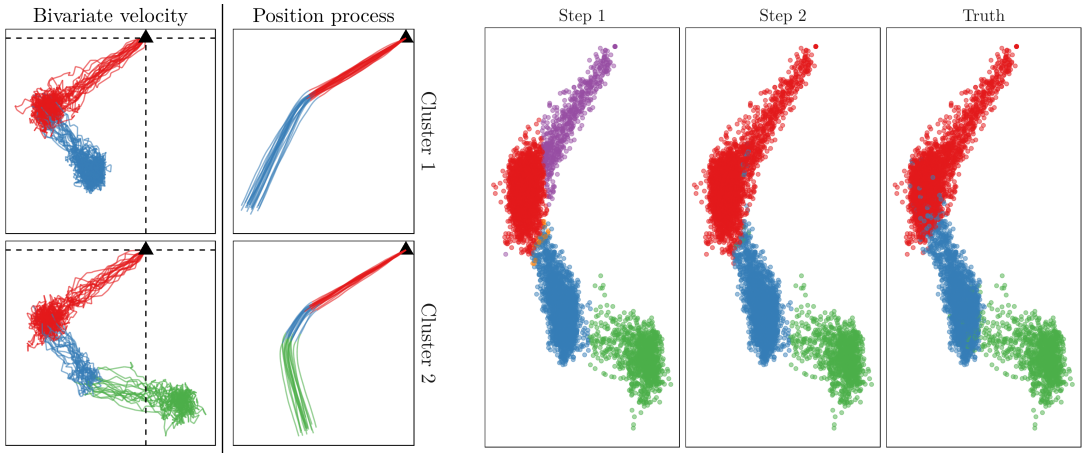


Figure 4.2: (Left) Three movement modes are shared by two clusters and the first two movement modes are present in both clusters. (Right) Estimated movement modes (in the bivariate velocity space) after one and two steps. The ground truth is shown on the right.

Experiments on maritime traffic data. The dataset records 6 months of AIS data of vessels steaming in the area of the Ushant traffic separation scheme (in Brittany, West of France). This is an area with one of the highest maritime traffic density in the world, with a clear separation scheme of two navigation lanes. Different kinds of vessels are sailing in the area, from cargos and tankers with high velocity and straight routes to sailboats or fishing vessels with lower speed and different sailing directions. As such, the area is highly monitored to avoid collision or grounding, and a better analysis and understanding

of the different ship behaviors is of prime importance. The whole trajectory dataset is shown on Fig. 4.3 and we made it online[1]. It consists in 18,603 trajectories, gathering at all more than 7 million GPS observations. As no ground truth is available, we cannot determine which solution

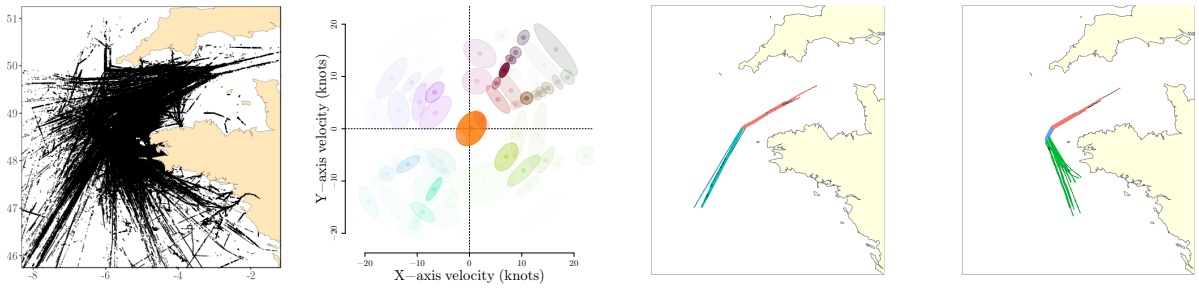


Figure 4.3: (Left) Whole trajectory dataset (Middle left) Mean and covariance of the 81 estimated movement modes (Right) Typical trajectories from two estimated clusters. In the first part of the trajectories starting at the South, clusters are differentiated by the movement mode. Then, all trajectories follow the same route.

fits better but we note by a visual inspection of the clusters that the competitor clustering fails at differentiating ships that follow the same traffic lane (i.e. identifying sub-routes in the main traffic lanes). One should also note that, by assigning movement modes to the observations, our method provides extra explanation for cluster assignments, compared to k-means that solely relies on inertia minimization.

4.2 Time series classification based on local features representation

4.2.1 Bag-of-Temporal-SIFT-Words

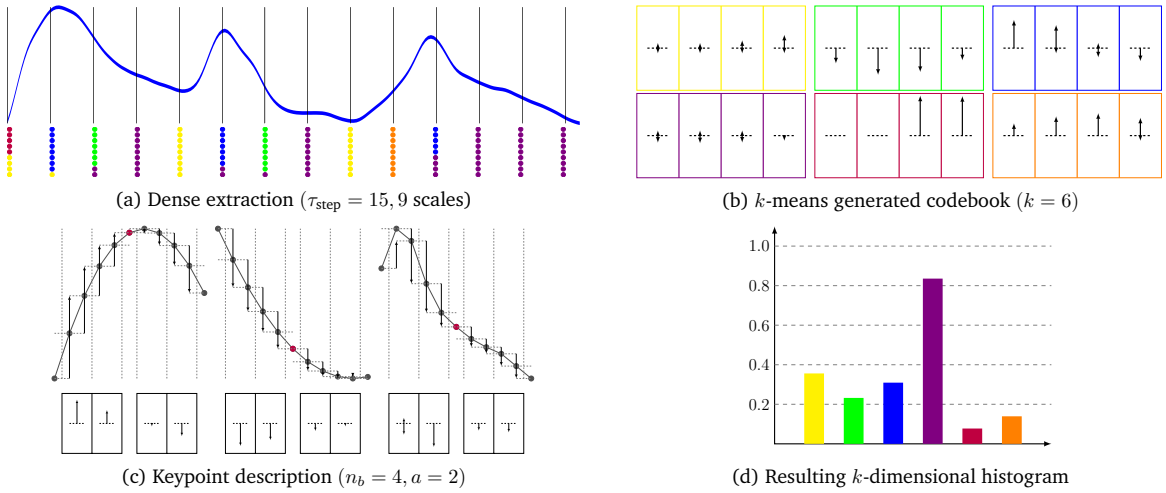
Bag-of-Temporal-SIFT-Words (BoTSW) is a method that aims at providing a new representation of time series that relies on SIFT features. Those features have been shown to be efficient in the computer vision community thanks to their ability to describe local behavior of images.

Method. In a nutshell, the algorithm combines the extraction of descriptive features at regular time steps in the time series with a histogram representation. The raw time series are transformed into a new representation that gathers the different descriptive characteristics into similarity groups, the new representation is then fed into a classifier. It is based on the Scale-Invariant Feature Transform (SIFT [Lowe 1999]) method that has been originally introduced in the computer vision community, and has several useful properties such as scale and location invariance. We have proposed two variants of the method: the Bag-of-Temporal-SIFT-Words [Bailly 2015b] and its dense counterpart (D-BoTSW) [Bailly 2015a], which is sum up here.

The dense Bag-of-Temporal-SIFT-Words (D-BoTSW) algorithm can be divided into the following steps. First, we extract keypoints inside time series, i.e. we extract a list of points at regular time step that serve as bases for our future descriptors. Then, we describe these keypoints using 1D-SIFT features: these features use neighboring points in order to describe the neighborhood of each keypoints and to detect characteristics such as peaks or valleys inside time series. Once those descriptors are defined, a codebook is generated, i.e. similar features are gathered together in order to simplify the representation. To do so, we transform each time series into a new representation corresponding to a normalized histogram of

[1] https://github.com/rtavenar/ushant_ais

words occurrences, which allow us to easily compare two time series. Finally, we perform the classification on the new representation (here a linear SVM is used). The proposed algorithm is robust to both noise and temporal shifts and is described in figure 4.4.



(a) Dense extraction ($\tau_{\text{step}} = 15, 9$ scales)

(b) k-means generated codebook ($k = 6$)

(c) Keypoint description ($n_b = 4, a = 2$)

(d) Resulting k-dimensional histogram

Figure 4.4: Bag-of-Temporal-SIFT-Words approach overview: (a) a time series and its dense (regular) extracted keypoints; (b) Codewords obtained using a k-means, the color is associated with the dots under each keypoin in (a); (c) Keypoint description is based on the time series filtered at the scale at which the keypoint is extracted, descriptors are quantized into words; (d) Histograms of word occurrences that can feed a classifier.

Pros and Cons of the method. D-BoTSW has many advantages such as its robustness to noise and its ability to handle temporal shifts. Nevertheless, the BoW representation causes information loss during the quantization step. It also ignores the temporal order of words. When the ordering of event matters, one can couple the D-BoTSW features to a dedicated classification algorithm such as [Tavenard 2017] (see section 5.1). The most computationally demanding step of D-BoTSW is the k-means learning process and the fitting of the classifier. The dense extraction, features description and transformation to BoW representation can be done efficiently: D-BoTSW is thus able to fastly classify new data.

4.2.2 Experiments on a remote sensing scenario

On the UCR archive. We compare the performances of D-BoTSW with the ones of several state-of-the-art time series classifiers on the datasets of the UCR archive. We show that it leads to better or same performances than all the competitors but one (which is an ensemble-based classifier). Figure 4.5 shows the performances of the method compared to a 1NN-DTW classifier.

On the Brazilian Amazon dataset. We extract two years of MODIS vegetation index time series corresponding to the cropping periods 2005-2006 and 2006-2007. The field data used for validation is the same as in [Arvor 2011]. The dataset is made of 46 MODIS images from July 2005 to July 2007, i.e. 23 MODIS images per year. The study area is located in the state of Mato Grosso, in the southern Brazilian amazon, that has suffered dramatic land use changes since the 1970s due to the rapid progress of an agricultural frontier. The vegetation index time series refers to Enhanced Vegetation Index (EVI) which

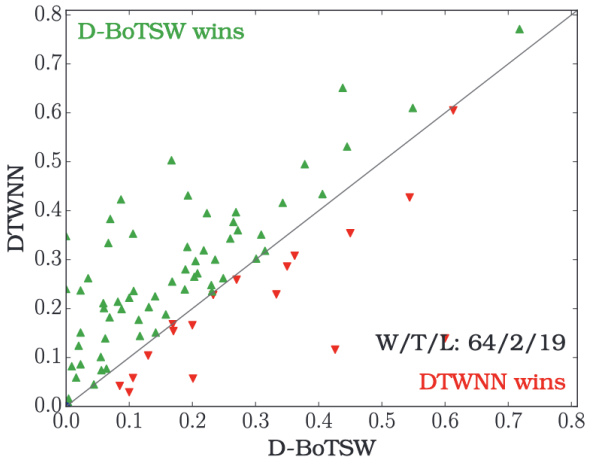


Figure 4.5: Error rates for D-BoTSW versus a 1NN-DTW classifier. Green upper triangle indicates a dataset for which D-BoTSW has a lower error rate than 1NN-DTW while a Green lower triangle indicates the opposite. Among all the tested datasets, 64 out of 85 datasets have a lower error rate for D-BoTSW.

is derived from the Normalized Difference Vegetation Index (NDVI). Five crop classes were identified, two of them referring to single cropping practices whereas the 3 remaining ones refer to double cropping practices. The aim is then to determine the crop practice with the evolution of the EVI. Figure 4.6 gives the histograms that have been extracted with D-BoTSW. When it comes to classification accuracies, D-BoTSW has the best performances when compared to standalone shallow classification methods. Detailed results are presented in [Bailly 2016, Bailly 2018]

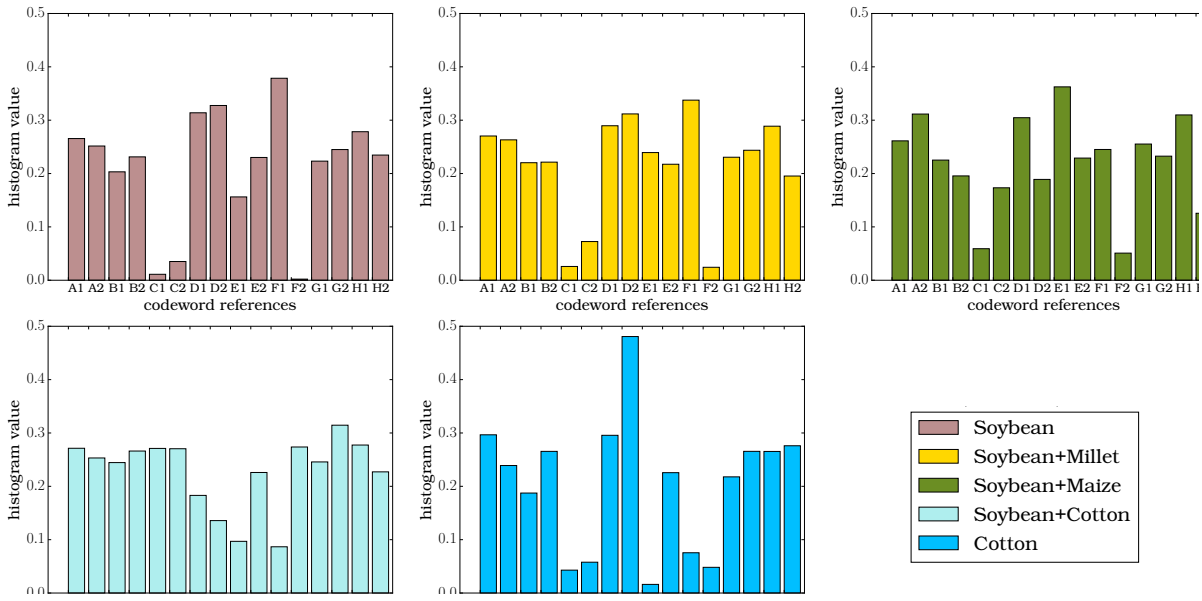


Figure 4.6: Average histogram per class.

Chapter 5

Bulding Sensible Metrics for Time Series

Contents

In the previous section, we have considered methods that first look for an appropriate embedding before applying a machine learning technique on it. In this section, we will cover works that aim at instead defining new metrics for time series, avoiding this embedding step. The first work [Tavenard 2017] defines a new kernel for time series that takes into account the temporal dimension when comparing time series. The second one [Vayer 2020b] builds on the DTW metric to compare time series that live in incomparable spaces or that have to be realigned. It relies on a global latent representation of the feature space in which the time series can be compared.

5.1 A temporal kernel for time series

In the time-series classification context, the majority of the most accurate core methods are based on the Bag-of-Words framework, in which sets of local features are first extracted from time series. A dictionary of words is then learned and each time series is finally represented by a histogram of word occurrences (see section 4.2 for a temporal SIFT-feature example). This representation induces a loss of information due to the quantization of features into words as all the time series are represented using the same fixed dictionary. In order to overcome this issue, we define a novel temporal kernel that takes as input a set of feature vectors extracted from the time series and their timestamps, allowing taking into account the feature localization.

5.1.1 A temporal kernel between sets of features

Match kernel and Signature Quadratic Form Distance. Our kernel relies on the match kernel [Bo 2009] (a.k.a. set kernel) that takes as input unordered sets of features:

$$K(\boldsymbol{v}, \boldsymbol{w}) = \sum_{i=1}^{n} \sum_{j=1}^{m} k_{\text{RBF}}(v_i, w_j) \tag{5.1}$$

and the Signature Quadratic Form Distance (SQFD) that enables the comparison of time series $\boldsymbol{x}$ and $\boldsymbol{y}$ represented by weighted sets of features $\boldsymbol{v}$ and $\boldsymbol{w}$ called feature signatures (assuming here that all features have the same weight $\frac{1}{n}$ and $\frac{1}{m}$):

$$\text{SQFD}(\boldsymbol{v}, \boldsymbol{w})^2 = \frac{1}{n^2} K(\boldsymbol{v}, \boldsymbol{v}) + \frac{1}{m^2} K(\boldsymbol{w}, \boldsymbol{w}) - \frac{2}{n \cdot m} K(\boldsymbol{v}, \boldsymbol{w}) \tag{5.2}$$

where k_{RBF} is the Gaussian kernel. Note that SQFD then corresponds to a biased estimator of the squared difference between the mean of the samples $\boldsymbol{v}$ and $\boldsymbol{w}$ and is classically used to test the difference between two distributions [Gretton 2012]. Finally, in order to kernelize SQFD, we can embed it into a RBF kernel:

$$K_{\text{SQFD}}(\boldsymbol{v}, \boldsymbol{w}) = e^{-\gamma_f \, \text{SQFD}(\boldsymbol{v}, \boldsymbol{w})^2}. \tag{5.3}$$

Time-sensitive feature set kernel. Kernel (5.3) ignores the temporal location of the features in the time series. To integrate this information, we propose to augment the features with the time index at which they are extracted. By defining $g(v_i, t_i) = \left(v_{i1}, \ldots, v_{id}, \sqrt{\frac{\gamma_t}{\gamma_f}}\, t_i \right)$, with γ_f a scale parameter and γ_t a temporal parameter, we obtained a time-sensitive kernel

$$k_{\text{tRBF}}((v_i, t_i), (w_j, t_j)) = e^{-\gamma_t (t_j - t_i)^2} \cdot k_{\text{RBF}}(v_i, w_j) = e^{-\gamma_f \, \| g(v_i, t_i) - g(w_j, t_j) \|^2}. \tag{5.4}$$

Efficient computation of the time-sensitive kernel. Kernel (5.4) is an RBF kernel itself, and Random Fourier Features [Rahimi 2007] can be used in order to approximate it with a linear kernel:

$$\begin{aligned}
\frac{1}{n \cdot m} K(\boldsymbol{v}, \boldsymbol{w}) &= \frac{1}{n \cdot m} \sum_{i=1}^{n} \sum_{j=1}^{m} k_{\text{RBF}}(v_i, w_j) && (5.5) \\
&\approx \frac{1}{n \cdot m} \sum_{i=1}^{n} \sum_{j=1}^{m} \langle \phi_{\text{RBF}}(v_i), \phi_{\text{RBF}}(w_j) \rangle && (5.6) \\
&\approx \left\langle \underbrace{\frac{1}{n} \sum_{i=1}^{n} \phi_{\text{RBF}}(v_i)}_{\phi(\boldsymbol{v})}, \underbrace{\frac{1}{m} \sum_{j=1}^{m} \phi_{\text{RBF}}(w_j)}_{\phi(\boldsymbol{w})} \right\rangle && (5.7)
\end{aligned}$$

The SQFD kernel then becomes $\text{SQFD}(\boldsymbol{v}, \boldsymbol{w})^2 = \| \phi(\boldsymbol{v}) - \phi(\boldsymbol{w}) \|^2$.

5.1.2 Experiments and conclusion

Impact of the temporal parameter. Figure 5.1 illustrates the impact of the temporal parameter γ_f: a low value takes all matches into account without considering their temporal locations, whereas large values favors diagonal matches. Regarding the effectiveness of the algorithm, we show that i) at training time, our kernel is faster than the original BoW approach; ii) at testing time, BoW and our kernel have almost constant computation needs, BoW being one order of magnitude faster.

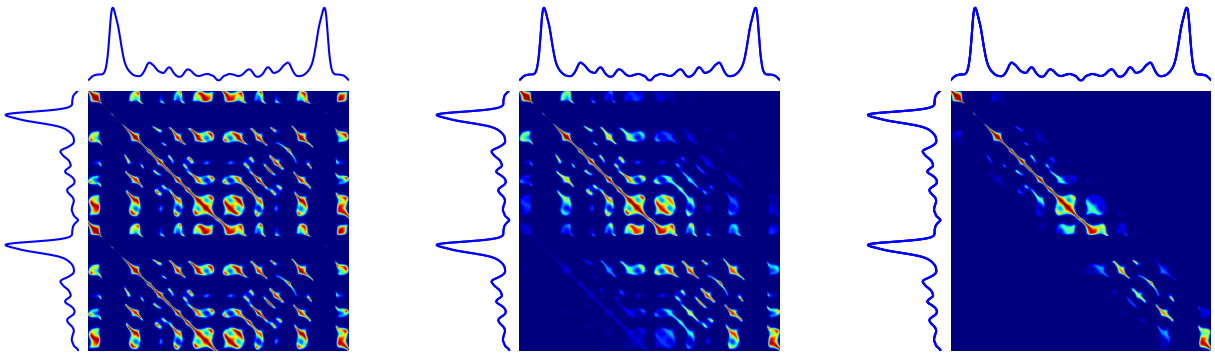


Figure 5.1: Impact of the k_{tRBF} kernel on similarity matrices. From left to right, growing γ_t values are used from $\gamma_t = 0$ to a large γ_t value that ignores almost all non-diagonal matches. Blue color indicate low similarity whereas red color represents high similarity.

On the UCR archive. We compare our kernel with i) the kernel with no temporal information ii) the original BoW approach. By performing a Wilcoxon rank test, we conclude that (see figure 5.2), on the 85 datasets of the UCR archive, our temporal kernel gives significantly better performances than its competitors.

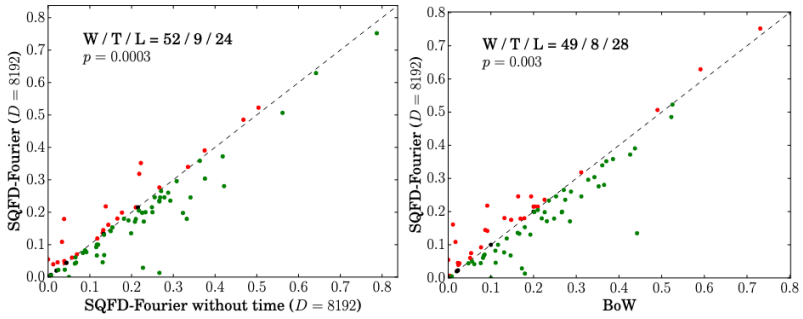


Figure 5.2: Pairwise performance comparisons. A green point indicates a dataset of the UCR archive for which our temporal kernel has a lower error rate than (left) the non-temporal version of the kernel (right) the original BoW method (no kernel). Among the datasets, the temporal kernel has a better performance on 52 datasets out of 85 than the non-temporal version, and on 49 out of 85 than the original BoW method.

5.2 Dynamic Time Warping with Global Invariances

We now focus on designing a metric that is able to compare time series of possibly different dimensions and that exhibit some invariances.

5.2.1 Definition

We propose here to solve the following joint optimization problem:

$$\text{DTW-GI}(\boldsymbol{x}, \boldsymbol{y}) = \min_{f \in \mathcal{F}, \pi \in \Pi(\boldsymbol{x}, \boldsymbol{y})} \sum_{(i,j) \in \pi} d(\boldsymbol{x}_i, f(\boldsymbol{y})_j)) = \min_{f \in \mathcal{F}, \pi \in \Pi(\boldsymbol{x}, \boldsymbol{y})} \langle \pi, C(\boldsymbol{x}, f(\boldsymbol{y})) \rangle \tag{5.8}$$

where $\mathcal{F}$ is a family of functions from $\mathbb{R}^p$ to $\mathbb{R}^r$, $f(\boldsymbol{y})$ is a shortcut notation for the transformation f applied to all observations in $\boldsymbol{y}$.

The $C(\mathbf{x}, f(\mathbf{y}))$ matrix is the cross-similarity matrix of squared Euclidean distances between samples from $\boldsymbol{x}$ and $f(\boldsymbol{y})$, respectively. These similarity measures estimate both temporal alignment and feature space transformation between time series simultaneously, allowing the alignment of time series when the similarity should be defined up to a global transformation. What should be noted in this formulation is that the dimension p and r do not have to be the same, as illustrated in figure 5.3. It is also straightforward to see that DTW-GI$(\boldsymbol{x}, \boldsymbol{x}) = 0$ for any $\boldsymbol{x}$ as soon as $\mathcal{F}$ contains the identity map.

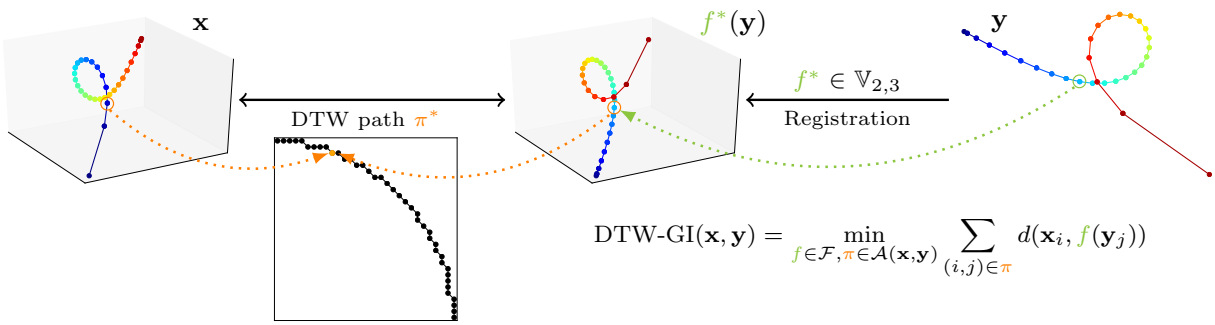


Figure 5.3: DTW-GI aligns time series by optimizing jointly on both temporal alignment (through Dynamic Time Warping) and feature space transformation (denoted f here). Time series represented here are color-coded trajectories, whose starting (resp. end) point is depicted in blue (resp. red).

This definition can be extended to the softDTW case:

$$\text{DTW}_\gamma\text{-GI}(\mathbf{x}, \mathbf{y}) = \min_{f \in \mathcal{F}} \min_{\pi \in \Pi(\mathbf{x},\mathbf{y})}{}^{\gamma} \langle \pi, C(\mathbf{x}, f(\mathbf{y}))\rangle = \min_{f \in \mathcal{F}} -\gamma \log \sum_{\pi \in \Pi(\mathbf{x},\mathbf{y})} e^{-\langle \pi, C(\mathbf{x}, f(\mathbf{y}))\rangle/\gamma} \tag{5.9}$$

5.2.2 Optimization

Depending of the nature of $\mathcal{F}$, optimization on the above-defined losses can be performed in several ways. We now present one optimization scheme for each loss.

Solving DTW-GI. When DTW-GI is concerned, by considering that f belong to the Stieffel manifold, we use a Block- Coordinate Descent (BCD) strategy that alternates over two steps:

- for a fixed f, minimizing the temporal alignments thanks to the DTW algorithm.
- for a fixed matrix π, the set of feature space transformations f is obtained by solving a singular value problem (SVD) on the matrix $\boldsymbol{x}^\top \pi \boldsymbol{y}$.

Figure 5.4 illustrates the ability of our method to recover invariance to rotation. To do so, we rely on a synthetic dataset of noisy spiral-like 2d trajectories. For increasing values of an angle θ, we generate pairs of spirals rotated by θ with additive gaussian noise. Alignments between a reference time series and variants that are subject to an increasing rotation θ are computed and repeated 50 times per angle. The ratio of each distance to the distance when $\theta = 0$ is reported on the figure. One can clearly see that the GI counterparts of DTW and softDTW are invariant to rotation in the 2d feature space, while DTW and softDTW are not.

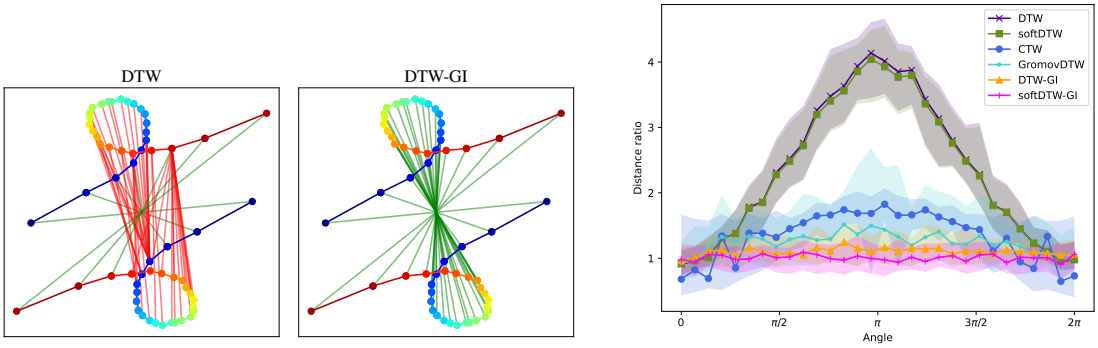


Figure 5.4: (Left) DTW-GI jointly estimates temporal alignment and global rotation between time series. On the contrary, standard DTW alignment fails at capturing feature space distortions and therefore produces mostly erroneous alignment (matching in red), except at the beginning and end of the time series, whose alignments are preserved thanks to DTW border constraints. (Right) CTW and GromovDTW, that should be invariant to rotation, still exhibit an increase in the loss with the angle, suggesting that their algorithm has more difficulties reaching a global minimum in practice.

Solving soft-DTW-GI. When $\mathcal{F}$ is a parametric family of functions that are differentiable with respect to their parameters, problem (5.9) can be solved with a gradient descent on the parameters of f. Since softDTW is smooth, this strategy can be used to compute gradients of DTW$_\gamma$-GI w.r.t. the parameter θ of $f(\theta)$.

Solving DTW-GI barycenters Let us now assume we are given a set $\{\mathbf{x}^{(i)}\}_i$ of time series of possibly different lengths and dimensionalities. A barycenter of this set in the DTW-GI sense is a solution to the following optimization problem:

$$\min_{\mathbf{b}\in\mathbb{R}^{T\times q}} \sum_i w_i \min_{f_i\in\mathcal{F}} DTW(\mathbf{x}^{(i)}, f_i(\mathbf{b})), \tag{5.10}$$

where weights $\{w_i\}_i$ as well as barycenter length T and dimensionality q are provided as input to the problem. For solving the problem, in the same way as DTW-GI, either a gradient descent can be performed, or a BCD procedure can be derived, alternating between barycentric coordinate estimation and DTW-GI alignments. Figure 5.5 provides some barycenters in different settings. Temporal alignments in (soft)DTW-GI successfully capture the irregular sampling from the samples to be averaged (denser towards the center of the spiral / loop of the folium) and all the barycenters can be considered as meaningful. On the contrary, baseline barycenters can fail (second line) or cannot be applied when datasets are made of series that do not lie in the same space.

5.2.3 Experiments

We tested DTW-GI in different settings. First, in a time series forecasting scenario, we provide qualitative and quantitative results in a dataset composed of 3.6 million video frames of 3 dimensional human movements. Quantitative results show that (soft)DTW-GI allows realigning the different poses while taking into account the temporal variability of the movement, while competitive methods can not take into account an explicit spatial transformation. We also consider a cover song identification, which is the task of retrieving, for a given query song, its covers (that is to say different versions of the same song) in a training set. Again, DTW-GI gives better results than it competitors, thanks to the versatility of

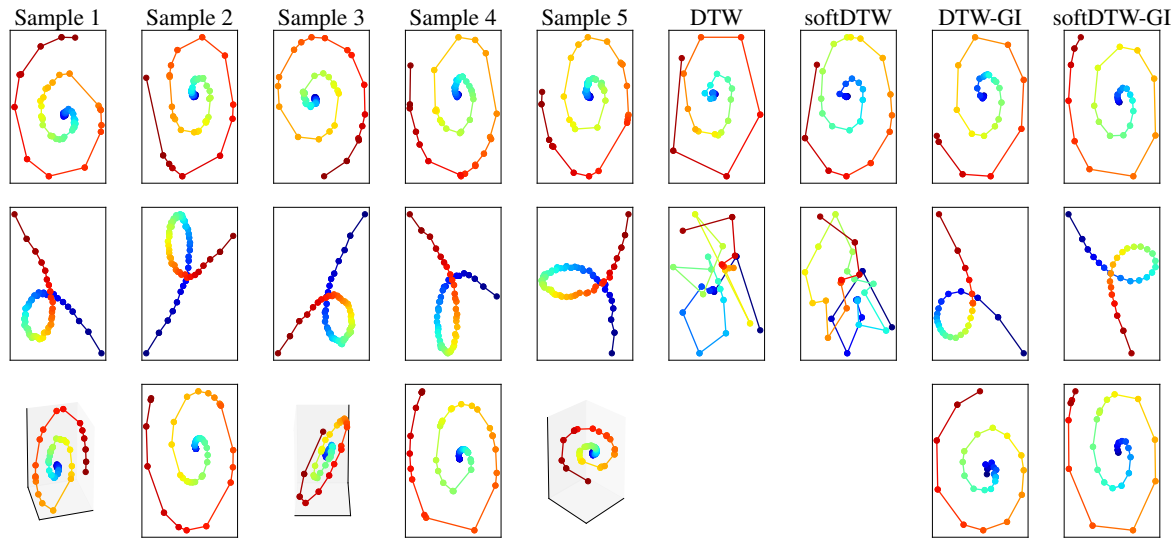


Figure 5.5: Barycenter computation using (i) DTW and softDTW baseline approaches, (ii) their rotation-invariant counterparts DTW-GI and softDTW-GI. Time series represented here are color-coded trajectories, whose starting (resp. end) point is depicted in blue (resp. red).

the method that allows introducing prior information about the feature space, with the definition of the space $\mathcal{F}$.

Part III

Contributions to Optimal Transport for Machine Learning with Applications on Graphs

Chapter 6

Discrete Optimal Transport for Machine Learning

Contents

In this chapter, I briefly introduce the main ingredients of discrete Optimal Transport (OT) that will be useful for the next chapters; there exist excellent books that describe widely the OT theory. The reader can refer to [Villani 2009] for a mathematical oriented overview of OT. A more accessible introduction for applied mathematician with a rigorous description of the theory of OT can be found in [Santambrogio 2015]. For the computational aspects, the reader can refer to the complete book of [Peyré 2019]. In this section, I will also review the Gromov-Wasserstein problem, for which a complete description can be found in [Mémoli 2011].

I only focus on discrete measures and will consider clouds of points $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^n$ and $\mathcal{Y} = \{\boldsymbol{y}_j\}_{j=1}^m$. Let $\boldsymbol{h} \in \mathbb{R}_n^+$ and $\boldsymbol{g} \in \mathbb{R}_m^+$ be two discrete distributions of mass on $\mathcal{X}$ and $\mathcal{Y}$, such that h_i (resp. g_j) is the mass at $\boldsymbol{x}_i$ (resp. $\boldsymbol{y}_j$). We set $\boldsymbol{h} \in \Sigma_n$ and $\boldsymbol{g} \in \Sigma_m$ where $\Sigma_n = \{(a_i)_i \geq 0, \sum_i a_i = 1\}$ is the probability simplex. A discrete measure μ_X (resp. μ_Y) with weights $\boldsymbol{h}$ (resp. $\boldsymbol{g}$) and locations $\mathcal{X}$ (resp. $\mathcal{Y}$) is given by:

$$\mu_X = \sum_{i=1}^{n} h_i \delta_{x_i} \tag{6.1}$$

with δ_{x_i} the Dirac at position $\boldsymbol{x}_i$. Note that, in the machine learning context, weights are often set as uniform ($h_i = \frac{1}{n}$, $g_j = \frac{1}{m}$).

Notations. $\mathbb{1}_n$ is a vector of n entries of ones and $\mathbb{I}_n$ is the identity matrix of size n. $\otimes$ is the kronecker product and $\langle \cdot, \cdot \rangle$ is the Frobenius inner product. D_ϕ is the Bregman divergence generated by the strictly convex and differentiable function ϕ, i.e., $D_\phi(\mathbf{u}, \mathbf{v}) = \sum_i [\phi(u_i) - \phi(v_i) - \phi'(v_i)(u_i - v_i)]$.

6.1 From Monge and Kantorovich formulations to the (Gromov-) Wasserstein distance

The OT problem was introduced by Gaspard Monge [Monge 1781] in a Mémoire on which he aims at moving "optimally" dirt from one place to another, i.e. by minimizing the overall cost of transportation of all the masses. Almost two centuries later, Kantorovich [Kantorovich 1942] proposed a relaxation of the problem in order to allocate optimally economical resources.

Monge problem. The Monge problem looks for a map $M : \{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\} \to \{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_m\}$ that pushes the mass μ_X toward μ_Y:

$$\forall 1 \leq j \leq m, \quad g_j = \sum_{i \text{ such that (s.t.) } M(\boldsymbol{x}_i)=\boldsymbol{y}_j} h_i. \tag{6.2}$$

Note here that each point $\boldsymbol{x}_i$ must be mapped to at most *one* point $\boldsymbol{y}_j$, the reverse being false. By associating a cost $d(\boldsymbol{x}_i, \boldsymbol{y}_j)$ of mapping $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$, the Monge map is the one such that:

$$\min_M \left\{ \sum_i d(\boldsymbol{x}_i, M(\boldsymbol{x}_i)) \text{ s.t. eq. (6.2) is satisfied} \right\}. \tag{6.3}$$

When $n = m$ and $h_i = g_j = \frac{1}{n}$, it comes down to an optimal assignment problem:

$$\min_{\sigma \in S_n} \sum_i d(\boldsymbol{x}_i, \boldsymbol{y}_{\sigma(i)}) \tag{6.4}$$

where $\sigma \in S_n$ is a one-to-one mapping $\{1, \cdots, n\} \to \{1, \cdots, n\}$.

Kantorovich problem. The Monge problem may not have a solution (just think of a simple case where $n = 1$ and $m = 2$, there is no map that allows mapping point $\boldsymbol{x}_1$ to only one point $\boldsymbol{y}_j$ while conserving the mass); Kantorovitch proposed to relax the problem by allowing the mass of a point $\boldsymbol{x}_i$ to be split across several locations. As such, it produces a *coupling matrix* (rather than a map) that describes how much mass from $\boldsymbol{x}_i$ should be sent to $\boldsymbol{y}_j$. The *balanced* OT problem, as defined by [Kantorovich 1942], is a linear problem that computes the minimum cost of moving $\boldsymbol{h}$ to $\boldsymbol{g}$:

$$\text{OT}(\boldsymbol{h}, \boldsymbol{g}) = \min_{\boldsymbol{T} \geq 0} \langle \boldsymbol{C}, \boldsymbol{T} \rangle = \sum_{i,j} C_{i,j} T_{i,j} \quad \text{s.t.} \quad \boldsymbol{T} \mathbb{1}_m = \boldsymbol{h}, \boldsymbol{T}^\top \mathbb{1}_n = \boldsymbol{g} \tag{6.5}$$

where $\boldsymbol{T} \in \mathbb{R}_+^{n \times m}$ is the *coupling* or *transport matrix* and $\boldsymbol{C} \in \mathbb{R}_+^{n \times m}$ is the *cost matrix*. The entry $C_{i,j} = (d(\boldsymbol{x}_i, \boldsymbol{y}_j))_{i,j}$ of $\boldsymbol{C}$ represents the cost of moving point $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$, where d is the ground cost (see figure 6.1 for an illustration).

The constraints on the transport matrix $\boldsymbol{T}$ require that $\|\boldsymbol{h}\|_1 = \|\boldsymbol{g}\|_1$ and that *all* the mass from $\boldsymbol{h}$ is transported to $\boldsymbol{g}$. It must then belong to the following set of constraints:

$$\boldsymbol{\Pi}(\boldsymbol{h}, \boldsymbol{g}) = \left\{ \boldsymbol{T} \in \mathbb{R}_+^{n \times m} | \boldsymbol{T} \mathbb{1}_m = \boldsymbol{h}, \boldsymbol{T}^\top \mathbb{1}_n = \boldsymbol{g} \right\}. \tag{6.6}$$

Wasserstein distance. When d is a distance, one can defined the q-Wasserstein distance at the power of q:

$$W_q^q(\boldsymbol{h}, \boldsymbol{g}) = \min_{\boldsymbol{T} \in \boldsymbol{\Pi}(\boldsymbol{h}, \boldsymbol{g})} \langle \boldsymbol{C}^q, \boldsymbol{T} \rangle = \min_{\boldsymbol{T} \in \boldsymbol{\Pi}(\boldsymbol{h}, \boldsymbol{g})} \sum_{i,j} C_{i,j}^q T_{i,j}. \tag{6.7}$$

The 1-Wasserstein distance (also known as the Earth mover's distance [Rubner 2000]) is obtained for $C_{i,j} = \|\boldsymbol{x}_i - \boldsymbol{y}_j\|$.

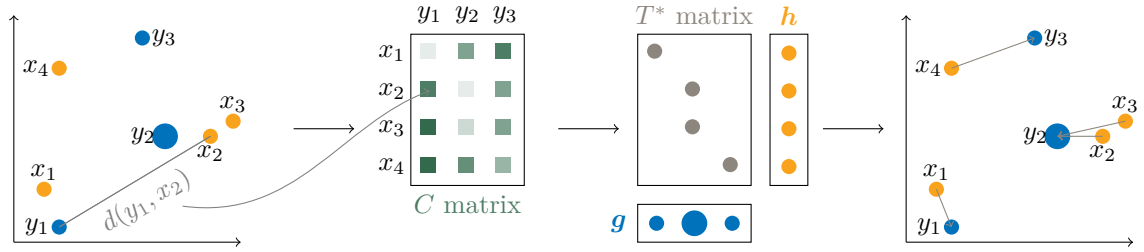


Figure 6.1: The Monge-Kantorovich problem. From two sets of points $\mathcal{X}$ and $\mathcal{Y}$ with the associated distributions $\boldsymbol{h}$ and $\boldsymbol{g}$ (depicted here by the size of the points) and a ground cost d, one look the optimal coupling matrix $\boldsymbol{T}^*$ (which corresponds to the optimal Monge map in this example) that minimizes the overall transport cost, by transporting *all* the mass from one distribution to another. This matrix produces a map that associates points $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$.

Gromov-Wassersein distance. Despite some appealing properties, the Kantorovich or Wasserstein distance fail at comparing point clouds that live in *incomparable* spaces, that is to say when they are not part of a common metric space (no function d can be defined to compare them, think of points with different dimensions or that contains different features for example). Gromov-Wasserstein aims at computing a distance in this case, also allowing interesting properties such as rotational and translational invariance. Let us note d_X (resp. d_Y) a ground cost between two samples of $\mathcal{X}$ (resp. $\mathcal{Y}$) and denote $\boldsymbol{C}_X = (d_X(\boldsymbol{x}_i, \boldsymbol{x}_k))_{i,k}$ (resp. $\boldsymbol{C}_Y = (d_Y(\boldsymbol{y}_j, \boldsymbol{y}_l))_{j,l}$). The Gromov-Wasserstein at the power of q is defined as:

$$GW_q^q(\boldsymbol{C}_X, \boldsymbol{C}_Y, \boldsymbol{h}, \boldsymbol{g}) = \min_{\boldsymbol{T} \in \boldsymbol{\Pi}(\boldsymbol{h}, \boldsymbol{g})} \langle L(\boldsymbol{C}_X, \boldsymbol{C}_Y)^q \otimes \boldsymbol{T}, \boldsymbol{T} \rangle = \min_{\boldsymbol{T} \in \boldsymbol{\Pi}(\boldsymbol{h}, \boldsymbol{g})} \sum_{i,j,k,l} L(d_X(\boldsymbol{x}_i, \boldsymbol{x}_k), d_Y(\boldsymbol{y}_j, \boldsymbol{y}_l))^q T_{i,j} T_{k,l} \tag{6.8}$$

in which $L(\boldsymbol{C}_X, \boldsymbol{C}_Y)$ is a loss function, e.g. $L(a, b) = |a - b|^2/2$ (see figure 6.2 for an illustration).

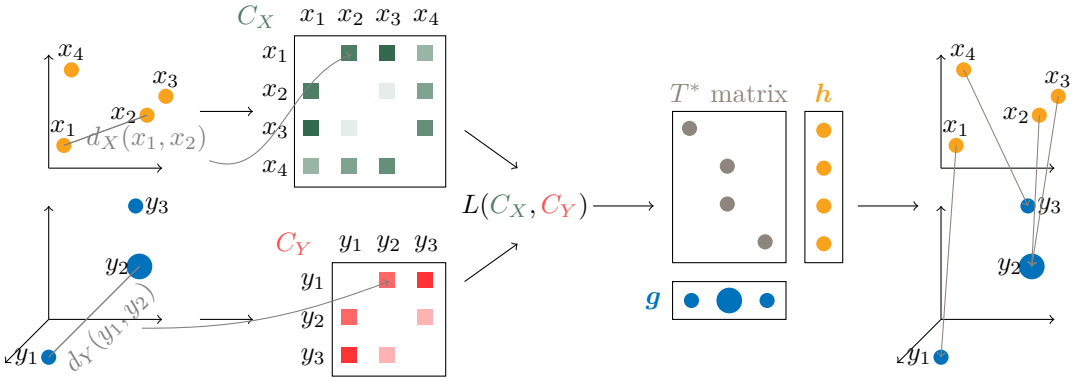


Figure 6.2: The Gromov-Wasserstein distance. From two sets of points $\mathcal{X} \subset \mathbb{R}^2$ and $\mathcal{Y} \subset \mathbb{R}^3$ with the associated distributions $\boldsymbol{h}$ and $\boldsymbol{g}$ (depicted here by the size of the points) and two ground costs d_X and d_Y, one look for the optimal coupling matrix $\boldsymbol{T}^*$ that minimizes the overall transport cost between intra-domain distances, by transporting *all* the mass from one distribution to another. In this example, this matrix produces a map that associates points $\boldsymbol{x}_i$ to $\boldsymbol{y}_j$, even when they leave into incomparable spaces.

6.2 Relaxed and regularized optimal transport

So far, the OT distances require the transport matrix $\boldsymbol{T}$ belong to the set of constraints (6.6), that is to say that $\|\boldsymbol{h}\|_1 = \|\boldsymbol{g}\|_1$ and that *all* the mass from $\boldsymbol{h}$ is transported to $\boldsymbol{g}$. It leads to a sparse matrix $\boldsymbol{T}^*$ which has at most $n+m-1$ values that are non zeros (see section 6.3). Nevertheless, in many or maybe most machine learning applications, these features may be undesirable. Let us describe 3 specific issues:

- one may rather work with unnormalized histograms as the mass may be an important feature of the problem, e.g. when dealing with neuroimaging data [Gramfort 2015] or performing color transfer between images [Rabin 2014], in which the source and target objects of interest have different masses;
- the data can contain some outliers that may drive OT to large and non-significant values (see figure 6.3);
- one may rather build on more that $n+m-1$ coupling values, allowing one to lower the weights of noisy observations.

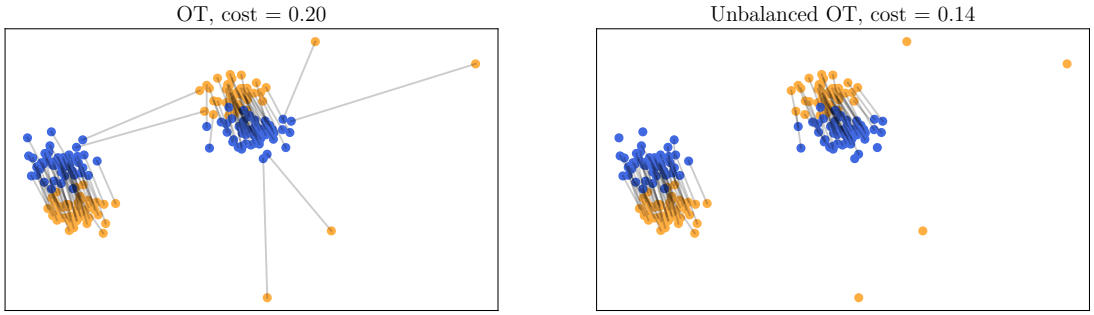


Figure 6.3: Few outliers can disrupt the transport plan and drive the OT distance to unexpected large values. (Left) the OT transport matrix is depicted with grey lines (Right) the partial OT solution, that allows excluding the outliers from the solution.

To overcome those issues, several relaxed and regularized versions have been proposed, whose interest depends on the applicative point of view. [Kantorovich 1957] first extended the OT distance to the unbalanced setting by introducing the waste function, allowing for cases in which it is "cheaper" to waste mass rather than transporting it [Guittet 2002]. Despite this first work, it is only in the last ten years that solid theories have been developed. Here, we describe some relaxations of the OT problem, first describing problems in which the mass conservation constraint is relaxed, then considering problems in which regularity conditions of the mapping is sought. We will refer to as *balanced* OT the unconstrained problem.

6.2.1 Unbalanced and partial optimal transport

Unbalanced optimal transport (UOT) [Benamou 2003] allows some mass variation in the transportation problem. It solves the Kantorovich problem but considers a different set of constraints: rather than imposing a strict conservation of the mass, the deviations from the true marginals are penalized by means of a given Bregman divergence D_φ (as introduced in [Chizat 2018a]), where γ_1 and γ_2 are hyperparameters that represent how much the deviation is penalized:

$$\boldsymbol{\Pi}^u(\boldsymbol{h},\boldsymbol{g}) = \left\{ \boldsymbol{T} \in \mathbb{R}_+^{n\times m} \mid D_\varphi(\boldsymbol{T}\mathbb{1}_m, \boldsymbol{h}) \leq \gamma_1,\ D_\varphi(\boldsymbol{T}^\top \mathbb{1}_n, \boldsymbol{g}) \leq \gamma_2 \right\}. \tag{6.9}$$

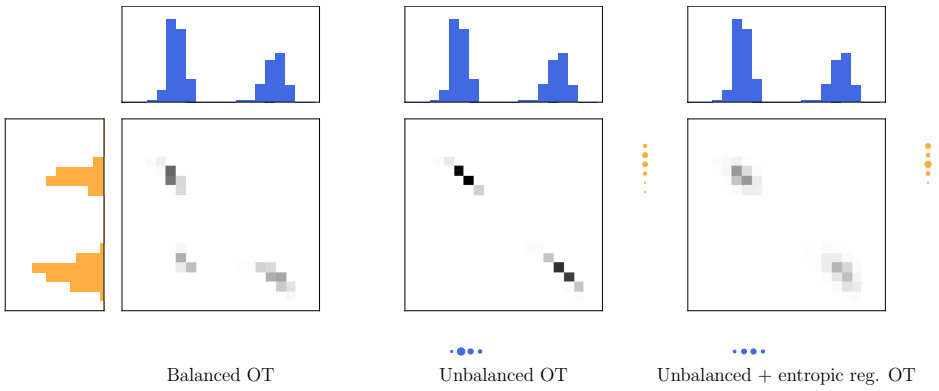


Figure 6.4: Transport matrices of the balanced OT (left), unbalanced OT (center) and UOT with an entropic regularization (right) between two empirical histograms. The orange and blue dots illustrate the amount of mass that is missing for each bin of the histograms (no dots indicate that all the mass has been transported). UOT allows accounting for a difference of mass between the 2 main modes of the histograms, avoiding "noisy" transportation. Adding the entropic regularization leads to a spreading of the mass (inspired from a presentation of Lenaïc Chizat).

Note that balanced OT is recovered when $\gamma_1 = \gamma_2 \to 0$. Furthermore, when γ_1 *or* $\gamma_2 \to 0$, we recover the semi-relaxed OT [Rabin 2014]. In practice, authors often set $\gamma_1 = \gamma_2 = \gamma$ for UOT in order to reduce the necessity of hyperparameter tuning. Figure 6.4 illustrates the advantages of the UOT formulation. Considering two empirical histograms with 2 modes with different mass, the balanced OT solution provides an extra mapping between two distinct modes of the distributions; when relaxing the constraint, one allows mass variation, avoiding unnecessary coupling: the bins of the 2 modes are then correctly associated.

Some problems require to quantify precisely the amount of transportation of mass $s \leq \min(\|\boldsymbol{h}\|_1, \|\boldsymbol{g}\|_1)$ needed to compare the distributions (e.g. when the proportion of outliers is known). In that case, we rather consider the following set of constraints:

$$\boldsymbol{\Pi}^p(\boldsymbol{h}, \boldsymbol{g}) = \left\{ \boldsymbol{T} \in \mathbb{R}_+^{n \times m} \mid \boldsymbol{T}\mathbb{1}_m = \boldsymbol{h},\ \boldsymbol{T}^\top \mathbb{1}_n = \boldsymbol{g},\ \mathbb{1}_n^\top \boldsymbol{T} \mathbb{1}_m = s \right\}. \tag{6.10}$$

This problem is coined the *partial* optimal transport problem [Caffarelli 2010, Figalli 2010].

Unbalanced and partial OT can be both rewritten as, using a Lagrangian formulation:

$$\mathrm{UOT}^{\boldsymbol{\lambda}}(\boldsymbol{h}, \boldsymbol{g}) = \langle \boldsymbol{C}, \boldsymbol{T}^* \rangle \quad \text{where} \quad \boldsymbol{T}^* = \min_{\boldsymbol{T} \geq 0} \ \langle \boldsymbol{C}, \boldsymbol{T} \rangle + \lambda_1 D_\varphi(\boldsymbol{T}\mathbb{1}_m, \boldsymbol{h}) + \lambda_2 D_\varphi(\boldsymbol{T}^\top \mathbb{1}_n, \boldsymbol{g}) \tag{6.11}$$

where λ_1 and λ_2 are hyperparameters that represent the strengths of penalization. Various divergences have been considered in the literature. The ℓ_1 norm gives rise to the partial optimal transport; the squared ℓ_2 norm provides a sparse and smooth transport plan [Blondel 2018] when introducing a strongly convex term in eq. (6.11); the Kullback-Leibler divergence has been widely considered in the literature [Chizat 2018b].

It has been shown in [Chizat 2018b] that partial optimal transport is an instance of the unbalanced OT when considering the ℓ_1 norm as a divergence.

6.2.2 Entropic-regularized optimal transport

The idea of adding an entropic regularization to the OT problem has been first introduced by [Cuturi 2013]. It penalizes the deviation of the transport matrix from independence, requiring the transport

matrix to belong to the following set of constraints:

$$\mathbf{\Pi}^e(\boldsymbol{h}, \boldsymbol{g}) = \left\{ \boldsymbol{T} \in \mathbb{R}_+^{n \times m} \mid h(\boldsymbol{h}\boldsymbol{g}^\top) - h(\boldsymbol{T}) \leq \alpha \right\} \tag{6.12}$$

in which h is the entropy of the matrices, e.g. $h(\boldsymbol{T}) = -\sum_{i,j} T_{ij} \log T_{ij}$, with the convention that $0 \log(0) = 0$. By setting $\alpha \to \infty$, it is easy to see that it comes down to solving an original OT problem; when $\alpha \to 0$, it can be shown that the optimal transport matrix tends to the independance kernel [Cuturi 2013]. Considering again a Lagrangian formulation, it can be equivalently rewritten as

$$\mathrm{OT}^\varepsilon(\boldsymbol{h}, \boldsymbol{g}) = \min_{\boldsymbol{T} \in \mathbf{\Pi}(\boldsymbol{h}, \boldsymbol{g})} \quad \langle \boldsymbol{C}, \boldsymbol{T} \rangle + \varepsilon h(P), \tag{6.13}$$

with ε a parameter that controls the strength of the regularization. This formulation has several virtues. First, since the entropy is a strongly convex function, problem (6.13) has a unique optimal solution. Intuitively, the entropic regularization pushes the solution away from the boundary of its convex polytope. With the increase of ε, the optimal solution moves progressively into the interior of the polytope, causing a loss of sparsity in P. This loss of sparsity can be desirable when the data is polluted with noise or outliers: it enables to smooth the transport plan, and hence spreads the impact of those data. Another advantage of this formulation is that it can be solved by iterating simple matrix products (see section 6.3), allowing the "lightspeed" computation of the solution. Those matrix products can be vectorized and are then amenable to GPU computations. Last but not least, it is differentiable [Luise 2018], which enables its use as a loss function.

6.2.3 Other regularizations and relaxations

Several other regularizations have been introduced in the literature, depending on the problem at hand. Among them, one can cite the capacity constrained optimal transport [Korman 2015] that puts an upper bound on each of the coupling weights. To deal with image processing tasks, and in order to take into account families of multimodal histograms, [Ferradans 2014] relaxed the 1:1 hypothesis (one point from the source associated to only one point of the target when $n = m$ and the weights are uniforms) by allowing each point of the source to be transported to multiple points of the target and vice versa. They also proposed a regularization to ensure the regularity of the solution, smoothing the optimal transport plan by controlling the displacement of pairs of points. In the same line of research, [Flamary 2014] propose a Laplacian regularization to promote the respect of the proximities observed in the original distribution after the transport; it has been successfully applied in domain adaptation problems. For computational efficiency issues, [Thibault 2021] propose to overrelax the Bregman projection operators within the Sinkhorn algorithm, allowing for faster convergence.

6.3 Numerical resolution of discrete optimal transport

Solving the OT problem is computationally intensive. In this section, we first review the particular case in which the distributions are 1-dimensional, then the main algorithms that solve problem (6.5) and finally review briefly the Sinkhorn algorithm that solves entropic OT (6.13).

1-dimensional distributions. Let us suppose that the data are one-dimensional, that $n = m$, $h_i = g_j = \frac{1}{n}$, and that we sort the points $\boldsymbol{x}_1 \leq \cdots \leq \boldsymbol{x}_n$ and $\boldsymbol{y}_1 \leq \cdots \leq \boldsymbol{y}_n$. Eq. (6.7) reads in that case:

$$W_q^q(\boldsymbol{h}, \boldsymbol{g}) = \min_{\sigma \in S_n} \frac{1}{n} \sum_i \left| x_i - y_{\sigma(i)} \right|^q \tag{6.14}$$

in which S_n is the set of all permutations. It can be shown that choosing $\sigma(i) = i$ gives the optimal solution, meaning that the solution can be computed in $O(n\log(n))$. This result can be extended when $n \neq m$ or when the masses are not uniform: in that case, we sort the points and "fill" the transport matrix as long as the current marginal is not full. This can be solved easily with a dynamic time warping approach but the computing cost is higher than simple sorts. While this case has a limited interest per se, it is one of the main ingredients of the sliced Wasserstein distance (see below).

There also exist efficient algorithms when the two histograms are unnormalized. In that case, and considering $n = m$ and uniform histograms, [Aggarwal 1992] and [Bonneel 2019] propose a quasi-linear algorithm that solves the semi-relaxed optimal transport problem.

Solving the exact OT problem. The most common algorithmic tools to solve the discrete OT problem are borrowed from combinatorial optimization and linear programming. Indeed, it is easy to see that the Kantorovitch problem can be expressed as a linear programming program with equality constraints. Let $\boldsymbol{t} = \text{vec}(\boldsymbol{T})$, $\boldsymbol{c} = \text{vec}(\boldsymbol{C})$ and $\boldsymbol{y}^\top = [\boldsymbol{h}^\top, \boldsymbol{g}^\top]$. Problem (6.5) can be re-written as

$$\min_{\boldsymbol{t}\geq 0} \quad F(\boldsymbol{t}) \stackrel{\text{def}}{=} \boldsymbol{c}^\top \boldsymbol{t} \quad \text{such that} \quad \boldsymbol{H}\boldsymbol{t} = \boldsymbol{y} \tag{6.15}$$

where the *design matrix* $\boldsymbol{H} = [\boldsymbol{H}_r^\top, \boldsymbol{H}_c^\top]^\top$ is a $(n+m)\times(nm)$ matrix that is the concatenation of the matrices $\boldsymbol{H}_r = \mathbb{I}_n \otimes \mathbb{1}_m^\top$ and $\boldsymbol{H}_c = \mathbb{1}_n^T \otimes \mathbb{I}_m$ that allows computing sums of the rows and columns of $\boldsymbol{T}$, respectively (note that we will use this vectorization in Chapter 8.2). This formulation emphasizes that there are $n+m$ constraints, with one of them being redundant (as the mass of the 2 distributions must be equal), leading to at most $n+m-1$ non zeros values in the solution. Methods such as the network simplex or interior point methods can then be used, with a complexity being $O(n^3\log(n))$, making their use prohibitive in large scale applications (more than a few tens of thousands of points).

Some lines of works have focused on defining approximated Wasserstein distance that may be faster to compute. For instance, [Pele 2009] get a lower bound on the Wasserstein distance by thresholding the cost matrix, allowing them to reduce the complexity. [Sommerfeld 2019] propose a probabilistic sampling scheme that takes random subset of the points, giving a fast approximation of the distance. In the same line, [Genevay 2016] propose a stochastic algorithm that can be used when the distributions can be sampled from. Building on the fast computation of one dimensional optimal transport, the sliced Wasserstein distance [Rabin 2011] linearly projects high dimensional data into sets of mono-dimensional distributions and approximate the Wasserstein distance as the average of the Wasserstein distances between all projected measures. This framework provides an efficient algorithm that can handle millions of points and has similar properties to the Wasserstein distance [Bonnotte 2013].

When it comes to unbalanced optimal transport problem, [Blondel 2018] use a L-BFGS algorithm [Liu 1989] to solve problem (6.11) when the divergence is the ℓ_2 norm. [Caffarelli 2010] show that the partial optimal transport problem can be solved by adding a tariff-free reservoir (in other word, adding a dummy point on each of the distributions with an associated cost value of 0), allowing the use of standard optimization tools to solve the problem, but it does not allow them to control the amount of mass that is finally going to be optimally transported. When dealing with unnormalized histograms, [Gramfort 2015] build on the previous result to compute the results of a semi-relaxed optimal transport problem.

Solving the (regularized) entropic OT problem. To reduce the computational burden, the entropic regularized OT (6.13) can be solved using the Sinkhorn-Knopp algorithm [Sinkhorn 1967] that takes advantage of a matrix scaling algorithm, providing a significant speed-up. [Cuturi 2013] shows that the

solution of the problem can be expressed as a factorization of matrices ($\boldsymbol{T} = \text{diag}(u)\boldsymbol{K}\text{diag}(v)$, where $\boldsymbol{K} = \exp -C/\varepsilon$ is the element-wise exponential of $-C/\varepsilon$, and the so-called scaling u and v are two nonnegative vectors). It leads to very simple closed form expressions for all steps of the algorithm, allowing its use on GPU. It is shown in [Altschuler 2017] that this approach allows finding an ε-approximation for the OT distance in $O(n^2/\varepsilon^3)$ arithmetic operations, leading to poor performances when ε is small, but allowing a significant speed up for moderate values of ε. [Benamou 2015] further show that the entropic regularized problem can be recast as minimizing a Kullback Leibler divergence, then allowing them to define iterative Bregman projections to solve the problem. This reformulation enables enriching the set of constraints, then defining algorithms that solve the partial optimal transport problem (6.11) for instance. As for the balanced entropic OT problem, the convergence speed of the algorithm degrades when $\varepsilon \to 0$. Solving the UOT problem (6.11) with an entropic penalty have been notably tackled by [Chizat 2018a] who propose scaling algorithms that work when considering KL or total variation penalties (see figure 6.4 for an illustration with the KL penalty).

Sliced-Wasserstein. Sliced-Wasserstein [Rabin 2011] is a method that gives an approximation of OT, relying on the closed form of OT for 1-dimensional probability distributions (eq.(6.14)). The main idea is to randomly select lines in $\mathbb{R}^d$, to project the samples into these lines and to compute the resulting 1d-Wasserstein distance. By averaging all these 1-dimensional distances, we obtain the sliced-Wasserstein distance, with a complexity of $O(Ldn + Ln\log(n))$ where L is the number of projection directions and d the dimension of the data. It has been shown to enjoy several interesting properties: it defines a distance [Bonnotte 2013] and the approximation converge towards the actual Wasserstein distance with a sample complexity independent of the dimension [Nadjahi 2020]. Note that, by averaging 1d-dimensional distances, sliced-Wasserstein only provides an approximation of the Wasserstein distance, but does not provide the associated transport plan.

Solving the Gromov-Wasserstein problem. Adding an entropic regularization to the Gromov-Wasserstein problem (6.8) allows defining (non convex) optimization problem, using a projected gradient descent algorithm: it iterates between solving a OT problem whose cost matrix is the gradient of the estimation of the transport matrix computed at the previous iteration and the Kullback-Leibler projector of the previous solution. When the step size is small enough, this algorithm is guaranteed to converge, but again, its computational performances degrade when ε takes small values.

A different approach is to rather solve a lower bound of the Gromov-Wasserstein distance, e.g. TLB [Mémoli 2011]. It decouples the alignment term into two separate terms, allowing removing the quadratic dependence, and optimizes over each term separately:

$$TLB_q^q(\boldsymbol{C}_X, \boldsymbol{C}_Y, \boldsymbol{h}, \boldsymbol{g}) = \min_{\boldsymbol{T}^1 \in \boldsymbol{\Pi}(\boldsymbol{h},\boldsymbol{g})} \sum_{k,l} \left(\min_{\boldsymbol{T}^2 \in \boldsymbol{\Pi}(\boldsymbol{h},\boldsymbol{g})} \sum_{i,j} L(d_X(\boldsymbol{x}_i, \boldsymbol{x}_k), d_Y(\boldsymbol{y}_j, \boldsymbol{y}_l))^q T^2_{i,j} \right) T^1_{k,l} \tag{6.16}$$

The problem then comes down to a "Wasserstein distance of a Wasserstein distance" and is illustrated in figure 6.5. It involves linear programming, hence is readily computable, and may be sufficient in some applications [Chowdhury 2019].

Existing implementations. There exist several implementations that allow using optimal transport tools in different context. The Python POT library [Flamary 2021] provides recent state-of-the-art solvers for various optimal transport problems related to statistics and machine learning, allowing one to solve

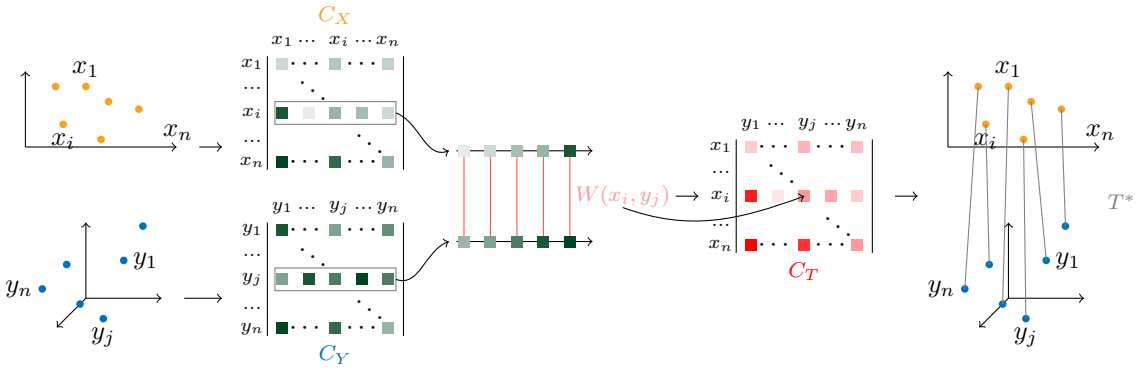


Figure 6.5: The TLB lower bound of the GW distance. From two sets of points $\mathcal{X} \subset \mathbb{R}^2$ and $\mathcal{Y} \subset \mathbb{R}^3$ with uniform distributions $\boldsymbol{h}$ and $\boldsymbol{g}$ (not depicted here), one look the optimal coupling matrix $\boldsymbol{T}^{1*}$ that minimizes the overall transport cost between intra-domain distances viewed as vectors, and then compute a second optimal coupling matrix based on the resulting cost matrix.

Kantorovich and Gromov-Wasserstein problems, compute barycenters, together with solvers for entropic OT, unbalanced and partial OT as well. It comes with many illustrative examples and notebooks. It is an open and collaborative toolbox, with more than 20 contributors, and is actively maintained and enriched.

The OTJulia [Zhang 2020] is a more complete package that can be seen as a Julia counterpart of POT. It implements several OT solvers with detailed documentation. Optimal Transport Tools (OTT, [Cuturi 2021]) is a JAX package that provides tools to compute and differentiate optimal transport problems. It is particularly efficient when dealing with the Sinkhorn algorithm.

6.4 Optimal transport and machine learning: current state and some challenges

Over the last few years, OT has quickly become a central topic in machine learning. From a field with marginal interest in the ML community about 5 years ago (Google Scholar indicates 156 results when searching for *"Optimal Transport"+"Machine Learning"* in 2016), it is now an integral component of many core ML methods (with almost 3000 results for 2021). It has been successfully employed in a wide range of challenging machine learning applications such as supervised learning [Frogner 2015], clustering [Ho 2017], generative modelling [Arjovsky 2017], domain adaptation [Courty 2016], Bayesian inference [El Moselhy 2012], reinforcement learning [Bellemare 2017], learning of structured data [Maretic 2019], fair learning [Jiang 2020] or natural language processing [Kusner 2015], among many others.

Optimal transport has been mainly used in machine learning applications in three different ways:

- use the distance (or discrepancy) provided by optimal transport: the Wasserstein distance has been shown to be a meaningful way to compare (empirical) distributions or diracs. As such, it can be used in distance-based ML algorithms, such as k-nearest neighbors [Backurs 2020] or kernel-based methods [Vayer 2019a]. The Wasserstein barycenter has also been used in k-means algorithm for instance [Cuturi 2014b], estimating parameters of Gaussian mixtures [Dessein 2017] or for missing data imputation [Muzellec 2020] to cite a few.

- use the optimal transport matrix, in applications such as domain adaptation [Courty 2016], positive-unlabeled learning [Chapel 2020] or color transfer [Korotin 2020], in which the matching between the source and the target is sought.

- use OT as a loss to update generative models. The aim can be to find a distribution that fits well the data, aligning the two of them by minimizing their expected Wasserstein distance estimation (Wasserstein GAN [Arjovsky 2017] is a flagship of such applications).

Not to mention Wasserstein gradient flows that can describe non-linear diffusion equations. All these successes have led to fruitful discussions between both the OT and ML communities. ML practitioners have first borrowed to the classical OT machinery developed by the mathematicians over the past century. But the challenges raised by the ML applications have also raised new problems about OT. In the following, I describe some of the aspects of the ML applications that have to be taken into account, that will be discussed in the next sections of this document.

Sensitivity to noise and/or outliers. A key consideration of ML algorithms is their robustness or resilience to noisy or mislabelled observations. The balanced OT formulation is sensitive to outliers since its objective function includes all the samples and features (through the cost matrix) and few outliers can disrupt the transport plan, driving the OT distance to unexpected large values (see figure 6.3). There is thus a need to develop solid theories and efficient mathematical framework that allows mitigating (or zeroing) the weights of some of the samples but also working with data in high dimensional spaces, with attributes that are prone to noise. In the very past few years, many works have been proposing robust variants of the OT distance, by mitigating the impact of the outliers thanks to the unbalanced or partial optimal transport formulation [Mukherjee 2021, Balaji 2020] or by projecting the measures into low-dimensional spaces [Paty 2019, Dhouib 2020, Jawanpuria 2020].

Need for large scale algorithms. ML practicionners usually deal with a large or huge amount of data. An important part of work then investigates large scale numerical optimization. The success of OT among the ML community is probably mainly due to the introduction of larger scale algorithms such that the entropy-regularized OT or the sliced-Wasserstein algorithm [Rabin 2011]. There is still a room for developing algorithms that challenge the heavy computational cost of solving (exact) OT; recent works rely on stochastic optimization with minibatch strategy [Genevay 2018, Fatras 2020] in order to reduce the cost per iteration when dealing with neural networks. Few works have addressed the large scale optimization of Gromov-Wasserstein: [Vayer 2019b] propose the sliced-Gromov Wasserstein that provides an approximation of the distance; [Kerdoncuff 2021] rely on subsamples to approximate the distance.

Dealing with incomparable data, that do not live in the same space, that may have suffer domain shift or are highly structured. In the machine learning community, data may not be directly comparable, because either they have been collected under distinct environments, are described by different features or cannot be represented in the form sample-features (see the introduction for a deeper discussion). There is a need for developing OT-based methods for that kind of data. Gromov-Wasserstein inherently deals with data that leave in incomparable spaces and, as such, have been used in different applications [Peyré 2016, Alvarez-Melis 2018, Maretic 2019]; CO-optimal transport aligns both the features and the samples in an uniform formulation [Redko 2020]. When it comes to data that have suffer a domain shift between the source and the target, OT have been proved to be successful when dealing with domain adaptation problems [Courty 2016, Redko 2017].

6.5 Part outline and contributions

Here is a short description of the 2 following chapters of the manuscript that give my main contributions in the field of discrete optimal transport for machine learning.

Chapter Optimal Transport for Structured Data deals with the definition of a new distance, the Fused Gromov-Wasserstein distance, that allows comparing structured data, especially graphs. It relies on Gromov-Wasserstein distance, which involves solving a quadratic programming that is intractable for moderate to large scale problem. I then describe the sliced-Gromov Wasserstein distance, built on the sliced-Wasserstein distance, that provides an approximation of the Gromov-Wasserstein distance. This approximation enables dealing with one million of points in 1 second.

Chapter Algorithms for Partial and Unbalanced Optimal Transport provides new algorithms for solving the exact partial (Gromov-)Wasserstein problem, allowing one to control the amount of mass that is going to be transported. It also makes the link with sparse numerical optimization algorithms, namely the Lasso (least absolute shrinkage and selection operator) method, allowing defining a regularization path for unbalanced OT but also efficient multiplicative algorithms for solving the exact UOT problem.

Chapter 7

Optimal Transport for Structured Data

Contents

This chapter deals with our work related to optimal transport for structured data, especially graphs. I first define the new metric that we introduced, namely the Fused Gromov-Wasserstein distance, which interpolates between the Wasserstein distance, computed on the features of the nodes of the graphs, and the Gromov-Wasserstein distance, computed between the structure of the two graphs. One of the drawbacks of the Gromov-Wasserstein distance is its high computational burden: I then present the Sliced Gromov-Wasserstein that, akin to the Sliced-Wasserstein, allows the computation of an approximate distance by averaging GW distances computed on random lines, providing a scalable algorithm. These works have been performed during the Titouan Vayer's PhD [Vayer 2020a] and were published in [Vayer 2019a, Vayer 2020c, Vayer 2019b].

7.1 Fused Gromov-Wasserstein for structured data

7.1.1 Structured objects defined as probability distributions

The notion of structured data is inspired from the discrete point of view where one aims at comparing labeled graphs. We consider undirected labeled graphs as tuples of the form $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \ell_f, \ell_s)$ where $(\mathcal{V}, \mathcal{E})$ are the set of vertices and edges of the graph. $\ell_f : \mathcal{V} \to \Omega$ is a labelling function which associates each vertex $v_i \in \mathcal{V}$ with a feature $\boldsymbol{a}_i \overset{\text{def}}{=} \ell_f(v_i)$ in some feature metric space (Ω, d). Similarly, $\ell_s : \mathcal{V} \to X$ maps a vertex v_i from the graph to its structure representation $\boldsymbol{x}_i \overset{\text{def}}{=} \ell_s(v_i)$ in some structure space (X, d_X) specific to each graph. $d_X : X \times X \to \mathbb{R}_+$ is a symmetric application which aims at measuring the similarity between the nodes in the graph. In the graph context, d_X can either encode the neighborhood information of the nodes, the edge information of the graph or more generally it can model a distance between the nodes such as the shortest path distance. When d_X is a metric, we naturally endow the structure with the metric space (X, d_X).

We enrich the previous definition of a structured object with a probability measure which serves the purpose of signaling the relative importance of the object's elements. For example, we can add a probability (also denoted as weight) $(h_i)_i \in \Sigma_n$ to each node in the graph. When all the nodes have the same importance, we set the weights to be equal. Weights can also be used to encode some a priori information. For instance, on segmented images, one can construct a graph using the spatial neighborhood of the segmented zones, the features can be taken as the average color in the zone, and the weights as the ratio of image pixels in the zone. This way, we define a fully supported probability measure $\mu = \sum_i h_i \delta_{(\boldsymbol{x}_i, \boldsymbol{a}_i)}$ which includes all the structured object information (see figure 7.1). Through this procedure, we derive the notion of structured data as a tuple $\mathcal{S} = (\mathcal{G}, h_{\mathcal{G}})$ where $\mathcal{G}$ is a graph and $h_{\mathcal{G}}$ is a function that associates a weight to each vertex.

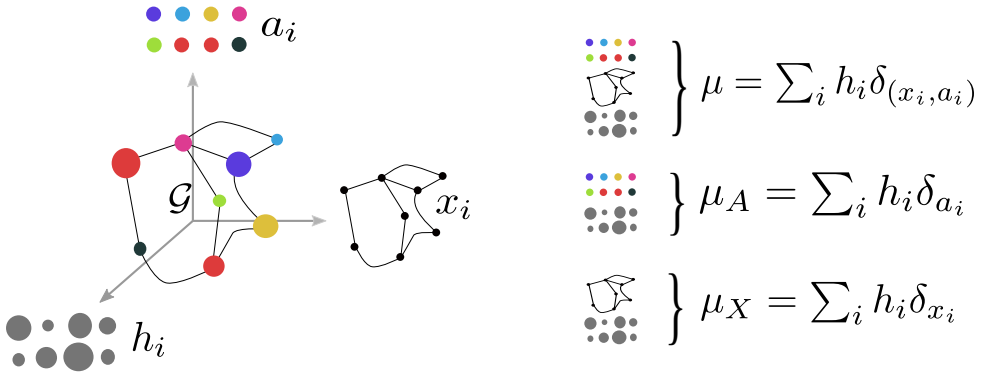


Figure 7.1: Discrete structured object (left) can be described by a labeled graph with $(\boldsymbol{a}_i)_i$ the feature information of the object and $(\boldsymbol{x}_i)_i$ the structure information. If we enrich this object with a histogram $(h_i)_i$ aiming at measuring the relative importance of the nodes, we can represent the structured object as a fully supported probability measure μ over the couple space of feature and structure with marginals μ_X and μ_A on the structure and the features respectively (right).

7.1.2 Fused Gromov-Wasserstein distance

We aim at defining a distance between two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$, described respectively by their probability measure $\mu = \sum_{i=1}^n h_i \delta_{(\boldsymbol{x}_i, \boldsymbol{a}_i)}$ and $\nu = \sum_{i=1}^m g_j \delta_{(\boldsymbol{y}_j, \boldsymbol{b}_j)}$, where $\boldsymbol{h}$ (resp. $\boldsymbol{g}$) is an histogram in the simplex Σ_n (resp. Σ_m).

The Fused Gromov-Wasserstein distance is defined for a trade-off parameter $\alpha \in [0, 1]$ as (see also figure 7.2):

$$FGW_{q,\alpha}(\boldsymbol{h}, \boldsymbol{g}) = \min_{\boldsymbol{T} \in \Pi(\boldsymbol{h}, \boldsymbol{g})} (1-\alpha) W_q^q(\boldsymbol{h}, \boldsymbol{g}) + \alpha GW_q^q(\boldsymbol{C}_1, \boldsymbol{C}_2, \boldsymbol{h}, \boldsymbol{g}) \tag{7.1}$$

$$= \min_{\boldsymbol{T} \in \Pi(\boldsymbol{h}, \boldsymbol{g})} \left\langle (1-\alpha) \boldsymbol{C}^q + \alpha L(\boldsymbol{C}_X, \boldsymbol{C}_Y)^q \otimes \boldsymbol{T}, \boldsymbol{T} \right\rangle \tag{7.2}$$

$$= \min_{\boldsymbol{T} \in \Pi(\boldsymbol{h}, \boldsymbol{g})} \sum_{i,j,k,l} \left((1-\alpha) d(\boldsymbol{a}_i, \boldsymbol{b}_j)^q + \alpha |d_X(\boldsymbol{x}_i, \boldsymbol{x}_k) - d_Y(\boldsymbol{y}_j, \boldsymbol{y}_l)|^q \right) T_{i,j} T_{k,l} \tag{7.3}$$

where $\boldsymbol{T}$ is a transport matrix whose marginals are $\boldsymbol{h}$ and $\boldsymbol{g}$.

With this definition, the resulting optimal coupling $\boldsymbol{T}^*$ is computed with respect to the structure and the features. It tends to associate pairs of feature and structure points with similar distances within each structure pair and with similar features. α allows a trade-off between the relative importance of

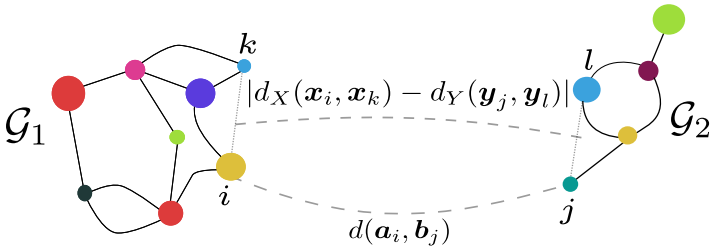


Figure 7.2: *FGW* depends on both a similarity between each feature of each node of each graph $(d(\boldsymbol{a}_i, \boldsymbol{b}_j))_{i,j}$ and between all intra-graph structure similarities $\left(\left| d_X(\boldsymbol{x}_i, \boldsymbol{x}_k) - d_Y(\boldsymbol{y}_j, \boldsymbol{y}_l) \right| \right)_{i,j,k,l}$.

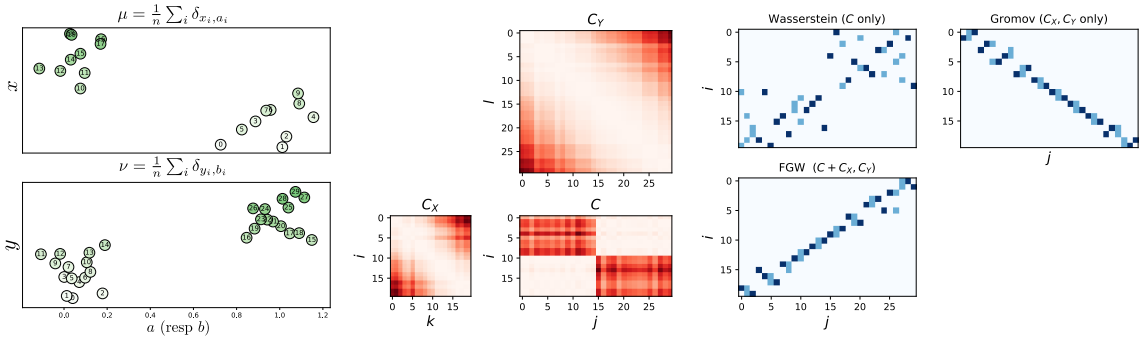


Figure 7.3: Illustration of the difference between W, GW and FGW couplings. (left) empirical distributions μ with 20 samples and ν with 30 samples whose color is proportional to their index. (middle) Cost matrices in the feature ($\boldsymbol{C}$) and structure domains ($\boldsymbol{C}_X, \boldsymbol{C}_Y$) with similar samples in white. (right) Solution for all methods. Dark blue indicates a non-zero coefficient of the transportation map between i and j. Feature distances is large between points laying on the diagonal of $\boldsymbol{C}$ such that Wasserstein maps is anti-diagonal but unstructured. Fused Gromov-Wasserstein incorporates both feature and structure maps in a single transport matrix.

the feature and the structural information. Figure 7.3 illustrates the differences between Wasserstein, Gromov-Wasserstein and Fused Gromov-Wasserstein couplings $\boldsymbol{T}^*$.

Properties and extension to continuous OT. We have shown that *FGW* enjoys some desirable properties. First, it can be defined even when the graphs have a different number of nodes that can be described by continuous or discrete attributes (basically, it can be computed as soon as intra-cost matrices $\boldsymbol{C}_X$, $\boldsymbol{C}_Y$ and outer-cost $\boldsymbol{C}$ can be defined). It also interpolates between the Wasserstein and the Gromov-Wasserstein distance: as α tends to zero, one recovers the Wasserstein distance between the features information; as α goes to one, one recovers the Gromov-Wasserstein distance between the structure information. It also defines a metric for $q = 1$ and a semi-metric for $q > 1$.

The previous graph representation for structured data with a finite number of vertices extends naturally to the continuous setting: we have also generalized FGW to general probability distributions and studied its mathematical properties in [Vayer 2020c].

Algorithmic solutions. Equation (7.3) is clearly a quadratic problem w.r.t. $\boldsymbol{T}$. We built upon a Frank-Wolfe optimization scheme [Frank 1956] a.k.a. conditional gradient algorithm [Demianov 1970]. While the problem is non convex, conditional gradient is known to converge to a local stationary point [Lacoste-Julien 2016]. Algorithm 1 gives the main step of the algorithm; note that despite its apparent $O(m^2n^2)$

complexity of computing the tensor product, one can simplify the sum to complexity $O(mn^2 + m^2n)$ when considering $q = 2$.

Algorithm 1 Frank-Wolfe algorithm for FGW

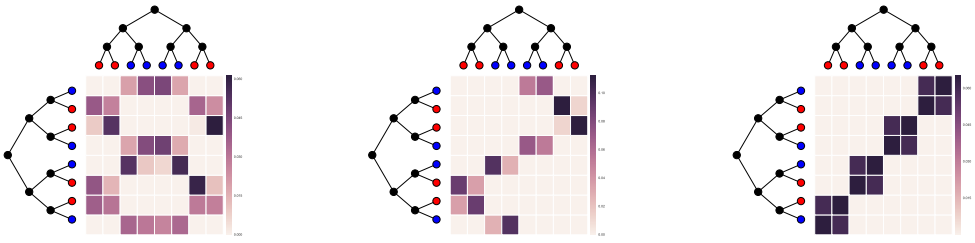
Input: Cost matrices $\boldsymbol{C}$, $\boldsymbol{C}_X$ and $\boldsymbol{C}_Y$, α, $q = 2$, initial guess $\boldsymbol{T}^{(0)}$
for $k = 0, 1, 2, 3, \ldots$ **do**
  $\boldsymbol{G}^{(k)} \leftarrow (1-\alpha)\boldsymbol{C}^q + 2\alpha L(\boldsymbol{C}_X, \boldsymbol{C}_Y)^q \otimes \boldsymbol{T}^{(k)}$ `// Compute the gradient of eq.` (7.3)
  $\widetilde{\boldsymbol{T}}^{(k)} \leftarrow \arg\min_{\boldsymbol{T} \in \Pi(\boldsymbol{h}, \boldsymbol{g})} \left\langle \boldsymbol{G}^{(k)}, \boldsymbol{T} \right\rangle$ `// Solve OT with ground cost` $\boldsymbol{G}^{(k)}$
  $\boldsymbol{E}^{(k)} \leftarrow \widetilde{\boldsymbol{T}}^{(k)} - \boldsymbol{T}^{(k)}$ `// Compute the gap`
  $\gamma^{(k)} \leftarrow \arg\min_{\gamma \in [0,1]} \left\langle (1-\alpha)\boldsymbol{C}^q + \alpha L(\boldsymbol{C}_X, \boldsymbol{C}_Y)^q \otimes (\boldsymbol{T}^{(k)} + \gamma \boldsymbol{E}^{(k)}), (\boldsymbol{T}^{(k)} + \gamma \boldsymbol{E}^{(k)}) \right\rangle$ `// Line-search`
  $\boldsymbol{T}^{(k+1)} \leftarrow (1 - \gamma^{(k)})\boldsymbol{T}^{(k)} + \gamma^{(k)} \widetilde{\boldsymbol{T}}^{(k)}$ `// Update`
end for
Return: $\boldsymbol{T}^{(k)}$

7.1.3 Experiments on structured data

We illustrate in this section the behavior of FGW on some synthetic and real datasets. The algorithm has been implemented in the Python Optimal Transport toolbox [Flamary 2021]. More results in a wider range of scenarii are provided in [Vayer 2019a] and [Vayer 2020c].

Illustration of FGW on trees

We construct two trees as illustrated in figure 7.4 where the 1D node features are shown with colors (in red, features belong to $[0, 1]$ and in blue, features are in $[9, 10]$). The structure similarity matrices $\boldsymbol{C}_X$ and $\boldsymbol{C}_Y$ are the shortest path between nodes. Figure 7.4 illustrates the behavior of regularized FGW distance when trade-off parameter α changes. One can see that, for an intermediate α (center), the bottom and first level structure is preserved as well as the feature matching (red on red and blue on blue).



(a) $\alpha = 0$ (b) Middle α (c) Strong α

Figure 7.4: Difference on transportation maps between FGW and W distances on synthetic trees for different values of alpha.

Illustration on time series

One of the main assets of FGW is that it can be used on a wide class of objects and time series are one more example of this. We consider here 25 monodimensional time series composed of two humps

in $[0, 1]$ with random uniform height between 0 and 1. A 2D embedding is computed from a FGW distance matrix with multidimensional scaling (MDS) in figure 7.5 (top). One can clearly see that the representation with a reasonable α value in the center is the most discriminant one. Figure 7.5 (bottom) illustrates the behavior of FGW on one pair of examples when going from Wasserstein to Gromov-Wasserstein, the black line depicting the matching provided by the transport matrix. Only FGW in the center finds a transport matrix that both respects the time sequences and aligns similar values in the signals.

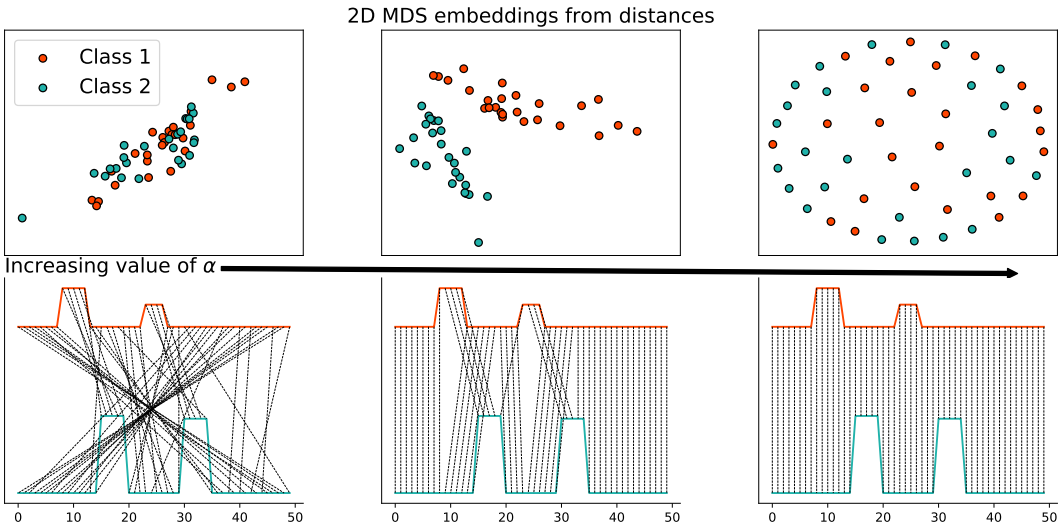


Figure 7.5: Behavior of trade-off parameter α on a toy time series classification problem. α is increasing from left ($\alpha = 0$: Wasserstein distance) to right ($\alpha = 1$: Gromov-Wasserstein distance). (Top row) 2D-embeddings are computed from the set of pairwise distances between samples with MDS. (Bottom row) illustration of couplings between two sample time series from opposite classes.

Experiments on graph classification

We also consider a graph-structured data classification context, considering several widely used benchmarked datasets that have either labeled or vector nodes attributes. We proposed to embed the FGW distance within a kernel $e^{-\gamma FGW}$, whose parameter $\gamma \geq 0$ is chosen by cross validation. Note that this kernel is not positive definite, and we regularize non-positive definite kernel shifting eigenvalue such that every eigenvalue turns positive. Relying on simple distance measure such that ℓ_2 distance between the features and the shortest path for the structure, we achieved very competitive performance when comparing classification accuracies with state-of-the-art graph kernels such as the Weisfeler Lehman Kernel [Shervashidze 2011] or the Weisfeler Optimal Assignment Kernel [Kriege 2016] and even a deep CNN on graphs (PSCN) from [Niepert 2016]. We also noticed in that context that no information can be discarded, leading to weaker performances when the nodes of the graphs have noisy labels or when the structure is corrupted by noise. This analysis drove us to study the partial or unbalanced optimal transport context, in which some mass can be discarded. Our contribution in this field will be described in the next chapter. We also face the issue of lack of stability of FGW, mainly due to the computational complexity of GW. In the next section, I describe a faster algorithm to approximate GW.

7.2 Sliced Gromov-Wasserstein

Computing (F)GW involves solving a costly non convex quadratic program which prevents its use in large scale scenarii. When it comes to solving the Wasserstein problem, two main approaches have been investigated to overcome scalability issues: i) use a regularized version of the OT formulation that allows speeding up the computations ii) slicing the problem and average low computational solutions to get an approximation. The sliced-Wasserstein approximation [Bonneel 2015] computes a solution with a complexity $O(Ldn + Ln\log(n))$ (L being the number of the sampled directions) and is several order of magnitude faster to compute. It leverages on two results: the closed-form of the Wasserstein distance when the distribution is 1-dimensional and the transformation of a distribution using Radon transforms into a set of projected 1D distributions. We build on this result to provide a fast approximation of GW. The first result is about a closed-form for GW for 1D distributions, the second is a projection operator that allows keeping the invariants (translation and rotation) of GW.

Recent results [Beinert 2022, Dumont 2022] show that eq. (7.4) is false and conterexamples exist. In practice, numerical simulations suggest that it is "often" true though and that the derived sliced Gromov-Wasserstein algorithm is efficient, that is why decided to present the closed-form for 1D GW in this manuscript.

7.2.1 Closed-form for 1D GW

We have provided and proved a solution for an 1D Quadratic Assignment Problem (QAP) with a quasi-linear time complexity of $O(n\log(n))$. This new special case of the QAP is shown to be equivalent to the *hard assignment* version of *GW* with squared Euclidean cost for distributions lying on the real line.

Let us suppose that $q = 2$, $n = m$ and $h_i = g_j = \frac{1}{n}$. We sort the points $\boldsymbol{x}_1 \leq \cdots \leq \boldsymbol{x}_n$ and $\boldsymbol{y}_1 \leq \cdots \leq \boldsymbol{y}_n$ and consider square Euclidean distance matrices $d_X(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2$ and $d_Y(\boldsymbol{y}_i, \boldsymbol{y}_j) = \|\boldsymbol{y}_i - \boldsymbol{y}_j\|_2^2$. The result states that if one wants to find the best one-to-one correspondence of real numbers such that their pairwise distances are best conserved,

$$GW_2^2(\boldsymbol{C}_X, \boldsymbol{C}_Y, \boldsymbol{h}, \boldsymbol{g}) = \min_{\sigma \in S_n} \frac{1}{n^2} \sum_{i,j} \left(d_X(\boldsymbol{x}_i, \boldsymbol{x}_j) - d_Y(\boldsymbol{y}_{\sigma(i)}, \boldsymbol{y}_{\sigma(j)})\right)^2 \tag{7.4}$$

where $\sigma \in S_n$ is a one-to-one mapping $\{1, \cdots, n\} \to \{1, \cdots, n\}$, the result is achieved by the identity $\sigma(i) = i$ or the anti-identity permutation $\sigma(i) = n + 1 - i$.

Note also that, while both possible solutions for problem (7.4) can be computed in $O(n\log(n))$, finding the best one requires the computation of the cost which seems, at first sight, to have a $O(n^2)$ complexity. However, the cost can be computed in $O(n)$ as one can develop the sum to compute it in $O(n)$ operations using binomial expansion so that the overall complexity of finding the best assignment and computing the cost is $O(n\log(n))$ which is the same complexity as the 1D Wasserstein distance.

7.2.2 Sliced Gromov-Wasserstein formulation

In order to define the sliced Gromov-Wasserstein distance, we have two problems to solve: i) how to register the two sampled directions (we cannot keep the same direction when $n \neq m$ for instance ii) how recovering some Gromov-Wasserstein invariances? We propose some solutions hereafter.

The closed-form solution of GW for one-dimensional distributions is an attractive property that gives rise to the sliced-GW distance, analogously to the sliced-Wasserstein distance. Let $\mu_X \in \mathcal{P}(\mathbb{R}^p)$ and $\mu_Y \in \mathcal{P}(\mathbb{R}^r)$, with $p \leq r$ be discrete measures with $\mu_X = \sum_i h_i \delta_{\boldsymbol{x}_i}$ and $\mu_Y = \sum_j g_j \delta_{\boldsymbol{y}_j}$. Let

$\mathbf{S}^{r-1} = \{\theta \in \mathbb{R}^r : \|\theta\|_{2,r} = 1\}$ be the r-dimensional hypersphere and λ_{r-1} the uniform measure on $\mathbf{S}^{r-1}$. For θ we note P_θ the projection on θ, *i.e.* $P_\theta(x) = \langle x, \theta \rangle$. For a linear map $\Delta \in \mathbb{R}^{r \times p}$ we define the Sliced Gromov-Wasserstein (SGW) (defined here on the measures) as follows:

$$SGW_\Delta(\mu_X, \mu_Y) = \mathbb{E}_{\theta \sim \lambda_{r-1}}[GW(\boldsymbol{C}_X, \boldsymbol{C}_Y, P_\theta \# \mu_{X\Delta}, P_\theta \# \mu_Y)] \tag{7.5}$$

where $\mu_{X\Delta} = \Delta \# \mu_X \in \mathcal{P}(\mathbb{R}^r)$. The function Δ acts as a mapping for a point in $\mathbb{R}^p$ of the measure μ_X onto $\mathbb{R}^r$. When $p < r$, one straightforward choice is $\Delta = \Delta_{pad}$ the "uplifting" operator which pads each point of the measure with zeros: $\Delta_{pad}(x) = (x_1, \dots, x_p, \underbrace{0, \dots, 0}_{r-p})$. The procedure is illustrated in Fig 7.6. We also proposed to optimize SGW_Δ with respect to Δ in the Stiefel manifold [Absil 2009], leading to the Rotation Invariant SGW ($RISGW$).

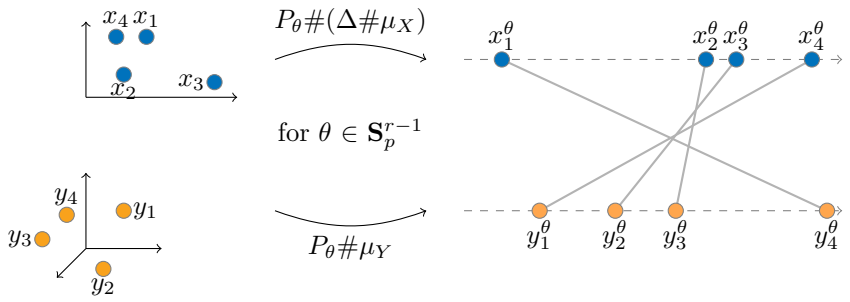


Figure 7.6: Example in dimension $p = 2$ and $q = 3$ (left) that are projected on the line (right). The solution for this projection is the anti-diagonal coupling.

Interestingly enough, SGW holds various properties of the GW distance: i) for all Δ, SGW_Δ and $RISGW$ are translation invariant. $RISGW$ is also rotational invariant when $p = q$ ii) SGW and $RISGW$ are symmetric, satisfy the triangle inequality and $SGW(\mu, \mu) = RISGW(\mu, \mu) = 0$ iii) if $SGW(\mu_X, \mu_Y) = 0$ then μ_X and μ_Y are isomorphic for the distance induced by the ℓ_1 norm on $\mathbb{R}^p$.

7.2.3 Runtimes comparison

We perform a comparison between runtimes of SGW, GW and its entropic counterpart in figure 7.7. We calculate these distances between two 2D random measures of $n \in \{1e2, ..., 1e6\}$ points. For SGW, the number of projections L is taken from $\{50, 200\}$. In this experiment, we compute SGW between 10^6 points in 1 second. Note that we recover exactly a quasi-linear slope, corresponding to the $O(n \log(n))$ complexity for SGW.

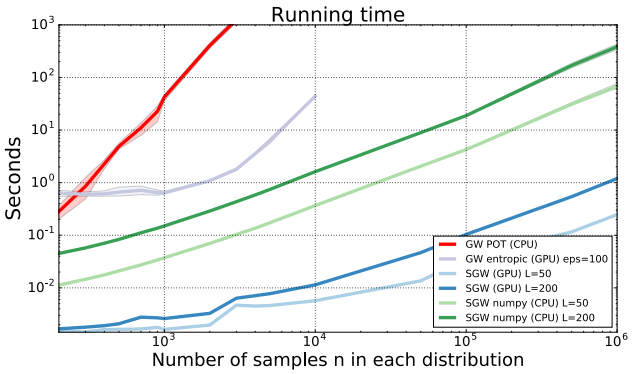


Figure 7.7: Runtimes comparison between SGW, GW, entropic-GW between two 2D random distributions with varying number of points from 0 to 10^6 in log-log scale. The time includes the calculation of the pair-to-pair distances.

Chapter 8

Algorithms for Partial and Unbalanced Optimal Transport

Contents

This chapter relates with our work on defining new algorithms for partial and unbalanced optimal transport. Optimal transport requires the two distributions to have the same total probability mass $\|\boldsymbol{h}\|_1 = \|\boldsymbol{g}\|_1$ and that all the mass has to be transported. This hypothesis may not be relevant when data are corrupted by noise, outliers or are mislabeled. Unbalanced and partial optimal transport deals with this specific problem by allowing some mass not to be transported. We proposed three new algorithms to solve these problems: i) an extension of the exact OT algorithms to compute partial (Gromov-)Wasserstein distances ii) a regularization path for solving the unbalanced OT problem for all the regularization parameter values iii) multiplicative algorithms for solving the UOT problem. From an applicative viewpoint, we also proposed to use optimal transport in the Positive-Unlabeled context, that we also describe in this chapter. Those works have been published in [Chapel 2020] and [Chapel 2021].

8.1 Exact partial Wasserstein and Gromov-Wasserstein distance

8.1.1 Partial Wasserstein as an extended Wasserstein problem

We propose here to directly solve the exact partial Wasserstein (partial-W) problem by adding *dummy* or *virtual* points $\boldsymbol{x}_{n+1}$ and $\boldsymbol{y}_{m+1}$ (with any features) and extending the cost matrix as follows:

$$\bar{\boldsymbol{C}} = \begin{bmatrix} \boldsymbol{C} & \xi \mathbb{1}_m \\ \xi \mathbb{1}_n^\top & 2\xi + A \end{bmatrix} \tag{8.1}$$

in which $A > 0$ and ξ is a fixed positive or nul scalar. When the mass of these dummy points is set such that $h_{n+1} = \|\boldsymbol{g}\|_1 - s$ and $g_{m+1} = \|\boldsymbol{h}\|_1 - s$, with s the amount of mass to be transported, computing partial-W distance boils down to solving a unconstrained problem $W_q^q(\bar{\boldsymbol{h}}, \bar{\boldsymbol{g}}) = \min_{\bar{\boldsymbol{T}} \in \Pi(\bar{\boldsymbol{h}}, \bar{\boldsymbol{g}})} \langle \bar{\boldsymbol{C}}^q, \bar{\boldsymbol{T}} \rangle_F$, where $\bar{\boldsymbol{g}} = [\boldsymbol{g}, \|\boldsymbol{h}\|_1 - s]$ and $\bar{\boldsymbol{h}} = [\boldsymbol{h}, \|\boldsymbol{g}\|_1 - s]$. It then allows using standard solvers to compute the distance. It can be shown that, when $A > 0$ and that ξ is a positive or nul scalar, one has $W_q^q(\bar{\boldsymbol{g}}, \bar{\boldsymbol{h}}) - PW_q^q(\boldsymbol{g}, \boldsymbol{h}) = \xi(\|\boldsymbol{g}\|_1 + \|\boldsymbol{h}\|_1 - 2s)$ and the optimum transport plan $\boldsymbol{T}^*$ of the partial Wasserstein problem is the optimum transport plan $\bar{\boldsymbol{T}}^*$ of $W_q^q(\bar{\boldsymbol{g}}, \bar{\boldsymbol{h}})$ deprived from its last row and column.

8.1.2 Partial Gromov-Wasserstein

We are now interested in the partial extension of Gromov-Wasserstein. In the case of a quadratic cost, $p = 2$, the partial-GW problem writes as

$$PGW_2^2(\boldsymbol{g}, \boldsymbol{h}) = \min_{\boldsymbol{T} \in \Pi^u(\boldsymbol{g}, \boldsymbol{h})} \frac{1}{2} \sum_{i,k=1}^{n} \sum_{j,l=1}^{m} \big(d_X(\boldsymbol{x}_i, \boldsymbol{x}_k) - d_Y(\boldsymbol{y}_j, \boldsymbol{y}_l)\big)^2 T_{ij} T_{kl}, \tag{8.2}$$

where $\Pi^u(\boldsymbol{g}, \boldsymbol{h})$ is defined in equation (6.9). The loss function is non-convex and the couplings feasibility domain $\Pi^u(\boldsymbol{g}, \boldsymbol{h})$ is convex and compact. One may expect to introduce virtual points in the GW formulation in order to solve the partial-GW problem. Nevertheless, this strategy is no longer valid as GW involves pairwise distances that do not allow the computations related to the dummy points to be isolated. In the following, we rather build upon a Frank-Wolfe optimization scheme [Frank 1956]. Our proposed Frank-Wolfe iterations strongly rely on computing exact partial-W distances and as such, achieve a sparse transport plan [Ferradans 2013].

Algorithm 2 Frank-Wolfe algorithm for partial-GW

Input: Cost matrices $\boldsymbol{C}_X$ and $\boldsymbol{C}_Y$, mass s, $p = 2$, initial guess $\boldsymbol{T}^{(0)}$
Build $\bar{\boldsymbol{g}} = [\boldsymbol{g}, \|\boldsymbol{h}\|_1 - s]$ and $\bar{\boldsymbol{h}} = [\boldsymbol{h}, \|\boldsymbol{g}\|_1 - s]$
for $k = 0, 1, 2, 3, \ldots$ **do**
  $\boldsymbol{G}^{(k)} \leftarrow L(\boldsymbol{C}_X, \boldsymbol{C}_Y) \circ \boldsymbol{T}^{(k)}$ `// Compute the gradient of eq.` (8.2)
  $\bar{\boldsymbol{T}}^{(k)} \leftarrow \arg\min_{\boldsymbol{T} \in \Pi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{h}})} \langle \bar{\boldsymbol{G}}^{(k)}, \boldsymbol{T} \rangle_F$ `// Compute partial-W, with` $\bar{\boldsymbol{G}}$ `as in eq.` (8.1)
  Get $\widetilde{\boldsymbol{T}}^{(k)}$ from $\bar{\boldsymbol{T}}^{(k)}$ `// Remove last row and column`
  $\boldsymbol{E}^{(k)} \leftarrow \widetilde{\boldsymbol{T}}^{(k)} - \boldsymbol{T}^{(k)}$ `// Compute the gap`
  $\gamma^{(k)} \leftarrow \arg\min_{\gamma \in [0,1]} \big\langle L(\boldsymbol{C}_X, \boldsymbol{C}_Y)^q \otimes (\boldsymbol{T}^{(k)} + \gamma \boldsymbol{E}^{(k)}), (\boldsymbol{T}^{(k)} + \gamma \boldsymbol{E}^{(k)}) \big\rangle$ `// Line-search`
  $\boldsymbol{T}^{(k+1)} \leftarrow (1 - \gamma^{(k)}) \boldsymbol{T}^{(k)} + \gamma^{(k)} \widetilde{\boldsymbol{T}}^{(k)}$ `// Update`
end for
Return: $\boldsymbol{T}^{(k)}$

Fig. 8.1 illustrates the interest of the approach when there exists a domain shift between the source and target points.

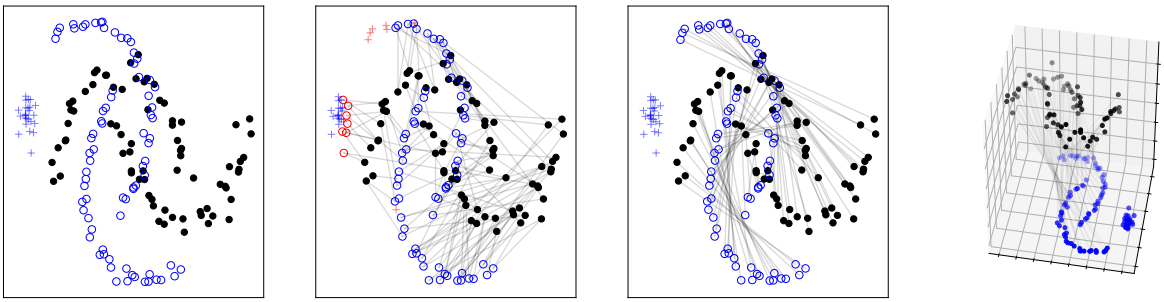


Figure 8.1: (Left) Source (in black) and target (in blue) samples that have been collected under distinct environments. The source domain contains only positive points (o) whereas the target domain contains both positives and negatives ($+$) (Middle left) Partial-W fails to assign correctly all the labels in such context, red symbols indicating wrong assignments (Middle right) Partial-GW recovers the correct labels of the unlabeled samples, with a consistent transportation plan (gray lines), even when the datasets do not live in the same state space (Right).

8.1.3 Application: partial optimal transport for Positive-Unlabeled learning

We hereafter investigate the application of partial optimal transport for learning from Positive and Unlabeled (PU) data. After introducing PU learning, we present how to formulate a PU learning problem into a partial-OT one.

Overview of PU learning. Learning from PU data is a variant of classical binary classification problem (the label taking two values: $y = 1$ or -1), in which the training data consist of only positive points, and the test data is composed of unlabeled positives and negatives. Let $\mathbf{Pos} = \{\boldsymbol{x}_i\}_{i=1}^{n_P}$ be the positive samples drawn according to the conditional distribution $p(\boldsymbol{x}|y = 1)$ and $\mathbf{Unl} = \{\boldsymbol{x}_i^U\}_{i=1}^{n_U}$ the unlabeled set sampled according to the marginal $p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y = 1) + (1 - \pi)p(\boldsymbol{x}|y = -1)$. The true proportion of positives, called class prior, is $\pi = p(y = 1)$ and $p(\boldsymbol{x}|y = -1)$ is the distribution of negative samples which are all unlabeled. The goal is to learn a binary classifier solely using **Pos** and **Unl**. A broad overview of existing PU learning approaches can be seen in [Bekker 2020].

PU learning formulation using partial optimal transport. We proposes to build on partial optimal transport to perform PU learning. In a nutshell, we aim at transporting a mass $s = \pi$ from the unlabeled (source) dataset to the positive (target) one. As such, the transport matrix $\boldsymbol{T}$ should be such that the unlabeled positive points are mapped to the positive samples (as they have similar features or intra-domain distance matrices) while the negatives are discarded (in our context, they are not transported at all).

Defining the optimal transport point-of-view of PU learning. More formally, the unlabeled points **Unl** represent the source distribution $\mathcal{X}$ and the positive points **Pos** are the target dataset $\mathcal{Y}$. We set the total probability mass to be transported as the proportion of positives in the unlabeled set, that is $s = \pi$. We look for an optimal transport plan that belongs to the following set of couplings, assuming

$n = n_U$, $m = n_P$, $g_i = \frac{1}{n}$ and $h_j = \frac{s}{m}$:

$$\Pi^{PU}(\boldsymbol{g}, \boldsymbol{h}) = \{\boldsymbol{T} \in \mathbb{R}_+^{n \times m} \mid \boldsymbol{T}\mathbb{1}_m = \{\boldsymbol{g}, 0\},\ \boldsymbol{T}^\top \mathbb{1}_n \leq \boldsymbol{h},\ \mathbb{1}_n^\top \boldsymbol{T}\mathbb{1}_m = s\}, \tag{8.3}$$

in which $\boldsymbol{T}\mathbb{1}_m = \{\boldsymbol{g}, 0\}$ means that $\sum_{j=1}^m T_{i,j} = g_i$ exactly or 0, $\forall i$ to avoid matching part of the mass of an unlabeled negative with a positive. This set is not empty as long as $s \mod g_i = 0$. The problem that we aim at solving is the following:

$$PUW_q^q(\boldsymbol{g}, \boldsymbol{h}) = \min_{\boldsymbol{T} \in \Pi^{PU}(\boldsymbol{g}, \boldsymbol{h})} \sum_{i=1}^n \sum_{j=1}^m C_{i,j}^q T_{i,j}.$$

Though the positive samples **Pos** are assumed easy to label, their features may be corrupted with noise or they may be mislabeled. Let assume $0 \leq \alpha \leq 1 - s$, the noise level.

Solving the PU problem. To enforce the condition $\boldsymbol{T}\mathbb{1}_m = \{\boldsymbol{g}, 0\}$, we adopt a regularized point of view of the partial OT problem as in [Courty 2016] and we solve the following problem:

$$\bar{\boldsymbol{T}}^* = \underset{\bar{\boldsymbol{T}} \in \Pi(\bar{\boldsymbol{g}}, \bar{\boldsymbol{h}})}{\arg\min} \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} \bar{C}_{i,j}^q \bar{T}_{i,j} + \eta \Omega(\bar{\boldsymbol{T}}) \tag{8.4}$$

where $g_i = \frac{1-\alpha}{n}$, $h_j = \frac{s+\alpha}{m}$, $\eta \geq 0$ is a regularization parameter and α is the percentage of **Pos** that we assume to be noisy (that is to say we do not want to map them to a point of **Unl**). We choose $\Omega(\bar{\boldsymbol{T}}) = \sum_{i=1}^n \left(\|\bar{\boldsymbol{T}}_{i(:m)}\|_2 + \|\bar{\boldsymbol{T}}_{i(m+1)}\|_2\right)$ where $\bar{\boldsymbol{T}}_{i(:m)}$ is a vector that contains the entries of the i^{th} row of $\bar{\boldsymbol{T}}$ associated to the first m columns. This group-lasso regularization leads to a sparse transportation map and enforces each of the **Unl** samples $\boldsymbol{x}_i$ to be mapped to only the **Pos** samples or to the dummy point $\boldsymbol{y}_{m+1}$. When partial-GW is involved, we use this regularized-OT in the step (i) of the Frank-Wolfe algorithm.

We have established that solving problem (8.4) provides the solution to PU learning using partial-OT. Assume that $A > 0$, ξ is a constant, there exists a large $\eta > 0$ such that: $W_q^{*q}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}) - PUW_q^q(\boldsymbol{p}, \boldsymbol{q}) = \xi(1 - s)$. where $W_q^{*q}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}) = \sum_{i=1}^{n+1} \sum_{j=1}^{m+1} \bar{C}_{i,j}^q \bar{T}_{i,j}^*$ with $\bar{\boldsymbol{T}}$ solution of eq. (8.4).

Experiments on different PU scenarii. Following previous works [Kato 2019, Hsieh 2019], we assume that the class prior π is known throughout the experiments; otherwise, it can be estimated from $\{\boldsymbol{x}_i\}_{i=1}^{n_P}$ and $\{\boldsymbol{x}_i^U\}_{i=1}^{n_U}$ using off-the-shelf methods, e.g. [Zeiberg 2020, Christoffel 2017, Jain 2016].

SCAR scenario. Most PU learning methods commonly rely on the selected completely at random (SCAR) assumption [Elkan 2008] which assumes that the labeled samples are drawn at random among the positive distribution, independently of their attributes. Considering six datasets from the UCI repository, we show that partial-W competes and sometimes outperforms state-of-the-art PU learning algorithms such as [Du Plessis 2014, Kato 2019] in this context. We also show that Partial-GW has competitive results, showing that relying on intra-domain matrices may allow discriminating the classes. It nevertheless under-performs relatively to partial-W, as the distance matrix $\boldsymbol{C}$ between **Pos** and **Unl** is more informative than only relying on intra-domain matrices.

SAR scenario. The SCAR assumption is often violated in real-case scenarii and PU data are often subject to selection biases, e.g. when part of the data may be easier to collect. The selected at random (SAR) setting [Bekker 2020, Hsieh 2019, Kato 2019] assumes that the positives are labeled according to a subset of features of the samples. When constructing a dataset whose probability of the points to be labeled depends on their label, we show that partial-GW allows maintaining a good level of accuracy while partial-W and classical PU methods have degraded performances.

Heteregeneous domains and/or features scenario. We further validate the proposed method in a domain adaptation context, in which the domains and/or the features may vary between the positive and the unlabeled dataset, considering the four domains of the `Caltech office` dataset [Saenko 2010] described by two types of features (`SURF` [Saenko 2010] and `DECAF` [Donahue 2014]). Again, we observe that partial-W competes with state-of-the-art PU learning algorithms when the domains and the features are the same; that partial-GW provides the best results when the domains differ, suggesting that it is able to capture the domain shift. We also propose a more challenging scenario in which both domains and features are different: in that case, only partial-GW is applicable and the performances we observe suggest that it is able to efficiently leverage on the discriminative information conveyed by intra-domain similarity matrices.

8.2 The regularization path of unbalanced optimal transport

After proposing an algorithm for solving partial (Gromov-)Wasserstein, we now focus on the unbalanced OT (UOT) formulation (eq. (6.11)) of OT.

8.2.1 UOT cast as a weighted Lasso problem with positivity constraints

We first consider UOT with a quadratic divergence for penalizing the deviation from the true marginals.

We first rewrite the Unbalanced OT problem associated with a ℓ_2 divergence in a vectorized form. Let $\boldsymbol{t} = \text{vec}(\boldsymbol{T})$, $\boldsymbol{c} = \text{vec}(\boldsymbol{C})$ and $\boldsymbol{y}^\top = [\boldsymbol{h}^\top, \boldsymbol{g}^\top]$. Problem (6.11) can be re-written as

$$\min_{\boldsymbol{t} \geq 0} \quad F_\lambda(\boldsymbol{t}) \stackrel{\text{def}}{=} \frac{1}{\lambda} \boldsymbol{c}^\top \boldsymbol{t} + \|\boldsymbol{H}\boldsymbol{t} - \boldsymbol{y}\|_2^2 \tag{8.5}$$

where the *design matrix* $\boldsymbol{H} = [\boldsymbol{H}_r^\top, \boldsymbol{H}_c^\top]^\top$ is the concatenation of the matrices $\boldsymbol{H}_r$ and $\boldsymbol{H}_c$ that compute sums of the rows and columns of $\boldsymbol{T}$, respectively (see eq. (6.15)). By noting that $\boldsymbol{t} \geq 0$ and $\boldsymbol{c} \geq 0$, the linear term can be expressed as $\frac{1}{\lambda}\boldsymbol{c}^\top \boldsymbol{t} = \frac{1}{\lambda}\sum_i c_i t_i = \frac{1}{\lambda}\sum_i c_i |t_i|$. This corresponds to a weighted ℓ_1 regularization, promoting sparsity in $\boldsymbol{t}$ and hence in the transport plans. Note that the "sparse" regularization is here controlled by $\frac{1}{\lambda}$ (instead of λ in classical penalized linear regression), meaning that the sparsity promoting term will be more aggressive for small λ. That analogy being pointed out, we leverage results about non-negative Lasso to design an efficient algorithm to compute the first regularization path of ℓ_2-penalized UOT, computing the whole set of solutions for a varying λ ranging from 0 to $+\infty$.

8.2.2 Regularization path of UOT

As for the Lasso, the path is piecewise linear in $1/\lambda$ between changes in the active set $\mathcal{A} = \text{supp}(\boldsymbol{t}^\lambda)$, where $\boldsymbol{t}^\lambda = \text{vec}(\boldsymbol{T}^\lambda)$ and $\boldsymbol{T}^\lambda$ is the OT plan for given hyperparameter λ. The main steps of the algorithm are roughly as follows: given a current solution $(\lambda_k, \boldsymbol{T}^{\lambda_k})$ and a current active set $\mathcal{A}_k$, we look for the next value $\lambda_{k+1} > \lambda_k$ such that the active set changes (i.e., $\mathcal{A}_{k+1} \neq \mathcal{A}_k$), either because one component enters the active set or because one leaves it. Thanks to the piecewise linearity of the path, we can also compute all the solutions for any $\lambda \in [\lambda_k, \lambda_{k+1}]$.

Piecewise linearity of the path Assume that, at iteration k, we know the current active set $\mathcal{A} = \mathcal{A}_k$ and we look for $\boldsymbol{t}_\mathcal{A}^\lambda$ (the other values of $\boldsymbol{t}_\mathcal{A}$ being 0). Let $\boldsymbol{H}_\mathcal{A}$, $\boldsymbol{m}_A$ and $\boldsymbol{c}_\mathcal{A}$ denote the corresponding sub-matrix and vectors. We showed that the optimal $\boldsymbol{t}_\mathcal{A}^\lambda$ (and hence $\boldsymbol{t}^\lambda$) can be solved for any $\lambda \in [\lambda_k, \lambda_{k+1}]$,

i.e., when the active set $\mathcal{A}$ remains the same, by solving a linear problem

$$\boldsymbol{t}_{\mathcal{A}}^{\lambda} = \tilde{\boldsymbol{m}}_{\mathcal{A}} - \frac{1}{\lambda}\tilde{\boldsymbol{c}}_{\mathcal{A}} \tag{8.6}$$

with $\tilde{\boldsymbol{m}}_{\mathcal{A}} = (\boldsymbol{H}_{\mathcal{A}}^{\top}\boldsymbol{H}_{\mathcal{A}})^{-1}\boldsymbol{m}_{\mathcal{A}}$, $\boldsymbol{m}_{\mathcal{A}}$ being the appropriate rows of $\boldsymbol{m} = \boldsymbol{H}^{\top}\boldsymbol{y} = \mathrm{vec}(\boldsymbol{a}\mathbb{1}_m^{\top} + \mathbb{1}_n\boldsymbol{b}^{\top})$, and $\tilde{\boldsymbol{c}}_{\mathcal{A}} = (\boldsymbol{H}_{\mathcal{A}}^{\top}\boldsymbol{H}_{\mathcal{A}})^{-1}\boldsymbol{c}_{\mathcal{A}}$. Eq. (8.6) reveals the piecewise linearity in λ^{-1} of the path when $\mathcal{A}$ is fixed. As expected, balanced OT is recovered when $\lambda \to \infty$. Eq. (8.6) involves the computation of the matrix $(\boldsymbol{H}_{\mathcal{A}}^{\top}\boldsymbol{H}_{\mathcal{A}})^{-1}$, whose computational burden can be alleviated by using the Schur complement as only one index leaves or enters the active set at each iteration.

Finding $(\lambda_{k+1}, \mathcal{A}_{k+1})$ given $(\lambda_k, \mathcal{A}_k)$. Given a current solution $(\lambda_k, \boldsymbol{t}^{\lambda_k})$, we look for the next λ_{k+1} such that we observe a change in the set of active components. This happens whenever the first of the following two situations occurs.

- One component in $\mathcal{A}$ becomes inactive. In that case, we remove the index $i \in \mathcal{A}$ with the smallest $\lambda_r > \lambda_k$ that violates the constraint which is given by

$$\lambda_r = \min_{>\lambda_k} \left(\frac{\tilde{\boldsymbol{c}}_{\mathcal{A}}}{\tilde{\boldsymbol{m}}_{\mathcal{A}}} \right) \tag{8.7}$$

  where $\min_{>\lambda_k}$ indicates the minimum value in the vector greater than λ_k and the division is entrywise.

- One component in $\bar{\mathcal{A}}$ becomes active. We remove the index $i \in \mathcal{A}$ with the smallest $\lambda_a > \lambda_k$ that violates the positivity constraint:

$$\lambda_a = \min_{>\lambda_k} \left(\frac{\boldsymbol{c}_{\bar{\mathcal{A}}} - \left[\boldsymbol{H}^{\top}\boldsymbol{H}\tilde{\boldsymbol{c}}\right]_{\bar{\mathcal{A}}}}{\boldsymbol{m}_{\bar{\mathcal{A}}} - \left[\boldsymbol{H}^{\top}\boldsymbol{H}\tilde{\boldsymbol{m}}\right]_{\bar{\mathcal{A}}}} \right), \tag{8.8}$$

  where $\tilde{\boldsymbol{m}}$ (resp. $\tilde{\boldsymbol{c}}$) equals $\tilde{\boldsymbol{m}}_{\mathcal{A}}$ (resp. $\tilde{\boldsymbol{c}}_{\mathcal{A}}$) on $\mathcal{A}$ and zero on $\bar{\mathcal{A}}$.

In practice, at each step of the path, we set $\lambda_{k+1} = \min\{\lambda_r, \lambda_a\}$ and update the active set accordingly.

8.2.3 Numerical illustration

We first illustrate the regularization path for ℓ_2-penalized UOT on a simple example between two distributions containing 3 points each, with different masses and a cost matrix $\boldsymbol{C}$ given in figure 8.2 (left). We can see in figure 8.2 (right) that, starting from $\lambda_0 = 0$ and $\boldsymbol{T} = 0$, we successively add or remove components in the active set $\mathcal{A}$ when increasing the λ values. When $\lambda = \infty$, we recover the balanced OT solution. Recall that the path is linear in $1/\lambda$ (and not λ). We then illustrate the path for both ℓ_2-penalized UOT and semi-relaxed UOT on two 2D distributions with $n = m = 100$ samples. We can see in figure 8.3 the difference between the two regularization paths for specific values of λ. UOT starts with an empty plan for $\lambda = 0$ and then activates samples from both source and target from the closest to the farthest ones until convergence to the balanced OT plan. Semi-relaxed UOT starts with all target samples active due to marginal constraints and progressively activates the source samples.

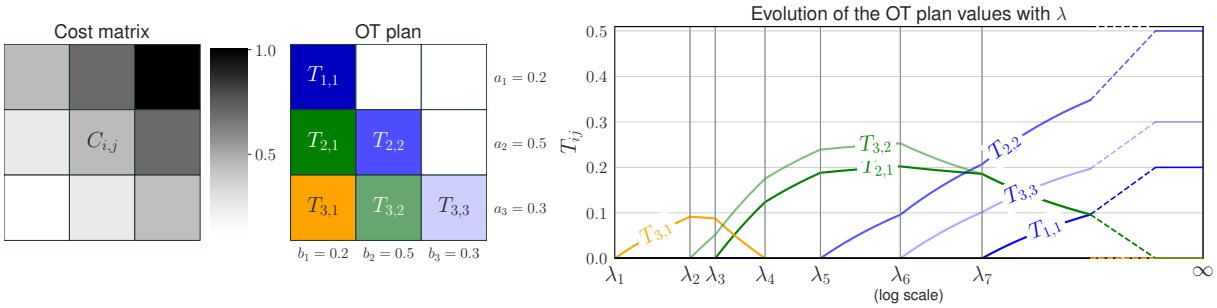


Figure 8.2: (Left) cost matrix $\boldsymbol{C}$ (the higher the cost, the darker the color); (middle) OT plan whose cells are color-coded with respect to the λ values at which they are activated. The blank cells never enter the active set as the corresponding cost it too high; (right) evolution of $T_{i,j}$ when λ increases. Note that the x-axis is in log scale and is discontinued between λ_7 and ∞.

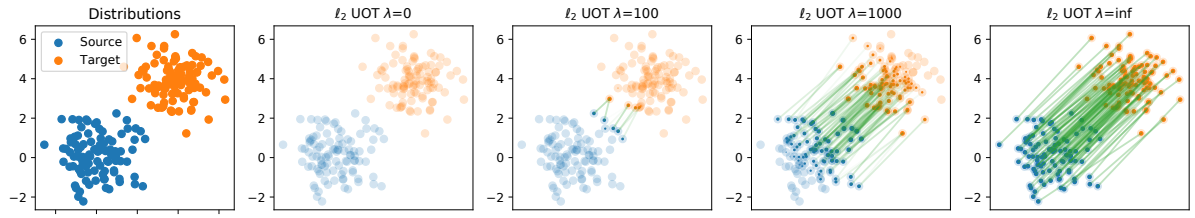


Figure 8.3: Regularization paths for 2D empirical distributions for ℓ_2-penalized UOT. The OT plan is shown as green lines between the source and target samples when $T_{i,j} > 0$ and the resulting marginals are shown as filled circles.

8.3 Multiplicative algorithms for unbalanced optimal transport

8.3.1 UOT cast as a regression problem with a Bregman divergence

More generally than eq. (8.5), we can use a Bregman divergence to quantify how two distributions differ. The UOT problem then writes as:

$$\min_{\boldsymbol{t} \geq 0} \quad F_\lambda(\boldsymbol{t}) \stackrel{\text{def}}{=} \frac{1}{\lambda} \boldsymbol{c}^\top \boldsymbol{t} + D_\varphi(\boldsymbol{H}\boldsymbol{t}, \boldsymbol{y}) \tag{8.9}$$

in which D_φ is the Bregman divergence generated by the strictly convex and differentiable function φ.

Problems of the form of eq. (8.9) are well-known in inverse problems and non negative matrix factorization (NMF). In contrast to problem (8.9), the data fitting term is more commonly $D_\varphi(\boldsymbol{y}, \boldsymbol{H}\boldsymbol{t})$ instead of $D_\varphi(\boldsymbol{H}\boldsymbol{t}, \boldsymbol{y})$ in those communities. This is because the former is a log-likelihood in disguise for the mean-parametrized exponential family, and takes important noise models as special cases, such as Poisson, additive Gaussian or multiplicative Gamma noise [Févotte 2011]. This analogy allows bringing into optimal transport multiplicative algorithms commonly used in the NMF and inverse problem communities.

8.3.2 Majorization-Minimization (MM) for UOT

General MM framework In a nutshell, MM consists in iteratively building and minimizing an upper bound of the objective function which is tight at the current parameter estimate (and referred to as *auxiliary function*), see [Hunter 2004, Sun 2017] for tutorials. In NMF, a common approach consists in

alternating the updates of the dictionary $\boldsymbol{H}$ and of the embeddings. In our case, $\boldsymbol{H}$ is fixed and we may use the results of [Dhillon 2005] to build an auxiliary function for term $D_\varphi(\boldsymbol{Ht}, \boldsymbol{y})$, to which we may simply add the linear term $\boldsymbol{c}^\top \boldsymbol{t}/\lambda$ to obtain a valid auxiliary function for $F_\lambda(\boldsymbol{t})$. Let $\tilde{\boldsymbol{t}}$ denote the current estimate of $\boldsymbol{t}$, $\tilde{Z}_{i,j} = \frac{H_{i,j}\tilde{t}_j}{\sum_l H_{i,l}\tilde{t}_l}$ and

$$G_\lambda(\boldsymbol{t}, \tilde{\boldsymbol{t}}) = \sum_{i,j} \tilde{Z}_{i,j} \varphi\left(\frac{H_{i,j} t_j}{\tilde{Z}_{i,j}}\right) + \sum_j \left[\frac{c_j}{\lambda} - \sum_i H_{i,j}\varphi'(y_i)\right] t_j + cst, \tag{8.10}$$

where $cst = \sum_i [\varphi'(y_i) y_i - \varphi(y_i)]$. Then, $G_\lambda(\boldsymbol{t}, \tilde{\boldsymbol{t}})$ is an auxiliary function for $F_\lambda(\boldsymbol{t})$, i.e., $\forall \boldsymbol{t}$, $G_\lambda(\boldsymbol{t}, \tilde{\boldsymbol{t}}) \geq F_\lambda(\boldsymbol{t})$ and $G_\lambda(\tilde{\boldsymbol{t}}, \tilde{\boldsymbol{t}}) = F_\lambda(\tilde{\boldsymbol{t}})$. Let $\boldsymbol{p}^{(k+1)} = \operatorname{argmin}_{\boldsymbol{t}\geq 0} G_\lambda(\boldsymbol{t}, \boldsymbol{t}^{(k)})$, then $F_\lambda(\boldsymbol{t}^{(k)}) = G_\lambda(\boldsymbol{t}^{(k)}, \boldsymbol{t}^{(k)}) \geq G_\lambda(\boldsymbol{t}^{(k+1)}, \boldsymbol{t}^{(k)}) \geq F_\lambda(\boldsymbol{t}^{(k+1)})$, producing a descent algorithm over F. The trick to obtain G is to apply Jensen inequality to $\varphi(\sum_j H_{i,j} t_j) = \varphi(\sum_j \tilde{Z}_{i,j} \frac{H_{i,j}}{\tilde{Z}_{i,j}} t_j) \leq \sum_j \tilde{Z}_{i,j} \varphi(\frac{H_{i,j}}{\tilde{Z}_{i,j}} t_j)$, thanks to the convexity of φ, see details in [Dhillon 2005]. We provide below the resulting algorithms for the KL and ℓ_2 penalizations.

MM for KL-penalized UOT. The KL divergence is obtained with $\varphi(y) = y \log y - y$. Minimizing $G_\lambda(\boldsymbol{t}, \boldsymbol{t}^{(k)})$ in that case leads to following multiplicative update:

$$t_j^{(k+1)} = t_j^{(k)} \exp\left(\frac{[\boldsymbol{H}^\top \log(\boldsymbol{y}) - \boldsymbol{H}^\top \log\left(\boldsymbol{H}\boldsymbol{t}^{(k)}\right)]_j - \frac{1}{\lambda} c_j}{[\boldsymbol{H}^\top \mathbb{1}]_j}\right). \tag{8.11}$$

Owing to the structure of $\boldsymbol{p}$ and $\boldsymbol{H}$, the update can be re-written in the following matrix form:

$$\boldsymbol{T}^{(k+1)} = \operatorname{diag}\left(\frac{\boldsymbol{a}}{\boldsymbol{T}^{(k)} \mathbb{1}_m}\right)^{\frac{1}{2}} \left(\boldsymbol{T}^{(k)} \odot \exp\left(-\frac{\boldsymbol{C}}{2\lambda}\right)\right) \operatorname{diag}\left(\frac{\boldsymbol{b}}{\boldsymbol{T}^{(k)\top} \mathbb{1}_n}\right)^{\frac{1}{2}}, \tag{8.12}$$

where $\odot$ is entrywise multiplication and divisions are taken entrywise as well. The multiplicative update in eq. (8.12) is remarkably similar to the Sinkhorn-Knopp algorithm. But instead of two separate steps for the left and right scaling, Eq. (8.12) applies these scalings simultaneously in a unique update using the diagonal matrices (and a form of geometrical average). Factor $\exp\left(-\frac{\boldsymbol{C}}{2\lambda}\right)$ penalizes along iterations the coefficients of the transport plan with large costs.

MM for ℓ_2-penalized UOT. The quadratic loss is obtained with $\varphi(y) = \frac{y^2}{2}$. In that case, minimizing $G_\lambda(\boldsymbol{t}, \boldsymbol{t}^{(k)})$ leads to following multiplicative update:

$$\boldsymbol{T}^{(k+1)} = \boldsymbol{T}^{(k)} \odot \frac{\max\left(0, \boldsymbol{a}\mathbb{1}_m^\top + \mathbb{1}_n \boldsymbol{b}^\top - \frac{1}{\lambda}\boldsymbol{C}\right)}{\boldsymbol{T}^{(k)} \mathbf{O}_m + \mathbf{O}_n \boldsymbol{T}^{(k)}} \quad \text{with} \quad \mathbf{O}_\ell = \mathbb{1}_\ell \mathbb{1}_\ell^\top. \tag{8.13}$$

8.3.3 Study of the performances of the algorithms

We provide an empirical evaluation of the running times of the proposed algorithms, using 2 sets of 10-dimensional points with $n = m = 500$ and drawn according to i.i.d. Gaussian distributions. The cost matrix $\boldsymbol{C}$ is computed using a squared ℓ_2 norm. We compare the running times of the current state-of-the-art BFGS algorithm [Blondel 2018] using SciPy [Virtanen 2020] and those of the ℓ_2-penalized UOT formulated as a Lasso problem (with both the Celer algorithm [Massias 2018] and the coordinate descent solvers from Scikit-learn [Pedregosa 2011]), the multiplicative algorithm for both the ℓ_2 and the KL penalties and the regularization path algorithm. Figure 8.4 shows the average running time for all algorithms. For ℓ_2-penalized UOT, we observe that, for large λ values, the Lasso solvers are the fastest and that, whatever the value λ, BFGS is the slowest. As for KL-penalized UOT, the BFGS algorithm

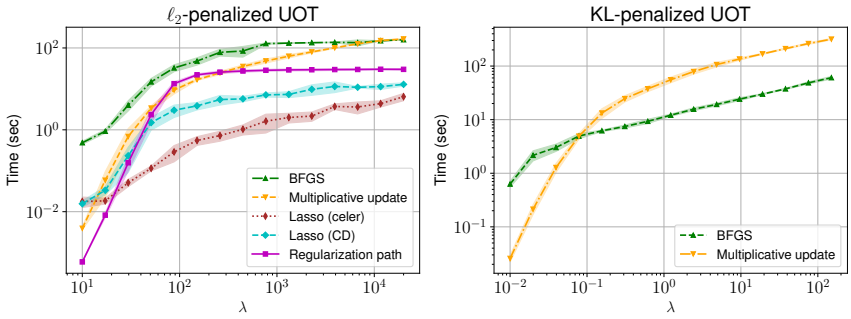


Figure 8.4: (Left) comparison of ℓ_2-penalized UOT with other algorithms (right) likewise for KL-penalized UOT. Dark curves (resp. shaded regions) represent average (resp. variance) values over 5 runs.

is more efficient when large values of λ are considered. One can also notice that, similarly to Sinkhorn which is fast for large regularization values, the multiplicative algorithms for both penalties are also fast for high $1/\lambda$ values.

Chapter 9

Concluding remarks and perspectives

In this last chapter, I provide some concluding remarks about my work but also give some perspectives that I plan to investigate or that I think are of prime importance.

In the field of optimal transport for machine learning, we have provided new algorithms for solving unbalanced optimal transport but I believe that there is still much to investigate in this context. Indeed, it is a much more realistic context for machine learning as it allows removing the noise or outliers that are very often present in the data. I see two main lines of work: i) build solutions that detect out-of-distribution samples, without the need to tune a parameter; ii) define more scalable solutions for UOT. With such tools at hand, one expects OT to reach better performances in machine learning applications, notably when applied to structured data.

Regarding time series, we have provided solutions that rely on well-established tools in the community, such as kernels and bag of words. I believe that a fruitful cross-fertilization between DTW and OT is worth being investigating as they share a lot of common features. Also, the topic of temporal transfer learning is still under-studied in the ML time series community despite being a natural framework for evolving data. Again, among other possible options, OT tools could be a natural playground in this context.

Finally, the highly structured features of remote sensing images drove me to the study of graphs as objects of interest, and led my more recent works in the field of optimal transport. The journey took some time, and from an application point of view, it is now time to go back and check if those methods are actually relevant for the RS community. In the meantime, hyperbolic geometry has been shown to be an efficient framework for representing tree-structured data. In a more methodological point of view, it is worth checking if this framework fits within the RS context and to develop more integrated frameworks that explicitely takes into account the structure within the learning process. In a more prospective line, transferring knowledge from different modalities is, to my viewpoint, one of the most interesting challenge to tackle in the community.

I now give some more details about those perspectives.

9.1 Perspectives for unbalanced optimal transport and machine learning

9.1.1 Detecting outliers or out-of-distribution samples

Optimal transport suffers from its brittleness with respect to outliers or corrupted samples, that often puts its relevance in jeopardy: just one corrupted value in the cost matrix can render the transport matrix and the OT distance arbitrary far away the true value. Recently, *robust* versions of optimal transport have been developped [Mukherjee 2021, Balaji 2020] with the aim to produce a transport plan,

a distance or a parametric distribution (when looking at minimum Kantorovitch estimators for instance) that do not take into account outliers or out-of-distribution samples. They often rely on unbalanced or partial optimal transport formulations; nevertheless, the problem of the choice of the *best* parameter remains. This problem is sometimes even crucial when its value should change over the iterations of the algorithm, e.g. when dealing with minibatches [Fatras 2021]. Some works choose the regularization parameter by cross-validation, but it then requires a supervised formulation of the problem. This leads to the following question: how to choose (in an unsupervised manner) the *best* regularization parameter? On the other hand, OT-based statistics are becoming more and more popular in statistics either for inference or testing purposes. For instance, [Del Barrio 2019] define a notion of variability of the Wasserstein distance, allowing them to reject the hypothesis of the similarity of two distributions when the minimal alignment cost exceeds some threshold. I believe that the regularization path algorithm that we provide in [Chapel 2021] can provide a tool to solve the problem, by introducing a comparison with an expected distribution of the OT distance under the hypothesis that the distributions are the same over two values of the path. This first implies to study in futher details the properties of the path of the UOT, e.g. does the monotonicity of the support of the marginals holds in the formulation, as it does for partial OT [Caffarelli 2010]? One other option would be to add an additional regularization parameter that introduces an additional term that penalizes the difference between two distributions (in the same vein as [Brécheteau 2019] for the Gromov-Wasserstein distance). In an other direction, the sliced-Wasserstein distance is based on averaging different ordering of the data (when projected on the line). It would be interesting to investigate if those projections could allow defining multivariate statistics such as quantiles or ranks (in the same vein as [Hallin 2021]) for statistical inference.

9.1.2 Defining scalable algorithms for unbalanced or partial optimal transport

Optimal transport also suffers from computational limitations that hinder its use for massive dataset. Despite the introduction of the Sinkhorn regularization, that is nearly $O(n^2)$, and the definition of stochastic procedures, the resolution of the problem is still slow for small regularization terms. The sliced-Wasserstein approximation has alleviated the computational burden and also improved the statistical properties in high dimensions, with a complexity of $O(n\log(n))$, and a convergence rate of $\sqrt{n}$ instead of $O(n^{-1/d})$ for the Wasserstein distance [Nadjahi 2019]. When it comes to unbalanced and partial optimal transport, such computational tools are still lacking, together with a analysis of their theoretical properties.

Building on 1d OT. Solving 1-dimensional optimal transport is straighforward as it suffices to sort the input samples. When it comes to exact 1d unbalanced or partial optimal transport, very few methods have been developped. One can cite [Bonneel 2019] that propose a solver for a simplified partial OT problem in $O(nm)$, in which the set of admissible transport plans is restricted to injections between the two distributions. The very recent work of [Sejourne 2021] build on iterations of a Frank-Wolfe algorithm, which leads to a $O(n+m)$ iteration approximate algorithm. Then, the questions that arise are the following: can we do better? How deriving a provable sliced-UOT algorithm, with a coherent regularization parameter among the projections? Can we derive an efficient regularization path in case of 1d-UOT? When it comes to structured data and the use of Gromov-Wasserstein, we have defined the 1d solution in [Vayer 2019b]. It would be of high interest to see its extension to 1d-unbalanced Gromov-Wasserstein.

Building on regularization paths. In order to speed up the computations, one other option is to define *approximate* solutions of the OT problem. The Sinkhorn algorithm is a flagship example of this strategy, but other schemes could be defined. When it comes to regularization path, the approach of [Mairal 2012] can compute a regularization path with precision ε in $O(1/\varepsilon)$ iterations. This would lead to a full complexity of $O(nm/\varepsilon)$ that is even interesting to approximate balanced OT. Albeit non convex, would it be possible to obtain a regularization path of GW? Indeed, the formulation of the ℓ_2 regularized UOT problem contains a quadratic term (corresponding to the deviation of the marginals) plus a weighted ℓ_1 regularization term (the OT term). One could then derive an alternative formulation, in which the quadratic term would correspond to the GW objective, and the ℓ_1 term to the penalization of the marginals. Obviously, one would not ensure the optimality of the solution but the convergence towards a fixed point may be obtained. In that case, it would allow lowering the computational cost of unbalanced GW computation.

9.1.3 Alternative formulations of the unbalanced optimal transport problem

Unbalanced or partial optimal transport relies on the fact that we autorize deviation from the true marginals. Other viewpoints could be considered in order to define alternative formulations. A crucial question in OT is the choice of the ground metric [Cuturi 2014a]: the ℓ_1 or ℓ_2 norm are often chosen but they can be poor choice in practice for high dimensional data, when samples live on a manifold or when they are structured. Several works have considered the optimization of OT that maximizes the ground cost [Niles-Weed 2019,Paty 2019] and the link between regularizing and maximizing optimal transport has been formalized in [Paty 2020]. Learning the ground metric would allow selecting an optimal transport distance w.r.t. the problem at hand. For classification of structured data, one could learn a data-dependant ground cost that reflects the class-similarities between the samples; for unbalanced optimal transport, one could set the ground cost to 0 (or small values) to samples that are out of the distribution in order to have a more meaningful OT distance. An alternative formulation would be to learn the weights of the samples / learn the probability measures. In machine learning applications, they are often set to $1/n$. It would be interesting to learn a *barycentric* or a *quasi semi-discrete* representation of the dataset, in which regions of high density would be represented by samples with high weights and outliers would have low or null weights. One could also add *slack* variables in order to lower the importance of outliers.

9.2 Perspectives for Machine Learning on Time Series

9.2.1 Temporal transfer learning

When time series are at stake, new difficulties arise when transfering knowledge from one domain to another as temporal shifts can be encountered in addition to the standard feature distribution shift. It is nevertheless a task of crucial importance when dealing with time series as, by definition, the phenomenom may evolve with time, and that the underlying process that have been observed at some time may differ from another one that is observed at another time. While being able to match distributions is proved to be an efficient approach for transfer learning, existing methods mainly rely on domain adversarial neural network approaches, with tuning and parameter setting issues. One could couple optimal transport and DTW in a single formulation, in which the temporal distortion could vary w.r.t. the inputs or the labels for instance. In addition, beyond the classification task that have been considered, other

applications still lack some solutions, such that missing data imputation, clustering or even classification of temporal structured data (such as graphs or hierarchical representation of images) under domain shift. This problematic also encompasses the concept of early classification: while classification of time series is usually performed when the whole time series is observed, early classification deals with data stream. The aim is then to make predictions as early as possible (should I wait for other observations or can I make the prediction right now?). It is then a multi-objective optimization problem as a trade-off between earliness and accuracy should be performed. This idea is somehow concomitant to the task of time series prediction: if one is able to accurately predict (long-term) time series, one could also predict the class of the time series. The early classification task should be enriched by the significant advances in time series prediction.

9.2.2 Exploring the link between Dynamic Time Warping and Optimal Transport

DTW and OT share a lot of common features in their formulation: both are a linear programs under some constraints but the constraint set of the coupling (or alignment) matrix differ. One may wonder how developments that have been made about DTW can enrich OT and vice-versa. First, DTW can be efficiently solved relying on a dynamic programming algorithm, that may inspire efficient unbalanced OT-1d solutions as the order has to be maintained. On the other side, OT has many extensions that could be beneficial to DTW: how to define a unbalanced DTW algorithm, that would be able to get rid of outliers or take into account missing points? How defining a *multi-marginal* DTW, that would robustify the DTW measure when several time series are at stake? When dealing in a domain adaptation scenario, and despite their widespread use, DTW distances are limited to find a temporal alignment between two time series, but are not adapted to find an alignment between two sets of series. On the other hand, optimal transport is well suited to match different elements in sets, but fails to handle sets of time series that might not share the same common sampling, and is not robust to temporal shared common shifts between the time series. It would be interesting to combine best of both worlds by defining a metric that would map time series from distinct datasets under a global temporal shift by jointly optimizing an optimal transport map (matching series to series) and a dynamic time warping transformation (aligning timestamps).

9.3 Perspectives for Machine Learning for Remote Sensing images

9.3.1 Fully exploiting the geometry of hierarchical data with hyperbolic spaces

In most machine learning applications, the learning is performed on a Euclidean space, mostly because it has convenient mathematical properties, such as vectorial structures or closed forms for computing distances. Nevertheless, in many domains, real-world data do not possess a Euclidean structure but can rather be represented with a hierarchical structure and, in that case, they cannot be embedded in a Euclidean space with low distortion [Sala 2018]. In the opposite, hyperbolic spaces [Nickel 2017] are manifolds that have been shown to represent efficiently hierarchical data in many applications. With the Manal Hamzaoui's[1] PhD thesis, we aim to investigate whether the promises given by hyperbolic spaces can be fulfilled in a remote sensing data context, in particular on the hierarchically-labeled scene classification context. We first considered a hyperbolic variational auto-encoder for remote sensing scene classification [Hamzaoui 2021]: first experimental results show that the embedding in a hyperbolic space

[1] under progress

does not improve the global classification accuracy when compared with a Euclidean space, but allows slightly improving the deviation among the misclassified examples when taking into account the distance between the predicted and the actual label in the label hierarchy. We now are investigating the benefit of such spaces when considering a classification of scene images when the labels are organized on a tree, using a hierarchical loss. Up to now, those loss functions are derived from Euclidean ones and the geometry of the space is not fully exploited. One remains to define dedicated tools for optimizing on that geometry, with important promises in terms of accuracy improvements. For example, building on the Gromov-hyperbolicity property of those spaces, Gromov-Wasserstein metric is a particularly attractive tool that should allow a better organization of the latent space. In an other line of research, coupling these geometries in few or zero shot learning would perfectly fit the current remote sensing challenges, in which labels are often scarse and sometimes even unseen at the training stage.

9.3.2 Multimodality and transfer learning for Remote Sensing images

Beyond the exceptional data volume to be handled in remote sensing, modern remote sensing missions (aerial and satellite) has raised up new challenges for the remote sensing communities. These sensors are now able to offer (very) high spatial resolution images with revisit time frequencies never achieved before, considering different kind of signals, e.g., multi(hyper)-spectral optical, radar, LiDAR and digital surface models. In this context, pattern recognition and artificial intelligence techniques play a crucial role to deal with such an impressive amount of multi-source, multi-resolution and multi-temporal data. Nevertheless, labeled data are still scarse as they are costly to obtain, necessitate expertise and may not be available for all the modalities. The question of *heterogeneous domain adaptation*, that is to say when the domains are described by different features, is then of prime importance for the remote sensing community. Very few has been done in this direction, both in term of computational solutions and on theoretical analysis; beyond the remote sensing community, being able to transfer knowledge between different domains and across different architectures is one of the current main challenges in machine learning. My interest on optimal transport came from the idea that it can be an adequate tool for solving this problem; one of my works in the near future is to check this assumption on a remote sensing context.

Bibliography

[Absil 2009] P-A Absil, Robert Mahony and Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. Princeton University Press, 2009.

[Aggarwal 1992] Alok Aggarwal, Amotz Bar-Noy, Samir Khuller, Dina Kravets and Baruch Schieber. *Efficient minimum cost matching using quadrangle inequality.* In Annual Symposium on Foundations of Computer Science, volume 33, pages 583–583, 1992.

[Alonso González 2014] Alberto Alonso González. *Multidimensional and temporal SAR data representation and processing based on binary partition trees.* PhD thesis, Universitat Politècnica de Catalunya, 2014.

[Altschuler 2017] Jason Altschuler, Jonathan Niles-Weed and Philippe Rigollet. *Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration.* In Neural Information Processing Systems, volume 30, 2017.

[Alvarez-Melis 2018] David Alvarez-Melis and Tommi S Jaakkola. *Gromov-Wasserstein Alignment of Word Embedding Spaces.* In Conference on Empirical Methods in Natural Language Processing, 2018.

[Arjovsky 2017] Martin Arjovsky, Soumith Chintala and Léon Bottou. *Wasserstein generative adversarial networks.* In International Conference on Machine Learning, pages 214–223, 2017.

[Arvor 2011] Damien Arvor, Milton Jonathan, Margareth Simões Penello Meirelles, Vincent Dubreuil and Laurent Durieux. *Classification of MODIS EVI time series for crop mapping in the state of Mato Grosso, Brazil.* International Journal of Remote Sensing, vol. 32, no. 22, pages 7847–7871, 2011.

[Audebert 2016] Nicolas Audebert, Bertrand Le Saux and Sébastien Lefèvre. *Semantic segmentation of Earth Observation data using multimodal and multi-scale deep networks.* In Asian conference on computer vision, pages 180–196. Springer, 2016.

[Babai 2018] László Babai. *Group, graphs, algorithms: the graph isomorphism problem.* In International Congress of Mathematicians, pages 3319–3336, 2018.

[Backurs 2020] Arturs Backurs, Yihe Dong, Piotr Indyk, Ilya Razenshteyn and Tal Wagner. *Scalable nearest neighbor search for optimal transport.* In International Conference on Machine Learning, pages 497–506, 2020.

[Bagnall 2017] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large and Eamonn Keogh. *The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances.* Data mining and knowledge discovery, vol. 31, no. 3, pages 606–660, 2017.

[Bailly 2015a] Adeline Bailly, Simon Malinowski, Romain Tavenard, Laetitia Chapel and Thomas Guyet. *Dense bag-of-temporal-SIFT-words for time series classification.* In Advanced Analysis and Learning on Temporal Data, pages 17–30. Lecture Notes in Computer Science, vol 9785. Springer, 2015.

[Bailly 2015b] Adeline Bailly, Simon Malinowski, Romain Tavenard, Thomas Guyet and Laetitia Chapel. *Bag-of-temporal-sift-words for time series classification.* In ECML/PKDD workshop on advanced analytics and learning on temporal data, 2015.

[Bailly 2016] Adeline Bailly, Damien Arvor, Laetitia Chapel and Romain Tavenard. *Classification of MODIS time series with dense bag-of-temporal-SIFT-words: Application to cropland mapping in the Brazilian Amazon.* In IEEE International Geoscience and Remote Sensing Symposium, pages 2300–2303, 2016.

[Bailly 2017] Adeline Bailly, Laetitia Chapel, Romain Tavenard and Gustau Camps-Valls. *Nonlinear time-series adaptation for land cover classification.* IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 6, pages 896–900, 2017.

[Bailly 2018] Adeline Bailly. *Time Series Classification with Application to Remote Sensing.* PhD thesis, Université Rennes 2, 2018.

[Baisantry 2021] Munmun Baisantry, Anil K Sao and Dericks Praise Shukla. *Discriminative Spectral-Spatial Feature Extraction-based Band Selection for Hyperspectral Image Classification.* IEEE Transactions on Geoscience and Remote Sensing, 2021.

[Balaji 2020] Yogesh Balaji, Rama Chellappa and Soheil Feizi. *Robust Optimal Transport with Applications in Generative Modeling and Domain Adaptation.* Neural Information Processing Systems, 2020.

[Battaglia 2018] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro and Ryan *et al.* Faulkner. *Relational inductive biases, deep learning, and graph networks.* arXiv preprint arXiv:1806.01261, 2018.

[Beinert 2022] Robert Beinert, Cosmas Heiss and Gabriele Steidl. *On Assignment Problems Related to Gromov-Wasserstein Distances on the Real Line.* arXiv preprint arXiv:2205.09006, 2022.

[Bekker 2020] Jessa Bekker and Jesse Davis. *Learning from positive and unlabeled data: A survey.* Machine Learning, vol. 109, no. 4, pages 719–760, 2020.

[Bellemare 2017] Marc G Bellemare, Will Dabney and Rémi Munos. *A distributional perspective on reinforcement learning.* In International Conference on Machine Learning, pages 449–458, 2017.

[Benamou 2003] Jean-David Benamou. *Numerical resolution of an "unbalanced" mass transport problem.* ESAIM: Mathematical Modelling and Numerical Analysis, vol. 37, no. 5, pages 851–868, 2003.

[Benamou 2015] Jean-David Benamou, Guillaume Carlier, Marco Cuturi, Luca Nenna and Gabriel Peyré. *Iterative Bregman projections for regularized transportation problems.* SIAM Journal on Scientific Computing, vol. 37, no. 2, pages A1111–A1138, 2015.

[Bioucas-Dias 2013] José M Bioucas-Dias, Antonio Plaza, Gustavo Camps-Valls, Paul Scheunders, Nasser Nasrabadi and Jocelyn Chanussot. *Hyperspectral remote sensing data analysis and future challenges.* IEEE Geoscience and remote sensing magazine, vol. 1, no. 2, pages 6–36, 2013.

[Blaschke 2014] Thomas Blaschke, Geoffrey J Hay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, Freek Van der Meer, Harald Van der Werff, Frieke Van Coillie*et al.* *Geographic object-based image analysis–towards a new paradigm.* ISPRS journal of photogrammetry and remote sensing, vol. 87, pages 180–191, 2014.

[Blei 2006] David M Blei and John D Lafferty. *Dynamic topic models.* In International Conference on Machine Learning, pages 113–120, 2006.

[Blondel 2018] Mathieu Blondel, Vivien Seguy and Antoine Rolet. *Smooth and Sparse Optimal Transport.* In International Conference on Artificial Intelligence and Statistics, pages 880–889, 2018.

[Bo 2009] Liefeng Bo and Cristian Sminchisescu. *Efficient match kernel between sets of features for visual recognition.* Neural Information Processing Systems, vol. 22, pages 135–143, 2009.

[Bonneel 2015] Nicolas Bonneel, Julien Rabin, Gabriel Peyré and Hanspeter Pfister. *Sliced and radon wasserstein barycenters of measures.* Journal of Mathematical Imaging and Vision, vol. 51, no. 1, pages 22–45, 2015.

[Bonneel 2019] Nicolas Bonneel and David Coeurjolly. *SPOT: Sliced Partial Optimal Transport.* ACM Transactions on Graphics (SIGGRAPH), vol. 38, no. 4, 2019.

[Bonnotte 2013] Nicolas Bonnotte. *Unidimensional and evolution methods for optimal transportation.* PhD thesis, Paris 11, 2013.

[Brécheteau 2019] Claire Brécheteau. *A statistical test of isomorphism between metric-measure spaces using the distance-to-a-measure signature.* Electronic journal of statistics, vol. 13, no. 1, pages 795–849, 2019.

[Caffarelli 2010] Luis A Caffarelli and Robert J McCann. *Free boundaries in optimal transport and Monge-Ampere obstacle problems.* Annals of mathematics, pages 673–730, 2010.

[Cai 2018] Hongyun Cai, Vincent W Zheng and Kevin Chen-Chuan Chang. *A comprehensive survey of graph embedding: Problems, techniques, and applications.* IEEE Transactions on Knowledge and Data Engineering, vol. 30, no. 9, pages 1616–1637, 2018.

[Camps-Valls 2006] Gustavo Camps-Valls, Luis Gomez-Chova, Jordi Muñoz-Marí, Joan Vila-Francés and Javier Calpe-Maravilla. *Composite kernels for hyperspectral image classification.* IEEE geoscience and remote sensing letters, vol. 3, no. 1, pages 93–97, 2006.

[Cavallaro 2017] Gabriele Cavallaro, Nicola Falco, Mauro Dalla Mura and Jon Atli Benediktsson. *Automatic attribute profiles.* IEEE transactions on image processing, vol. 26, no. 4, pages 1859–1872, 2017.

[Chapel 2014] Laetitia Chapel, Thomas Burger, Nicolas Courty and Sébastien Lefèvre. *PerTurbo manifold learning algorithm for weakly labeled hyperspectral image classification.* IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 7, no. 4, pages 1070–1078, 2014.

[Chapel 2020] Laetitia Chapel, Mokhtar Z Alaya and Gilles Gasso. *Partial optimal tranport with applications on positive-unlabeled learning.* Neural Information Processing Systems, vol. 33, pages 2903–2913, 2020.

[Chapel 2021] Laetitia Chapel, Rémi Flamary, Haoran Wu, Cédric Févotte and Gilles Gasso. *Unbalanced Optimal Transport through Non-negative Penalized Linear Regression.* In Neural Information Processing Systems, 2021.

[Chen 2015] Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen and Gustavo Batista. *The UCR Time Series Classification Archive*, July 2015. www.cs.ucr.edu/~eamonn/time_series_data/.

[Chen 2018] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt and David Duvenaud. *Neural ordinary differential equations.* In Neural Information Processing Systems, pages 6572–6583, 2018.

[Chizat 2018a] Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer and François-Xavier Vialard. *Scaling algorithms for Unbalanced Optimal Transport problems.* Mathematics of Computation, vol. 87, no. 314, pages 2563–2609, 2018.

[Chizat 2018b] Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer and François-Xavier Vialard. *Unbalanced optimal transport: Dynamic and Kantorovich formulations.* Journal of Functional Analysis, vol. 274, no. 11, pages 3090–3123, 2018.

[Cho 2014] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau and Yoshua Bengio. *On the Properties of Neural Machine Translation: Encoder–Decoder Approaches.* In Workshop on Syntax, Semantics and Structure in Statistical Translation, pages 103–111, 2014.

[Chowdhury 2019] Samir Chowdhury and Facundo Mémoli. *The gromov–wasserstein distance between networks and stable network invariants.* Information and Inference, vol. 8, no. 4, pages 757–787, 2019.

[Christoffel 2017] Marthinus Christoffel, Gang Niu and Masashi Sugiyama. *Class-Prior Estimation for Learning from Positive and Unlabeled Data.* Machine Learning, vol. 106, no. 4, pages 463–492, 2017.

[Cohen 2021] Samuel Cohen, Giulia Luise, Alexander Terenin, Brandon Amos and Marc Deisenroth. *Aligning Time Series on Incomparable Spaces.* In International Conference on Artificial Intelligence and Statistics, pages 1036–1044, 2021.

[Courty 2011] Nicolas Courty, Thomas Burger and Johann Laurent. *PerTurbo: a new classification algorithm based on the spectrum perturbations of the Laplace-Beltrami operator.* In European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, pages 359–374. Springer, 2011.

[Courty 2016] Nicolas Courty, Rémi Flamary, Devis Tuia and Alain Rakotomamonjy. *Optimal transport for domain adaptation.* IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 9, pages 1853–1865, 2016.

[Cui 2015] Yanwei Cui, Laetitia Chapel and Sébastien Lefèvre. *A subpath kernel for learning hierarchical image representations.* In International Workshop on Graph-Based Representations in Pattern Recognition, pages 34–43. Springer, 2015.

[Cui 2016a] Yanwei Cui, Laetitia Chapel and Sébastien Lefèvre. *Combining multiscale features for classification of hyperspectral images: A sequence-based kernel approach.* In 8th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, pages 1–5. IEEE, 2016.

[Cui 2016b] Yanwei Cui, Sébastien Lefèvre, Laetitia Chapel and Anne Puissant. *Combining Multiple Resolutions into Hierarchical Representations for kernel-based Image Classification.* In international Conference on Geographic Object-Based Image Analysis (GEOBIA), 2016.

[Cui 2017a] Yanwei Cui. *Kernel-based learning on hierarchical image representations: applications to remote sensing data classification.* PhD thesis, Université Bretagne Sud, 2017.

[Cui 2017b] Yanwei Cui, Laetitia Chapel and Sébastien Lefèvre. *Scalable bag of subpaths kernel for learning on hierarchical image representations and multi-source remote sensing data classification.* Remote sensing, vol. 9, no. 3, page 196, 2017.

[Cuturi 2007] Marco Cuturi, Jean-Philippe Vert, Oystein Birkenes and Tomoko Matsui. *A kernel for time series based on global alignments.* In IEEE International Conference on Acoustics, Speech and Signal Processing, volume 2, pages II–413, 2007.

[Cuturi 2011] Marco Cuturi. *Fast global alignment kernels.* In International Conference on Machine Learning, pages 929–936, 2011.

[Cuturi 2013] Marco Cuturi. *Sinkhorn distances: Lightspeed computation of optimal transport.* Neural Information Processing Systems, vol. 26, pages 2292–2300, 2013.

[Cuturi 2014a] Marco Cuturi and David Avis. *Ground metric learning.* The Journal of Machine Learning Research, vol. 15, no. 1, pages 533–564, 2014.

[Cuturi 2014b] Marco Cuturi and Arnaud Doucet. *Fast computation of Wasserstein barycenters.* In International Conference on Machine Learning, pages 685–693, 2014.

[Cuturi 2017] Marco Cuturi and Mathieu Blondel. *Soft-DTW: a differentiable loss function for time-series.* In International Conference on Machine Learning, pages 894–903, 2017.

[Cuturi 2021] Marco Cuturi, Laetitia Papaxanthos and Olivier Teboul. *Optimal Transport Tools (OTT), A toolbox for everything Wasserstein*, 2021.

[Davidson 2018] Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf and Jakub M Tomczak. *Hyperspherical variational auto-encoders.* In Conference on Uncertainty in Artificial Intelligence, pages 856–865, 2018.

[Dechesne 2017] Clément Dechesne. *Segmentation sémantique de peuplement forestiers par analyse conjointe d'imagerie multispectrale très haute résolution et de données 3D Lidar aéroportées.* PhD thesis, Paris Est, 2017.

[Del Barrio 2019] Eustasio Del Barrio, Paula Gordaliza, Hélène Lescornel and Jean-Michel Loubes. *Central limit theorem and bootstrap procedure for Wasserstein's variations with an application to structural relationships between distributions.* Journal of Multivariate Analysis, vol. 169, pages 341–362, 2019.

[Demianov 1970] Vladimir Fedorovich Demianov and Aleksandr Moiseevich Rubinov. *Approximate methods in optimization problems.* vol. 53, 1970.

[Deng 2020] Huiqi Deng, Weifu Chen, Qi Shen, Andy J Ma, Pong C Yuen and Guocan Feng. *Invariant subspace learning for time series data based on dynamic time warping distance.* Pattern Recognition, vol. 102, page 107210, 2020.

[Dessein 2017] Arnaud Dessein, Nicolas Papadakis and Charles-Alban Deledalle. *Parameter estimation in finite mixture models by regularized optimal transport: A unified framework for hard and soft clustering.* arXiv preprint arXiv:1711.04366, 2017.

[Dhillon 2005] Inderjit S Dhillon and Suvrit Sra. *Generalized nonnegative matrix approximations with Bregman divergences.* In Neural Information Processing Systems, volume 18, 2005.

[Dhouib 2020] Sofien Dhouib, Ievgen Redko, Tanguy Kerdoncuff, Rémi Emonet and Marc Sebban. *A swiss army knife for minimax optimal transport.* In International Conference on Machine Learning, pages 2504–2513, 2020.

[Donahue 2014] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng and Trevor Darrell. *Decaf: A deep convolutional activation feature for generic visual recognition.* In International Conference on Machine Learning, pages 647–655, 2014.

[Du Plessis 2014] Marthinus C Du Plessis, Gang Niu and Masashi Sugiyama. *Analysis of learning from positive and unlabeled data.* In Neural Information Processing Systems, pages 703–711, 2014.

[Dumont 2022] Theo Dumont, Théo Lacombe and François-Xavier Vialard. *On The Existence Of Monge Maps For The Gromov-wasserstein Distance.* arXiv preprint arXiv:2210.11945, 2022.

[El Moselhy 2012] Tarek A El Moselhy and Youssef M Marzouk. *Bayesian inference with optimal maps.* Journal of Computational Physics, vol. 231, no. 23, pages 7815–7850, 2012.

[Elkan 2008] Charles Elkan and Keith Noto. *Learning classifiers from only positive and unlabeled data.* In ACM SIGKDD international conference on Knowledge discovery and data mining, pages 213–220, 2008.

[Fatras 2020] Kilian Fatras, Younes Zine, Rémi Flamary, Rémi Gribonval and Nicolas Courty. *Learning with minibatch Wasserstein: asymptotic and gradient properties.* In International Conference on Artificial Intelligence and Statistics, 2020.

[Fatras 2021] Kilian Fatras, Thibault Séjourné, Rémi Flamary and Nicolas Courty. *Unbalanced minibatch optimal transport; applications to domain adaptation.* In International Conference on Machine Learning, pages 3186–3197, 2021.

[Ferradans 2013] S. Ferradans, N. Papadakis, J. Rabin, G. Peyré and J. F. Aujol. *Regularized Discrete Optimal Transport*. In Scale Space and Variational Methods in Computer Vision, pages 428–439, 2013.

[Ferradans 2014] Sira Ferradans, Nicolas Papadakis, Gabriel Peyré and Jean-François Aujol. *Regularized discrete optimal transport*. SIAM Journal on Imaging Sciences, vol. 7, no. 3, pages 1853–1882, 2014.

[Févotte 2011] Cédric Févotte and Jérôme Idier. *Algorithms for nonnegative matrix factorization with the β-divergence*. Neural computation, vol. 23, no. 9, pages 2421–2456, 2011.

[Figalli 2010] Alessio Figalli. *The optimal partial transport problem*. Archive for rational mechanics and analysis, vol. 195, 2010.

[Flamary 2014] Rémi Flamary, Nicolas Courty, Alain Rakotomamonjy and Devis Tuia. *Optimal transport with Laplacian regularization*. In Workshop on Optimal Transport and Machine Learning at NeurIPS, 2014.

[Flamary 2021] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras and Nemo et al. Fournier. *POT: Python Optimal Transport*. Journal of Machine Learning Research, vol. 22, no. 78, pages 1–8, 2021.

[Frank 1956] Marguerite Frank and Philip Wolfe. *An algorithm for quadratic programming*. Naval Research Logistics Quarterly, vol. 3, no. 1-2, pages 95–110, 1956.

[Frogner 2015] Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya-Polo and Tomaso Poggio. *Learning with a Wasserstein Loss*. In Neural Information Processing Systems, pages 2053–2061, 2015.

[Genevay 2016] Aude Genevay, Marco Cuturi, Gabriel Peyré and Francis Bach. *Stochastic Optimization for Large-scale Optimal Transport*. In Neural Information Processing Systems, Barcelona, Spain, 2016.

[Genevay 2018] Aude Genevay, Gabriel Peyré and Marco Cuturi. *Learning generative models with sinkhorn divergences*. In International Conference on Artificial Intelligence and Statistics, pages 1608–1617, 2018.

[Ghamisi 2014] Pedram Ghamisi, Mauro Dalla Mura and Jon Atli Benediktsson. *A survey on spectral–spatial classification techniques based on attribute profiles*. IEEE Transactions on Geoscience and Remote Sensing, vol. 53, no. 5, pages 2335–2353, 2014.

[Gloaguen 2021] Pierre Gloaguen, Laetitia Chapel, Chloé Friguet and Romain Tavenard. *Scalable clustering of segmented trajectories within a continuous time framework: application to maritime traffic data*. Machine Learning, pages 1–27, 2021.

[Grabocka 2014] Josif Grabocka, Nicolas Schilling, Martin Wistuba and Lars Schmidt-Thieme. *Learning time-series shapelets*. In ACM SIGKDD international conference on Knowledge discovery and data mining, pages 392–401, 2014.

[Gramfort 2015] Alexandre Gramfort, Gabriel Peyré and Marco Cuturi. *Fast optimal transport averaging of neuroimaging data*. In International Conference on Information Processing in Medical Imaging, pages 261–272. Springer, 2015.

[Gretton 2012] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf and Alexander Smola. *A kernel two-sample test*. The Journal of Machine Learning Research, vol. 13, no. 1, pages 723–773, 2012.

[Gu 2018] Albert Gu, Frederic Sala, Beliz Gunel and Christopher Ré. *Learning mixed-curvature representations in product spaces.* In International Conference on Learning Representations, 2018.

[Guittet 2002] Kevin Guittet. *Extended Kantorovich norms: a tool for optimization.* Rapport technique, 2002.

[Hallin 2021] Marc Hallin, Eustasio Del Barrio, Juan Cuesta-Albertos and Carlos Matrán. *Distribution and quantile functions, ranks and signs in dimension d: A measure transportation approach.* The Annals of Statistics, vol. 49, no. 2, pages 1139–1165, 2021.

[Hamzaoui 2021] Manal Hamzaoui, Laetitia Chapel, Minh-Tan Pham and Sébastien Lefèvre. *Hyperbolic Variational Auto-Encoder for Remote Sensing Scene Classification.* In ORASIS, 2021.

[Haussler 1999] David Haussler. *Convolution kernels on discrete structures.* Rapport technique, Technical report, Department of Computer Science, University of California, 1999.

[Ho 2017] Nhat Ho, XuanLong Nguyen, Mikhail Yurochkin, Hung Hai Bui, Viet Huynh and Dinh Phung. *Multilevel Clustering via Wasserstein Means.* In International Conference on Machine Learning, volume 70, pages 1501–1509, 2017.

[Hochreiter 1997] Sepp Hochreiter and Jürgen Schmidhuber. *Long short-term memory.* Neural computation, vol. 9, no. 8, pages 1735–1780, 1997.

[Hsieh 2019] Yu-Guan Hsieh, Gang Niu and Masashi Sugiyama. *Classification from Positive, Unlabeled and Biased Negative Data.* In International Conference on Machine Learning, volume 97, pages 2820–2829, 2019.

[Hunter 2004] David R Hunter and Kenneth Lange. *A tutorial on MM algorithms.* The American Statistician, vol. 58, no. 1, pages 30–37, 2004.

[Ismail Fawaz 2019] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar and Pierre-Alain Muller. *Deep Neural Network Ensembles for Time Series Classification.* In 2019 International Joint Conference on Neural Networks (IJCNN), pages 1–6, 2019.

[Jain 2016] Shantanu Jain, Martha White, Michael W Trosset and Predrag Radivojac. *Nonparametric semi-supervised learning of class proportions.* Rapport technique, 2016.

[Jawanpuria 2020] Pratik Jawanpuria, NTV Dev and Bamdev Mishra. *Efficient robust optimal transport: formulations and algorithms.* In Workshop on Optimization for Machine Learning at NeurIPS, 2020.

[Jiang 2020] Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang and Silvia Chiappa. *Wasserstein fair classification.* In Uncertainty in Artificial Intelligence, pages 862–872, 2020.

[Kantorovich 1942] Leonid Kantorovich. *On the transfer of masses (in Russian).* Doklady Akademii Nauk, vol. 2, pages 227–229, 1942.

[Kantorovich 1957] Leonid Kantorovich and Gennadii Shlemovich Rubinshtein. *On a functional space and certain extremum problems.* In Doklady Akademii Nauk, volume 115, pages 1058–1061. Russian Academy of Sciences, 1957.

[Kato 2019] Masahiro Kato, Takeshi Teshima and Junya Honda. *Learning from positive and unlabeled data with a selection bias.* In International Conference on Learning Representations, 2019.

[Kerdoncuff 2021] Tanguy Kerdoncuff, Rémi Emonet and Marc Sebban. *Sampled Gromov Wasserstein.* machine learning, 2021.

[Kidger 2020] Patrick Kidger, James Morrill, James Foster and Terry J. Lyons. *Neural Controlled Differential Equations for Irregular Time Series.* In Neural Information Processing Systems, 2020.

[Kimura 2011] Daisuke Kimura, Tetsuji Kuboyama, Tetsuo Shibuya and Hisashi Kashima. *A subpath kernel for rooted unordered trees.* In Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 62–74. Springer, 2011.

[Korman 2015] Jonathan Korman and Robert McCann. *Optimal transportation with capacity constraints.* Transactions of the American Mathematical Society, vol. 367, no. 3, pages 1501–1521, 2015.

[Korotin 2020] Alexander Korotin, Vage Egiazarian, Arip Asadulaev, Alexander Safin and Evgeny Burnaev. *Wasserstein-2 Generative Networks.* In International Conference on Learning Representations, 2020.

[Kramer 2001] Stefan Kramer, Nada Lavrač and Peter Flach. *Propositionalization approaches to relational data mining.* Relational data mining, pages 262–291, 2001.

[Kriege 2016] Nils M. Kriege, Pierre-Louis Giscard and Richard C. Wilson. *On Valid Optimal Assignment Kernels and Applications to Graph Classification.* CoRR, vol. abs/1606.01141, 2016.

[Kullback 1951] Solomon Kullback and Richard A Leibler. *On information and sufficiency.* The annals of mathematical statistics, vol. 22, no. 1, pages 79–86, 1951.

[Kusner 2015] Matt Kusner, Yu Sun, Nicholas Kolkin and Kilian Weinberger. *From word embeddings to document distances.* In International Conference on Machine Learning, pages 957–966, 2015.

[Lacoste-Julien 2016] Simon Lacoste-Julien. *Convergence Rate of Frank-Wolfe for Non-Convex Objectives.* CoRR, vol. abs/1607.00345, 2016.

[Lefèvre 2014] Sébastien Lefèvre, Laetitia Chapel and François Merciol. *Hyperspectral image classification from multiscale description with constrained connectivity and metric learning.* In Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, pages 1–4, 2014.

[Levenshtein 1966] Vladimir I Levenshtein. *Binary codes capable of correcting deletions, insertions, and reversals.* In Soviet physics doklady, volume 10, pages 707–710. Soviet Union, 1966.

[Li 2019] Shutao Li, Weiwei Song, Leyuan Fang, Yushi Chen, Pedram Ghamisi and Jon Atli Benediktsson. *Deep learning for hyperspectral image classification: An overview.* IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 9, pages 6690–6709, 2019.

[Lin 2007] Jessica Lin, Eamonn Keogh, Li Wei and Stefano Lonardi. *Experiencing SAX: a novel symbolic representation of time series.* Data Mining and knowledge discovery, vol. 15, no. 2, pages 107–144, 2007.

[Lines 2012] Jason Lines, Luke M Davis, Jon Hills and Anthony Bagnall. *A shapelet transform for time series classification.* In ACM SIGKDD international conference on Knowledge discovery and data mining, pages 289–297, 2012.

[Lines 2018] Jason Lines, Sarah Taylor and Anthony Bagnall. *Time series classification with HIVE-COTE: The hierarchical vote collective of transformation-based ensembles.* ACM Transactions on Knowledge Discovery from Data, vol. 12, no. 5, 2018.

[Liu 1989] Dong C Liu and Jorge Nocedal. *On the limited memory BFGS method for large scale optimization.* Mathematical programming, vol. 45, no. 1, pages 503–528, 1989.

[Liu 2021] Qiao Liu and Hui Xue. *Adversarial Spectral Kernel Matching for Unsupervised Time Series Domain Adaptation.* In International Joint Conference on Artificial Intelligence, pages 2744–2750, 2021.

[Loossens 2021] Tim Loossens, Francis Tuerlinckx and Stijn Verdonck. *A comparison of continuous and discrete time modeling of affective processes in terms of predictive accuracy.* Scientific reports, vol. 11, no. 1, pages 1–11, 2021.

[Lowe 1999] David G Lowe. *Object recognition from local scale-invariant features.* In IEEE International Conference on Computer Vision, volume 2, pages 1150–1157, 1999.

[Luise 2018] Giulia Luise, Alessandro Rudi, Massimiliano Pontil and Carlo Ciliberto. *Differential Properties of Sinkhorn Approximation for Learning with Wasserstein Distance.* In Neural Information Processing Systems, volume 31, 2018.

[Lv 2019] Xianwei Lv, Dongping Ming, YangYang Chen and Min Wang. *Very high resolution remote sensing image classification with SEEDS-CNN and scale effect analysis for superpixel CNN classification.* International Journal of Remote Sensing, vol. 40, no. 2, pages 506–531, 2019.

[Mairal 2012] Julien Mairal and Bin Yu. *Complexity analysis of the lasso regularization path.* In International Conference on Machine Learning, 2012.

[Maretic 2019] Hermina Petric Maretic, Mireille EL Gheche, Giovanni Chierchia and Pascal Frossard. *GOT: An Optimal Transport framework for Graph comparison.* In Neural Information Processing Systems, volume 32, 2019.

[Massias 2018] Mathurin Massias, Alexandre Gramfort and Joseph Salmon. *Celer: a Fast Solver for the Lasso with Dual Extrapolation.* In International Conference on Machine Learning, volume 80, pages 3321–3330, 2018.

[Mémoli 2011] Facundo Mémoli. *Gromov–Wasserstein distances and the metric approach to object matching.* Foundations of computational mathematics, vol. 11, no. 4, pages 417–487, 2011.

[Mikolov 2013] Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. *Efficient estimation of word representations in vector space.* arXiv preprint arXiv:1301.3781, 2013.

[Monge 1781] Gaspard Monge. *Mémoire sur la théorie des déblais et des remblais.* Histoire de l'Académie Royale des Sciences de Paris, 1781.

[Mukherjee 2021] Debarghya Mukherjee, Aritra Guha, Justin M Solomon, Yuekai Sun and Mikhail Yurochkin. *Outlier-robust optimal transport.* In International Conference on Machine Learning, pages 7850–7860, 2021.

[Muzellec 2020] Boris Muzellec, Julie Josse, Claire Boyer and Marco Cuturi. *Missing data imputation using optimal transport.* In International Conference on Machine Learning, pages 7130–7140, 2020.

[Nadjahi 2019] Kimia Nadjahi, Alain Durmus, Umut Şimşekli and Roland Badeau. *Asymptotic guarantees for learning generative models with the sliced-wasserstein distance.* In Neural Information Processing Systems, pages 250–260, 2019.

[Nadjahi 2020] Kimia Nadjahi, Alain Durmus, Lénaïc Chizat, Soheil Kolouri, Shahin Shahrampour and Umut Simsekli. *Statistical and Topological Properties of Sliced Probability Divergences.* Neural Information Processing Systems, vol. 33, 2020.

[Ng 2001] Andrew Ng and Michael Jordan. *On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes.* Neural Information Processing Systems, vol. 14, 2001.

[Nickel 2017] Maximillian Nickel and Douwe Kiela. *Poincaré embeddings for learning hierarchical representations.* In Neural Information Processing Systems, volume 30, pages 6338–6347, 2017.

[Niepert 2016] Mathias Niepert, Mohamed Ahmed and Konstantin Kutzkov. *Learning convolutional neural networks for graphs.* In International Conference on Machine Learning, pages 2014–2023, 2016.

[Niles-Weed 2019] Jonathan Niles-Weed and Philippe Rigollet. *Estimation of Wasserstein distances in the Spiked Transport Model.* arXiv e-prints, pages arXiv–1909, 2019.

[Oliver 2018] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk and Ian Goodfellow. *Realistic Evaluation of Deep Semi-Supervised Learning Algorithms*. Neural Information Processing Systems, vol. 31, pages 3235–3246, 2018.

[Painblanc 2019] François Painblanc. *Neural Ordinary Differential Equation model*, Master thesis, 2019.

[Paty 2019] François-Pierre Paty and Marco Cuturi. *Subspace robust Wasserstein distances*. In International Conference on Machine Learning, pages 5072–5081, 2019.

[Paty 2020] François-Pierre Paty and Marco Cuturi. *Regularized optimal transport is ground cost adversarial*. In International Conference on Machine Learning, pages 7532–7542, 2020.

[Pedregosa 2011] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg*et al.* *Scikit-learn: Machine learning in Python*. Journal of Machine Learning Research, vol. 12, pages 2825–2830, 2011.

[Pele 2009] Ofir Pele and Michael Werman. *Fast and robust Earth mover's distances*. In IEEE International Conference on Computer Vision, pages 460–467, 2009.

[Pelletier 2019] Charlotte Pelletier, Geoffrey I Webb and François Petitjean. *Temporal convolutional neural network for the classification of satellite image time series*. Remote Sensing, vol. 11, no. 5, page 523, 2019.

[Pennington 2014] Jeffrey Pennington, Richard Socher and Christopher D Manning. *Glove: Global vectors for word representation*. In Conference on Empirical Methods in Natural Language Processing, pages 1532–1543, 2014.

[Peyré 2016] Gabriel Peyré, Marco Cuturi and Justin Solomon. *Gromov-Wasserstein Averaging of Kernel and Distance Matrices*. In International Conference on Machine Learning, volume 48, pages 2664–2672, 2016.

[Peyré 2019] Gabriel Peyré and Marco Cuturi. *Computational optimal transport: With applications to data science*. Foundations and Trends in Machine Learning, vol. 11, no. 5-6, pages 355–607, 2019.

[Pires de Lima 2020] Rafael Pires de Lima and Kurt Marfurt. *Convolutional neural network for remote-sensing scene classification: Transfer learning analysis*. Remote Sensing, vol. 12, no. 1, page 86, 2020.

[Rabin 2011] Julien Rabin, Gabriel Peyré, Julie Delon and Marc Bernot. *Wasserstein barycenter and its application to texture mixing*. In International Conference on Scale Space and Variational Methods in Computer Vision, pages 435–446. Springer, 2011.

[Rabin 2014] Julien Rabin, Sira Ferradans and Nicolas Papadakis. *Adaptive color transfer with relaxed optimal transport*. In 2014 IEEE International Conference on Image Processing (ICIP), pages 4852–4856. IEEE, 2014.

[Rahimi 2007] Ali Rahimi and Benjamin Recht. *Random Features for Large-Scale Kernel Machines*. In Neural Information Processing Systems, volume 3, page 5, 2007.

[Redko 2017] Ievgen Redko, Amaury Habrard and Marc Sebban. *Theoretical analysis of domain adaptation with optimal transport*. In European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, pages 737–753. Springer, 2017.

[Redko 2020] Ievgen Redko, Titouan Vayer, Rémi Flamary and Nicolas Courty. *CO-Optimal Transport*. In Neural Information Processing Systems, 2020.

[Ribeiro 2017] Leonardo FR Ribeiro, Pedro HP Saverese and Daniel R Figueiredo. *struc2vec: Learning node representations from structural identity*. In ACM SIGKDD international conference on knowledge discovery and data mining, pages 385–394, 2017.

[Roweis 2000] Sam T Roweis and Lawrence K Saul. *Nonlinear dimensionality reduction by locally linear embedding*. science, vol. 290, no. 5500, pages 2323–2326, 2000.

[Rubanova 2019] Yulia Rubanova, Ricky TQ Chen and David Duvenaud. *Latent ODEs for irregularly-sampled time series*. arXiv preprint arXiv:1907.03907, 2019.

[Rubner 2000] Yossi Rubner, Carlo Tomasi and Leonidas J Guibas. *The earth mover's distance as a metric for image retrieval*. International journal of computer vision, vol. 40, no. 2, pages 99–121, 2000.

[Saenko 2010] Kate Saenko, Brian Kulis, Mario Fritz and Trevor Darrell. *Adapting visual category models to new domains*. In European Conference on Computer Vision, pages 213–226, 2010.

[Sakoe 1978] Hiroaki Sakoe and Seibi Chiba. *Dynamic programming algorithm optimization for spoken word recognition*. IEEE transactions on acoustics, speech, and signal processing, vol. 26, no. 1, pages 43–49, 1978.

[Sala 2018] Frederic Sala, Chris De Sa, Albert Gu and Christopher Ré. *Representation tradeoffs for hyperbolic embeddings*. In International conference on machine learning, pages 4460–4469, 2018.

[Santambrogio 2015] Filippo Santambrogio. *Optimal transport for applied mathematicians*. Birkäuser, NY, vol. 55, no. 58-63, page 94, 2015.

[Santana Maia 2021] Deise Santana Maia, Minh-Tan Pham, Erchan Aptoula, Florent Guiotte and Sébastien Lefèvre. *Classification of remote sensing data with morphological attributes profiles: a decade of advances*. IEEE geoscience and remote sensing magazine, 2021.

[Schmitt 2016] Michael Schmitt and Xiao Xiang Zhu. *Data fusion and remote sensing: An ever-growing relationship*. IEEE Geoscience and Remote Sensing Magazine, vol. 4, no. 4, pages 6–23, 2016.

[Sejourne 2021] Thibault Sejourne, Francois-Xavier Vialard and Gabriel Peyré. *Faster Unbalanced Optimal Transport: Translation invariant Sinkhorn and 1-D Frank-Wolfe*. In Optimal Transport and Machine Learning Workshop at NeurIPS, 2021.

[Shervashidze 2011] Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn and Karsten M. Borgwardt. *Weisfeiler-Lehman Graph Kernels*. Journal of Machine Learning Research, vol. 12, pages 2539–2561, Novembre 2011.

[Si Salah 2020] Hayet Si Salah, Sally E Goldin, Abdelmounaam Rezgui, Bachari Nour El Islam and Samy Ait-Aoudia. *What is a remote sensing change detection technique? Towards a conceptual framework*. International Journal of Remote Sensing, vol. 41, no. 5, pages 1788–1812, 2020.

[Sinkhorn 1967] Richard Sinkhorn and Paul Knopp. *Concerning nonnegative matrices and doubly stochastic matrices*. Pacific Journal of Mathematics, vol. 21, no. 2, pages 343–348, 1967.

[Sivic 2008] Josef Sivic and Andrew Zisserman. *Efficient visual search of videos cast as text retrieval*. IEEE transactions on pattern analysis and machine intelligence, vol. 31, no. 4, pages 591–606, 2008.

[Sommerfeld 2019] Max Sommerfeld, Jörn Schrieber, Yoav Zemel and Axel Munk. *Optimal Transport: Fast Probabilistic Approximation with Exact Solvers*. J. Mach. Learn. Res., vol. 20, pages 105–1, 2019.

[Sun 2017] Ying Sun, Prabhu Babu and Daniel P. Palomar. *Majorization-minimization algorithms in signal processing, communications, and machine learning*. IEEE Transactions on Signal Processing, vol. 65, no. 3, pages 794–816, 2017.

[Taskin 2021] Gulsen Taskin and Gustau Camps-Valls. *Graph Embedding via High Dimensional Model Representation for Hyperspectral Images*. arXiv preprint arXiv:2111.14680, 2021.

[Tavenard 2017] Romain Tavenard, Simon Malinowski, Laetitia Chapel, Adeline Bailly, Heider Sanchez and Benjamin Bustos. *Efficient temporal kernels between feature sets for time series classification.* In European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, pages 528–543. Springer, 2017.

[Ten Holt 2007] Gineke A Ten Holt, Marcel JT Reinders and Emile A Hendriks. *Multi-dimensional dynamic time warping for gesture recognition.* In Thirteenth annual conference of the Advanced School for Computing and Imaging, volume 300, page 1, 2007.

[Tenenbaum 2000] Joshua B Tenenbaum, Vin De Silva and John C Langford. *A global geometric framework for nonlinear dimensionality reduction.* Science, vol. 290, no. 5500, pages 2319–2323, 2000.

[Thibault 2021] Alexis Thibault, Lénaïc Chizat, Charles Dossal and Nicolas Papadakis. *Overrelaxed Sinkhorn–Knopp Algorithm for Regularized Optimal Transport.* Algorithms, vol. 14, no. 5, page 143, 2021.

[Tochon 2015] Guillaume Tochon. *Analyse hiérarchique d'images multimodales.* PhD thesis, Université Grenoble Alpes (ComUE), 2015.

[Trigeorgis 2016] George Trigeorgis, Mihalis A Nicolaou, Stefanos Zafeiriou and Bjorn W Schuller. *Deep canonical time warping.* In IEEE Conference on Computer Vision and Pattern Recognition, pages 5110–5118, 2016.

[Tuia 2016] Devis Tuia and Gustau Camps-Valls. *Kernel manifold alignment for domain adaptation.* PloS one, vol. 11, no. 2, page e0148655, 2016.

[Tuna 2020] Çaglayan Tuna. *Morphological Hierarchies for Satellite Image Time Series.* PhD thesis, Université Bretagne Sud, 2020.

[Uhlenbeck 1930] George E Uhlenbeck and Leonard S Ornstein. *On the theory of the Brownian motion.* Physical review, vol. 36, no. 5, page 823, 1930.

[Valero 2011] Silvia Valero, Philippe Salembier, Jocelyn Chanussot and Carles M Cuadras. *Improved binary partition tree construction for hyperspectral images: Application to object detection.* In 2011 IEEE International Geoscience and Remote Sensing Symposium, pages 2515–2518. IEEE, 2011.

[Vayer 2019a] Titouan Vayer, Laetitia Chapel, Rémi Flamary, Romain Tavenard and Nicolas Courty. *Optimal Transport for structured data with application on graphs.* In International Conference on Machine Learning, volume 97, pages 6275–6284, 2019.

[Vayer 2019b] Titouan Vayer, Rémi Flamary, Romain Tavenard, Laetitia Chapel and Nicolas Courty. *Sliced Gromov-Wasserstein.* In Neural Information Processing Systems, volume 32, 2019.

[Vayer 2020a] Titouan Vayer. *A contribution to Optimal Transport on incomparable spaces.* PhD thesis, Université Bretagne Sud, 2020.

[Vayer 2020b] Titouan Vayer, Laetitia Chapel, Nicolas Courty, Rémi Flamary, Yann Soullard and Romain Tavenard. *Time series alignment with global invariances.* arXiv preprint arXiv:2002.03848, 2020.

[Vayer 2020c] Titouan Vayer, Laetitia Chapel, Rémi Flamary, Romain Tavenard and Nicolas Courty. *Fused Gromov-Wasserstein Distance for Structured Objects.* Algorithms, vol. 13, no. 9, 2020.

[Villani 2009] Cédric Villani. Optimal transport: old and new, volume 338. Springer, 2009.

[Virtanen 2020] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero,

Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt and SciPy 1.0 Contributors. *SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python.* Nature Methods, vol. 17, pages 261–272, 2020.

[Vishwanathan 2004] SVN Vishwanathan and Alexander Smola. *Fast kernels for string and tree matching.* Kernel methods in computational biology, vol. 15, pages 113–130, 2004.

[Vishwanathan 2010] S Vichy N Vishwanathan, Nicol N Schraudolph, Risi Kondor and Karsten M Borgwardt. *Graph kernels.* Journal of Machine Learning Research, vol. 11, pages 1201–1242, 2010.

[Voreiter 2020] Claire Voreiter, Jean-Christophe Burnel, Pierre Lassalle, Marc Spigai, Romain Hugues and Nicolas Courty. *A cycle GAN approach for heterogeneous domain adaptation in land use classification.* In IEEE International Geoscience and Remote Sensing Symposium, pages 1961–1964, 2020.

[Wang 2006] Xuerui Wang and Andrew McCallum. *Topics over time: a non-markov continuous-time model of topical trends.* In ACM SIGKDD international conference on Knowledge discovery and data mining, pages 424–433, 2006.

[Wang 2008] Chong Wang, David Blei and David Heckerman. *Continuous time dynamic topic models.* In Conference on Uncertainty in Artificial Intelligence, pages 579–586, 2008.

[Wilson 2020] Garrett Wilson, Janardhan Rao Doppa and Diane J Cook. *Multi-source deep domain adaptation with weak supervision for time-series sensor data.* In ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pages 1768–1778, 2020.

[Yalniz 2019] I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri and Dhruv Mahajan. *Billion-scale semi-supervised learning for image classification.* arXiv, 2019.

[Yang 2008] Yi Yang and Shawn Newsam. *Comparing SIFT descriptors and Gabor texture features for classification of remote sensed imagery.* In 2008 15th IEEE international conference on image processing, pages 1852–1855. IEEE, 2008.

[Yeh 2014] Yi-Ren Yeh, Chun-Hao Huang and Yu-Chiang Frank Wang. *Heterogeneous domain adaptation and classification by exploiting the correlation subspace.* IEEE Transactions on Image Processing, vol. 23, no. 5, pages 2009–2018, 2014.

[Yuan 2015] Yuan Yuan, Lichao Mou and Xiaoqiang Lu. *Scene recognition by manifold regularized deep learning architecture.* IEEE transactions on neural networks and learning systems, vol. 26, no. 10, pages 2222–2233, 2015.

[Zeiberg 2020] Daniel Zeiberg, Shantanu Jain and Predrag Radivojac. *Fast Nonparametric Estimation of Class Proportions in the Positive-Unlabeled Classification Setting.* In AAAI Conference on Artificial Intelligence, volume 34, pages 6729–6736, 2020.

[Zhang 2017] Zheng Zhang, Romain Tavenard, Adeline Bailly, Xiaotong Tang, Ping Tang and Thomas Corpetti. *Dynamic time warping under limited warping path length.* Information Sciences, vol. 393, pages 91–107, 2017.

[Zhang 2019] Wei Zhang, Ping Tang and Lijun Zhao. *Remote sensing image scene classification using CNN-CapsNet.* Remote Sensing, vol. 11, no. 5, page 494, 2019.

[Zhang 2020] Steven Zhang, David Widmann and Davi Barreira. *Optimal Transport in Julia*, 2020.

[Zhang 2021] Pei Zhang, Yunpeng Bai, Dong Wang, Bendu Bai and Ying Li. *Few-shot classification of aerial scene images via meta-learning.* Remote Sensing, vol. 13, no. 1, page 108, 2021.

[Zhou 2009] Feng Zhou and Fernando Torre. *Canonical time warping for alignment of human behavior.* Neural Information Processing Systems, vol. 22, pages 2286–2294, 2009.