

Méthodes variationnelles pour les problèmes de données à grande échelle en imagerie.

Variational methods for large-scale data problems in imaging.

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°580 Sciences et Technologie de l'Information et de la Communication
(STIC)

Spécialité de doctorat: Sciences du Traitement du signal et des images

Graduate School : Sciences de l'ingénierie et des systèmes.

Référent : CentraleSupélec

Thèse préparée dans l'unité de recherche

Centre de la Vision Numérique (Université Paris-Saclay, CentraleSupélec),
sous la direction de

Jean-Christophe PESQUET, Professeur, Université Paris-Saclay, CentraleSupélec, Inria, CVN,
et **Ismail BEN AYED**, Professeur, ETS Montreal.

Thèse soutenue à Paris-Saclay, le 26 janvier 2024, par

Ségolène MARTIN

Composition du jury

Membres du jury avec voix délibérative

Aleksandra PIZURICA	Rapporteur
Professeure, TELIN-GAIM, Ghent University (Ghent, Belgique).	
Nelly PUSTELNIK	Rapporteur
Directrice de recherche, Laboratoire de Physique, ENS Lyon, CNRS UMR 5672 (Lyon, France).	
François MALGOUYRES	Examinateur
Professeur, Institut de Mathématiques de Toulouse, Université de Toulouse, CNRS UMR 5219 (Toulouse, France).	
Gabriele STEIDL	Examinateur
Professeure, Institute of Mathematics, TU Berlin (Berlin, Allemagne).	
Nicolas THOME	Examinateur
Professeur, MLIA, ISIR, Sorbonne Université (Paris, France).	

Titre: Méthodes variationnelles pour les problèmes de données à grande échelle en imagerie

Mots clés: optimisation grande-échelle, problèmes inverses, restauration d'image, classification few-shot, apprentissage transductif, algorithmes de majoration-minimisation

Résumé: Cette thèse explore l'utilisation de méthodes variationnelles pour aborder les problèmes d'imagerie de grande échelle, tant en termes de dimensionnalité qu'en nombre de données. Elle met un accent particulier sur les problèmes inverses en restauration d'image et la classification en régime dit "few-shot", i.e., avec peu de données labelisées. Le fil conducteur entre les sujets abordés est le développement de méthodes et d'algorithmes à la fois performants, fiables, interprétables et efficaces via des outils d'optimisation avancés.

Dans la première partie de la thèse, nous nous intéressons à la résolution de problèmes inverses de restauration d'image à partir d'une unique observation dégradée de grande taille. Un algorithme innovant de Majoration-Minimisation locale par sous-espace, couplé avec une méthode de pénalisation, est introduit pour l'optimisation lisse et sous contrainte. Des garanties théoriques de convergence sont données et l'efficacité, en terme de temps de convergence, est démontrée sur plusieurs exemples applicatifs. Toujours sur la thématique de la restauration, une étude applicative complète est conduite dans le cadre d'images tridimensionnelles obtenues par microscopie multiphoton, allant de l'estimation de l'opérateur de flou (Point Spread Function ou PSF) jusqu'à la déconvolution. Pour l'étape d'estimation de la PSF, nous proposons un nouveau modèle de

calibration basé sur des billes de large diamètre, ainsi qu'un algorithme proximal alterné. Pour la phase de restauration, nous adoptons une formulation contrainte sans paramètre de régularisation, en supposant un bruit gaussien hétéroscédastique.

La seconde partie de la thèse s'intéresse à la classification faiblement supervisée d'images en utilisant des modèles neuronaux pré-entraînés sur d'autres bases de données. Une attention particulière est accordée au scénario transductif, où les prédictions sont effectuées sur un ensemble de données non labelisées plutôt qu'individuellement. L'approche présentée dans cette thèse aborde le problème de la classification "few-shot" en présence d'un déséquilibre de classe dans les données, en mettant en œuvre un critère pénalisé et en effectuant la minimisation par un algorithme alterné efficace. Dans le prolongement de ce travail, nous explorons la classification "zero-shot" et "few-shot" en utilisant des modèles vision-langage qui combinent des représentations visuelles et textuelles, tels que CLIP. Nous présentons la première approche transductive dans ce domaine en utilisant les vecteurs de probabilités fournis par CLIP comme entrées de notre classifieur. Ceci conduit à un problème original de minimisation sur un simplexe. Nous démontrons que des gains de performance significatifs peuvent être obtenus dans le scénario transductif.

Title: Variational methods for large-scale data problems in imaging

Keywords: large-scale optimisation, inverse problems, image restoration, few-shot classification, transductive inference, majorization-minimization algorithms

Abstract: This thesis is dedicated to exploring the use of variational methods for addressing large-scale imaging problems, both in terms of data dimensionality and quantity. It focuses on inverse problems in image restoration and classification within the so-called few-shot regime, i.e., with a limited number of labeled data. The common goal across the addressed topics is the design of methods and algorithms that are efficient, reliable, interpretable, and effective, utilizing advanced optimization tools.

In the first part of the thesis, we are interested in solving inverse image restoration problems starting from a single, large, degraded observation. An innovative local Majorization-Minimization subspace algorithm, coupled with a penalization method, is introduced for smooth and constrained optimization. Theoretical convergence guarantees are provided, and the efficiency, in terms of convergence time, is demonstrated across several applicative examples. In the continuation of this work on restoration, a detailed applicative study is conducted in the context of three-dimensional images obtained by multiphoton microscopy, ranging from the estimation of the Point Spread Function (PSF) operator to deconvolution. For the PSF estimation step, we

propose a new calibration model based on large diameter beads, along with an alternating proximal algorithm. For the restoration phase, we adopt a constrained formulation with no regularization parameter, assuming heteroscedastic Gaussian noise.

The second part of the thesis investigates weakly supervised image classification using neural models pre-trained on other databases. Particular attention is paid to the transductive scenario, where predictions are made on a set of unlabeled data rather than individually. In this context, the first problem tackled in this thesis is few-shot classification in the presence of class imbalance in the data. A novel penalized criterion is proposed and an effective alternating algorithm is developed to minimize this criterion. In a further study, we explore zero-shot and few-shot classification using vision-language models that combine visual and textual representations, like CLIP. We present the first transductive approach in this area using probability vectors provided by CLIP as inputs to our classifier. This leads to a an original minimization problem on a simplex. We demonstrate significant performance gains that can be achieved in the transductive scenario.

Long résumé

Au cours des deux dernières décennies, l'évolution numérique des systèmes d'imagerie a conduit à une augmentation sans précédent de données de grande dimension. Cela est particulièrement évident dans l'imagerie médicale et satellitaire, où chaque image 2D ou 3D peut représenter plusieurs gigaoctets. La dimension peut également se référer au nombre important d'images accumulées, comme c'est le cas sur nos smartphones personnels. Ces nouvelles données à grande échelle soulèvent des défis en vision par ordinateur, tels que la reconstruction d'images, la détection d'objets ou d'anomalies, la classification, la segmentation, etc. Il devient alors nécessaire de développer des méthodes permettant de traiter ces grandes masses de données. Cette thèse vise à relever ces défis en exploitant des méthodes variationnelles et des techniques d'optimisation avancées pour développer des solutions algorithmiques pour deux tâches d'imagerie à grande échelle spécifiques : la restauration d'images du point de vue du problème inverse et les tâches de classification (en particulier dans le régime de few-shot). Lors de l'élaboration de ces méthodes, plusieurs points sont à considérer. Tout d'abord, la rapidité de l'algorithme d'optimisation sera au cœur de cette thèse. De grandes données impliquent des temps de calcul importants avec des méthodes traditionnelles. Ensuite, la question des garanties théoriques de convergence vers un minimiseur (si possible global) d'une fonction de coût donnée est importante, car cela peut éviter des comportements trop variables de la méthode en fonction de l'initialisation, des données, du bruit, et garantir que l'algorithme produit en un temps fini une solution cohérente avec ce qui a été demandé. Finalement, la question de la scalabilité des méthodes employées devient urgente. Face aux temps et coûts pharamineux nécessaires au développement de solutions d'apprentissage supervisé, il n'est plus raisonnable de réentraîner les modèles à chaque nouvel ajout de données et/ou changement de la source des données. Un paradigme beaucoup développé ces dernières années consiste à s'orienter vers l'apprentissage de représentations des données plutôt que l'apprentissage spécifique à des tâches. Les représentations peuvent ensuite être utilisées pour diverses applications. En combinant des perspectives théoriques avec des stratégies algorithmiques pratiques, la thèse contribue au domaine plus large de l'imagerie computationnelle, offrant de nouvelles perspectives et outils pour exploiter pleinement le potentiel des données d'image dans diverses applications scientifiques et industrielles.

Un premier travail introduit l'algorithme Penalized Majorization-Minimization Subspace (P-MMS), une approche innovante pour les problèmes d'optimisation lisses et contraints, courants dans la restauration d'images. C'est l'une des premières approches Majoration-Minimisation (MM) pour l'optimisation contrainte. Concrètement, l'approche P-MMS consiste à incorporer un algorithme de sous-espace dans une procédure de pénalité extérieure inexacte, exploitant l'avantage de la lissité de la fonction de coût pénalisée tout en gérant efficacement une diversité de contraintes. Le principal inconvénient des méth-

odes de pénalité extérieure, lié au mauvais conditionnement pour les grandes valeurs du paramètre de pénalité, est résolu grâce à une technique inspirée des régions de confiance. L’analyse de convergence de l’algorithme proposé est approfondie, et les expériences numériques sur deux applications de restauration d’images à grande échelle confirment ses performances en termes de rapidité par rapport aux algorithmes existants.

Un travail dans la continuité du premier explore une approche variationnelle innovante pour la restauration d’images en microscopie multiphoton (MPM), commençant par l’estimation de la fonction d’étalement du point (PSF) et aboutissant à la déconvolution 3D. Nous estimons la PSF à partir de l’observation de billes relativement grandes (d’environ $1\mu\text{m}$ de diamètre). Cela passe par la formulation et la résolution d’un problème de minimisation non convexe original, pour lequel nous proposons une méthode alternée proximale avec des garanties de convergence. Durant la phase de restauration d’image, nous nous appuyons sur un modèle de bruit hétéroscédastique pour lequel nous estimons les paramètres. Nous résolvons ensuite un problème d’optimisation contraint pour restaurer l’image, en tenant compte de la PSF et du bruit estimés, tout en permettant un réglage minimal des hyper-paramètres. Des garanties théoriques sont données pour l’algorithme de restauration. Ces contributions algorithmiques mènent à une pipeline complète pour la restauration d’images 3D en MPM. Nous démontrons son efficacité à travers plusieurs expériences sur des données simulées et réelles.

La seconde partie de la thèse est consacrée à la classification dans un régime “few-shot”, c’est à dire dans une situation où le nombre de données labélisées est faible. Chaque données labelisée est appelée un “shot” et l’ensemble des shots constitue l’ensemble “support”. L’ensemble “query” désigne quant à lui les échantillons dont on veut prédire les classes. L’un des chapitres de cette thèse explore une nouvelle approche de l’apprentissage transductif en few-shot dans un cadre pratique réaliste. Les benchmarks habituels en few-shot reposent souvent sur des suppositions simplificatrices qui ne tiennent pas compte de la variabilité réelle de l’ensemble de “query”. Dans notre approche, ces hypothèses sont relaxées: les classes présentes dans le “query” sont inconnues et font partie d’un ensemble bien plus vaste de classes potentielles. Cette configuration s’avère être un cas pratique de la classification déséquilibrée. Face à cette configuration, les performances des méthodes de l’état de l’art chutent. Pour addresser ce problème, nous abordons le problème du few-shot sous l’angle variationnel, en introduisant un terme de pénalité inspiré par le principe du *Minimum Description Length*. Cette stratégie nous permet de gérer efficacement le déséquilibre des classes. De plus, un algorithme de descente en bloc est proposé. L’algorithme est rapide, avec une complexité linéaire à chaque itération, et des garanties de convergence sont données. Des tests sur des base de données standard en few-shot démontrent les performances compétitives de notre méthode.

Finalement, le dernier chapitre de cette thèse est une première dans l’application des techniques de classification transductive au domaine émergent des modèles vision-langage, en particulier pour les défis de classification zero-shot et few-shot. Ce travail présente une nouvelle méthodologie pour générer des vecteurs de probabilité vision-texte, établissant ainsi une base solide pour une approche de classification transductive few-shot spécialement adaptée aux modèles vision-language. En s’inspirant de l’algorithme Expectation-Maximization (EM) et en modelant la distribution des données avec des lois de Dirichlet, nous introduisons un nouvel algorithme de Majorization-Minimization pour estimer conjointement les prédictions et les paramètres des distributions. Une série d’expériences approfondies sur 11 datasets de

classification. Les résultats sur des tâches zero-shot avec des lots de test de 75 échantillons montrent une amélioration de la précision sur ImageNet de près de 20 % par rapport à la performance zero-shot de CLIP, et notre méthode dépasse les méthodes de pointe dans le contexte du few-shot, confirmant l'efficacité de notre approche d'inférence en groupe.

Remerciements

Je remercie chaleureusement mes directeurs de thèse, Jean-Christophe Pesquet et Ismail Ben Ayed, pour leurs précieux conseils prodigués durant ces années de thèse. Au-delà de la quantité épataante de bonnes idées que Jean-Christophe a insufflées à mon travail, j'ai apprécié sa rigueur et sa méthodologie, qui continueront de me servir pour la suite de ma carrière. Sa gestion habile de mon pessimisme, souvent tenace, mérite une mention spéciale. Quant à Ismail, son enthousiasme contagieux et sa patience ont été cruciaux pour que nos deux domaines, avec leurs vocabulaires spécifiques, se rencontrent. Grâce à lui, j'ai aussi eu la chance de vivre une expérience de recherche unique à Montréal. Ensemble, ils ont orienté ma recherche vers des thématiques modernes et totalement nouvelles pour moi. Il y a quatre ans, l'idée même d'effleurer un réseau de neurones durant ma thèse aurait provoqué mon scepticisme (le mot tabou pour quelqu'un qui a fait des études de mathématiques). Aujourd'hui, je me réjouis de mon profil hybride.

Je tiens à exprimer ma gratitude aux membres de mon jury, pour le temps et l'attention qu'ils ont consacrés à l'évaluation de ma thèse. Je remercie le président du jury François Malgouyres, les rapporteurs Aleksandra Pizurica et Nelly Pustelnik pour leurs remarques constructives sur mon manuscrit, ainsi Gabriele Steidl et Nicolas Thome.

Durant ma thèse, j'ai eu la chance de collaborer avec plusieurs chercheurs. Un grand merci à Emilie Chouzenoux pour son aide précieuse sur plusieurs articles et pour ses relectures minutieuses. Je suis aussi très reconnaissante envers Claire Lefort pour son enthousiasme et sa bienveillance, qui ont grandement contribué au succès de notre projet en microscopie. Je n'oublie pas les autres doctorants et post-doctorants avec qui j'ai travaillé : merci à Malik Boudiaf, Aymen Sadraoui, Yunshi Yuang, Fereshteh Shakeri ; ainsi que les stagiaires que j'ai eu l'honneur d'encadrer : Yannick Sun, Julien Ajdenbaum, et Elliott Barbot.

Ma thèse n'aurait pas eu la même saveur sans les personnes rencontrées au CVN. Un grand merci à Jana pour ta disponibilité et la joie que tu apportes au laboratoire. Merci à vous, mes collègues, pour les rires, les commentaires acerbes et jugements, les pétanques, les abominables quiz de géographie... Vous m'avez apporté de la positivité, j'espère vous avoir apporté une touche d'absurde. Au labo, j'ai connu (l'irremplaçable) Claire - autrefois voisine de bureau et maintenant véritable amie – avec qui nous avons fait profiter tout l'open-space de nos chamailleries. De mon autre voisin de bureau, JB, je garde un souvenir épique de nos escapades romaines et des pistes de ski. Malgré mes multiples tentatives, j'ai échoué à convertir Alex au monde académique mais j'ai, mieux que rien, gagné un ami. Je n'ai pas non plus réussi à calmer les ardeurs scientifiques et combatives de Younes, bien stimulé, il faut le dire, par Thomas. Un jour les réseaux de neurones k-means révolutionneront le monde et je m'en morderai les doigts... Une mention spéciale à Aymen, le pilier du CVN, qui regrettera bientôt le départ de son

ennemie, ainsi qu'à Mario pour (entre autres) le débat sur les amphibiens. Un clin d'œil reconnaissant à Mathieu, qui a généreusement géré les aspects informatiques du laboratoire (et mes problèmes avec grande patience), à Jhony pour sa gentillesse sans limite, à Théodore pour les débats houleux, à Gabriele pour sa constante bonne humeur et à Loïc pour ne pas être venu plus souvent au CVN (!). Une pensée pour Mouna, Marion, Clément, Alix, Fernando et tous les autres qui ont marqué ce parcours. Mes remerciements s'étendent également aux collègues du LIVIA à Montréal : Eduardo, Maxime, Julien et Jeanne pour les aventures que nous avons partagées, à Jérôme pour son accueil chaleureux et à Florent pour son humour toujours bienvenu.

En dehors du laboratoire, je pense à mes amis de l'ENS, futures stars du cinéma et lauréats de la médaille Fields. Sœur Elo, frères Lucas, Alex, Didou, Sylvain : partager avec vous l'expérience étrange de la thèse, cette sensation de tomber dans un trou sans fond, a été un honneur. On remet ça quand vous voulez. Merci aussi à Lola, qui m'a toujours apporté une amitié plus qu'importante pour moi.

Mes derniers remerciements vont à ma famille ; à Mildred que j'étais très heureuse de voir à ma soutenance, à mon frère à qui je souhaite beaucoup de succès dans son doctorat, et bien sûr à mes parents, qui ont toujours été une source d'inspiration. Votre soutien, vos conseils avisés et encouragements ont été essentiels tout au long de ces années.

Notation

$\mathbb{R}, \mathbb{R}_+, \mathbb{R}^*$: sets of real, non-negative real and non-zero real numbers
$\mathbb{R}^{m \times n}$: set of matrices with m rows and n columns
\mathbb{N}, \mathbb{N}^*	: sets of non-negative and positive integers
\preceq	: Loewner partial order for matrices
α	: scalars will be denoted by lowercase letters
\boldsymbol{x}	: vectors will be denoted by lowercase bold letters
\boldsymbol{H}	: matrices will be denoted by uppercase bold letters
\boldsymbol{I}_n	: identity matrix of size n
$\mathbf{1}_n$: vector of \mathbb{R}^n with all entries equal to 1
$\langle \boldsymbol{u}, \boldsymbol{v} \rangle$: Euclidean scalar product between two vectors
$\ \boldsymbol{x}\ _p$: ℓ_p -norm of a vector \boldsymbol{x} and when not specified, $p = 2$
$\text{Diag}(\boldsymbol{x})$: diagonal matrix with diagonal entries given by \boldsymbol{x}
$\text{diag}(\boldsymbol{H})$: vector of diagonal entries of \boldsymbol{H}
\mathcal{S}_n	: set of symmetric matrices of size n
\mathcal{S}_n^+	: set of symmetric positive semidefinite matrices of size n
\mathcal{S}_n^{++}	: set of symmetric positive definite matrices of size n
ι_C	: indicator function of a set C
d_C	: distance to a set C
p_C	: projection of the non-empty, closed, convex set C
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$: Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
Δ_K	: unit simplex set of \mathbb{R}^K

Acronyms

FB	: Forward-Backward
FS	: Few-Shot
FWHM	: Full Width at Half Maximum
KŁ	: Kurdyka-Łojasiewicz
l.s.c.	: lower semicontinuous
ML	: Maximum Likelihood
MAP	: Maximum A Posteriori
MDL	: Minimum Description Length
MM	: Majorization-Minimization
MMS	: Majorization-Minimization Subspace algorithm
MPM	: Multi-Photon Microscopy
PET	: Positron Emission Tomography
PSF	: Point-Spread Function
PSNR	: Peak Signal-to-Noise Ratio
SNR	: Signal-to-Noise Ratio
TV	: Total Variation

Table of contents

1	Introduction	15
1.1	Context	15
1.2	Outline and contributions	16
1.3	Collaborations	17
1.4	Publications	18
2	An overview of variational approaches	20
2.1	Introduction to variational problems	20
2.1.1	General formulation	20
2.1.2	Bayesian interpretation	21
2.1.3	Inverse problems for image recovery	22
2.1.3.1	Introduction	22
2.1.3.2	Noise model	24
2.1.3.3	Regularization	25
2.1.3.4	Examples of constraints	26
2.1.4	Image classification	28
2.1.4.1	Supervised image classification	28
2.1.4.2	Few-shot classification	29
2.1.4.3	Few-shot classification as a variational problem	32
2.1.4.4	Examples of few-shot classification methods	33
2.2	Variational analysis tools	35
2.2.1	Proper lower semi-continuous functions	36
2.2.2	Gradient and sub-gradients	36
2.2.3	Proximal operator	37
2.2.4	Kurdyka-Łojasiewicz's inequality	39
2.3	Algorithms	40
2.3.1	Smooth optimization	40
2.3.1.1	Line-search and trust region methods	40
2.3.1.2	Subspace methods	41
2.3.1.3	Majorization-Minimization approaches	42
2.3.2	Splitting algorithms	47
2.3.2.1	Forward-Backward algorithm	48

2.3.2.2	Dual Forward-Backward algorithm	49
2.3.2.3	Primal-dual Splitting Algorithm	50
2.3.3	Alternating minimization	50
2.3.3.1	Block-coordinate method	51
2.3.3.2	Proximal alternating minimization	51
2.3.3.3	Mixed alternating minimization algorithm	53
3	A local MM subspace method for solving constrained variational problems in image recovery	55
3.1	Introduction	55
3.2	Considered optimization problem	58
3.2.1	Notation and definitions	58
3.2.2	Optimization problem	59
3.3	Majorization-Minimization subspace algorithm	60
3.3.1	Majorization-Minimization principle	60
3.3.2	Majorization-Minimization subspace algorithm	61
3.3.2.1	Subspace minimization	61
3.3.2.2	MMS algorithm	63
3.4	Proposed local variant of MMS	64
3.4.1	Description of the algorithm	64
3.4.2	Convergence analysis of the proposed algorithm	67
3.4.2.1	Assumptions	67
3.4.2.2	Convergence proof	68
3.5	Embedding the proposed algorithm in a sequential penalty framework	71
3.5.1	Inexact exterior penalty method	71
3.5.2	Assumptions	72
3.5.3	Convergence analysis	74
3.5.4	Initial guess at each iteration	76
3.6	Numerical experiments	77
3.6.1	Frame-based image restoration in presence of heteroscedastic noise	77
3.6.1.1	Tight frame decomposition	78
3.6.1.2	Optimization problem	78
3.6.1.3	Proposed algorithm	79
3.6.1.4	Results	80
3.6.2	Image reconstruction in the presence of Poisson noise	81
3.6.2.1	Positron Emission Tomography	81
3.6.2.2	Considered optimization problem	82
3.6.2.3	Proposed algorithm	84
3.6.2.4	Results	85
3.7	Conclusion	87
3.A	Appendix	87

4 A novel variational approach for multiphoton microscopy image restoration	90
4.1 Introduction	91
4.2 A novel approach to PSF calibration using large beads	93
4.2.1 Preparing the data	93
4.2.2 Modeling of the problem	93
4.2.3 Proposed minimization problem	95
4.2.4 Minimization algorithm	96
4.2.4.1 Proposed algorithm	97
4.2.4.2 Expressions of the updates	97
4.2.4.3 Convergence analysis	100
4.2.5 Validation of the PSF calibration method	102
4.2.5.1 Experiments in simulated scenarios	102
4.2.5.2 Experiments on real beads in homogeneous medium	103
4.3 Proposed solution for the inverse problem	104
4.3.1 A heteroscedastic noise model for MPM	105
4.3.2 Estimation of the noise parameters	106
4.3.3 A constrained parameter-free deconvolution framework	107
4.3.4 Choices for the data-fidelity and regularization functions	108
4.3.5 Restoration algorithm	108
4.3.6 Validation of the restoration method on simulated data	110
4.4 Application of the proposed pipeline to muscle tissue imaging	111
4.4.1 Presentation of the experimental setup	112
4.4.2 Results for PSF calibration step	112
4.4.3 Results for the restoration step	114
4.5 Conclusion	115
5 Transductive few-shot learning for unbalanced image classification	118
5.1 Introduction	118
5.2 Proposed few-shot inference formulation	121
5.3 Primal-Dual Block-Coordinate Descent Optimization	124
5.4 Experiments	126
5.4.1 Experimental details	127
5.4.2 Results	128
5.5 Conclusion and limitations	131
5.A Appendices	132
5.A.1 Proof of Proposition 5.1	132
5.A.2 About the hyper-parameter in our method	134
6 Transductive zero-shot and few-shot CLIP	136
6.1 Introduction	136
6.2 Related works	138

6.2.1	Vision-language models	138
6.2.2	Few-shot classification	138
6.3	Proposed method	139
6.3.1	Computing informative feature vectors	140
6.3.2	Data distribution	141
6.3.3	Simplex-based classification criterion	141
6.4	Proposed algorithm	142
6.4.1	Minimization step w.r.t Dirichlet parameter	142
6.4.2	Minimization step w.r.t assignment variable	143
6.4.3	Global algorithm and class-assignment	144
6.5	Links with other clustering and transductive few-shot objectives	145
6.6	Experiments	147
6.6.1	Zero-shot	147
6.6.2	Few-shot	148
6.7	Conclusion	150
6.A	Appendices	150
6.A.1	About CLIP	150
6.A.2	Illustration of a Dirichlet distribution	150
6.A.3	Majorization-Minimization algorithm	151
6.A.4	Estimation step on assignments in our algorithm	152
6.A.5	Zero-shot performance as a function of the size the query set	153
6.A.6	Additional results in the few-shot setting	153
7	Conclusion and perspectives	156
7.1	Summary of contributions	156
7.2	Perspectives	157
8	Appendices	160
8.A	A non-convex proximal approach for centroid-based classification	160
8.A.1	Introduction	160
8.A.2	Centroid-based classification	161
8.A.3	A non-convex formulation to centroid-based classification	163
8.A.3.1	Separation of the centroids	163
8.A.3.2	Bounding the problem	163
8.A.3.3	An equivalent formulation	164
8.A.4	Proposed algorithm	164
8.A.4.1	Proximal alternating algorithm	164
8.A.4.2	Computation of the proximal operators	165
8.A.4.3	Overall algorithm	165
8.A.5	Numerical experiments	166
8.A.6	Conclusion	166

8.B A transductive few-shot learning approach for classification of digital histopathological slides from liver cancer	167
8.B.1 Introduction	167
8.B.2 Medical multiclass problem	168
8.B.3 Proposed method	169
8.B.3.1 Problem formulation	169
8.B.3.2 Transductive methodology	169
8.B.3.3 Minimization problem	170
8.B.3.4 Algorithm	171
8.B.4 Experiments	171
8.B.4.1 Experimental setting	171
8.B.4.2 Results	172
8.B.5 Conclusion	173
8.B.6 Ethics approval	174
8.B.7 Acknowledgments	174
List of Figures	174
List of Tables	176
Bibliography	203

- Chapter 1 -

Introduction

1.1 Context

The digital evolution in imaging technology over the last two decades has generated an extraordinary growth in high-dimensional data. It is particularly noticeable in medical and satellite imaging, where individual images can reach gigabytes in size, or in video data. Beyond single images, the cumulative volume of photographic content on devices such as smartphones contributes to the ever-growing data complexity. This proliferation of data presents a series of computational challenges in computer vision applications, demanding advanced solutions to process information effectively.

The research presented in this thesis introduces numerical solutions to two distinct large-scale imaging tasks:

- image restoration, treated as an inverse problem
- classification, addressed through the few-shot learning framework.

The common denominator across these studies is the use of variational-based optimization strategies at the core of the methods. Throughout the development of the approaches exposed in this manuscript, a number of crucial requirements have been taken into account across the various chapters.

A primary consideration is the computational efficiency of the proposed algorithms. The huge volume of information in modern datasets renders traditional computational methods inefficient. To this end, Chapter 3 is dedicated to the creation of a novel algorithm for smooth, constrained optimization, demonstrably faster than the existing alternatives. This algorithm is later applied in Chapter 4 to the practical field of multi-photon microscopy, which involves the restoration of large-scale image volumes.

The thesis also emphasizes the importance of theoretical convergence guarantees for optimization algorithms. These guarantees ensure that the algorithm yields a coherent solution and maintains stable performance despite variations in initialization, data characteristics, or noise levels. These considerations are articulated in Chapter 3 for the optimization method introduced. In Chapters 4 and 5, which focus respectively on PSF kernel estimation and few-shot classification, the discussion extends to these themes, drawing on established convergence results for alternating minimization methods.

Lastly, the thesis addresses the critical issue of scalability. Given the increasing amount of time and resources necessary for the development of supervised learning models and also energy consumption

concerns, re-training a model for each new dataset and each new task is no longer a viable approach. Instead, in recent years, the paradigm that has emerged as a promising strategy is that of transfer learning with foundation models, which are able to learn highly informative and versatile representations. To this end, Chapter 5 proposes a novel approach to few-shot learning, which is both scalable and robust to class imbalance. Chapter 6 extends this methodology to the recent field of complex vision-language models, which are particularly promising for zero-shot and few-shot image classification.

1.2 Outline and contributions

Chapter 2: An overview of variational approaches

This chapter aims to provide the reader with a comprehensive overview of the variational techniques employed in the contributions developed throughout this thesis. Specifically, we discuss variational formulations for image-related problems and their probabilistic interpretation as Maximum a Posteriori approaches. The themes of inverse problems for image restoration and few-shot image classification are presented and formulated as optimization problems. Finally, mathematical tools (such as proximal operators and sub-differentials) and algorithms relevant for this manuscript (including Majorization-Minimization methods and alternating minimization) are introduced.

Chapter 3: A local MM subspace method for solving constrained variational problems in image recovery

Chapter 3 of the thesis introduces the Penalized Majorization-Minimization Subspace (P-MMS) algorithm, a novel approach for tackling smooth, constrained optimization problems often encountered in image recovery. The method innovatively addresses the issue of ill-conditioning, which is prevalent with exterior penalty methods, by employing a trust-region-like strategy that adapts the curvature of the majorant locally. Convergence of the P-MMS algorithm is not only theoretically guaranteed but also practically validated; the algorithm demonstrates superior computational efficiency in large-scale image recovery applications, outperforming current state-of-the-art algorithms.

Chapter 4: A novel variational approach for multiphoton microscopy image restoration: from PSF estimation to 3D deconvolution

In Chapter 4, we delve into a novel variational approach for multiphoton microscopy image restoration, that begins with Point Spread Function (PSF) estimation and culminates with 3D deconvolution. The comprehensive pipeline features two major advancements: an innovative optimization technique for PSF estimation that calibrates beads of large diameters using a Gaussian shape prior, and an alternating proximal method with guaranteed convergence. In the image restoration phase, we model the noise as additive heteroscedastic and we propose a method for noise parameter estimation. Additionally, a constrained formulation for the restoration inverse problem is introduced, providing stability through an upper bound on data fidelity and utilizing the algorithm developed in Chapter 3.

Chapter 5: Realistic transductive few-shot learning

Chapter 5 shifts focus to the domain of realistic transductive few-shot learning, proposing a new practical setting that more closely mirrors real-world scenarios than traditional few-shot benchmarks. The chapter presents the few-shot problem within this novel context as a minimization challenge, incorporating a penalty term inspired by the Minimum Description Length principle to address class imbalance. An efficient primal-dual block coordinate descent algorithm is crafted to minimize the criterion with guaranteed convergence. The efficacy of the proposed method is demonstrated by extensive testing on standard benchmark datasets, illustrating its potential in managing unbalanced few-shot tasks.

Chapter 6: Transductive zero-shot and few-shot with vision-language models

Chapter 6 explores for the first time the application of transductive classification techniques to the field of vision-language models, particularly in zero-shot and few-shot scenarios. We put forward a methodology for calculating text-vision probability feature vectors, laying the groundwork for a transductive few-shot classification approach that is specifically designed for Vision-Language models. By deploying an EM-inspired algorithm that models data distribution with Dirichlet distributions, we introduce a new Majorization-Minimization algorithm for estimating jointly the class assignments and the parameters of the Dirichlet distributions. The chapter concludes with a series of comprehensive experiments across 11 classification datasets, establishing that the transductive methodology outperforms the inductive one, mirroring the patterns observed with vision-only models.

Chapter 7: Conclusion

We conclude this thesis by summarizing the key contributions and offering perspectives on potential future research directions based on the themes explored.

Chapter 8: Appendices

This manuscript includes two additional, preliminary works that were achieved at the periphery of the thesis. In one of the appendices, we explore supervised classification through transform learning, where a linear projection of data into a lower-dimensional space is learned concurrently with the classifier parameters. For this, we adopt a non-convex approach and employ a primal-dual algorithm for its resolution. In the second appendix, we apply the few-shot method from Chapter 5 to the classification of histopathology patches for liver cancer diagnostic, taking advantage of the transductive setting to account for spacial coherence in the predictions. We show promising results on real histopathology slides.

1.3 Collaborations

In addition to its joint supervision by Jean-Christophe Pesquet and Ismail Ben Ayed, this thesis has been partly co-supervised by Emilie Chouzenoux from the Centre de Vision Numérique, Université Paris-Saclay, Inria, CentraleSupélec, Gif-sur-Yvette, France. Emilie Chouzenoux was involved in several works exposed in this thesis, specifically Chapters 3, 4 and 5.

The project on restoration of multiphoton microscopy images was developed in collaboration with Claire Lefort, a physicist from the XLIM Research Institute, UMR CNRS 7252, Université de Limoges, Limoges, France. Claire Lefort provided us with the data and brought her expertise on the subject. Julien Ajdenbaum, Emilie's intern I co-supervised, contributed to the implementation of the PSF estimation pipeline. This work is exposed in Chapter 4.

The supervision of this thesis by Ismail Ben Ayed has initiated several collaborations with the Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle (LIVIA), which has since merged with the International Laboratory on Learning Systems (ILLIS) at ETS Montréal. On the topic of few-shot learning, Malik Boudiaf, a former Ph.D. student and now research scientist at Coactive AI, contributed to the numerical experiments presented in Chapter 5. Regarding vision-language models discussed in Chapter 6, Yunshi Yuang (a postdoctoral researcher) and Fereshteh Shakeri (a Ph.D. candidate) provided help in the extensive numerical evaluation of two few-shot methods, CoOp and Tip-Adapter.

The project on supervised classification, detailed in Appendix 8.A, was conducted with Meweh-Hezoudah Kahanam, Jean-Christophe Pesquet's intern.

Additionally, in collaboration with Aymen Sadraoui, a Ph.D. candidate at CVN, we co-supervised an intern, Elliott Barbot. Our work on few-shot histopathology classification in Appendix 8.B involved a collaboration with Catherine Guettier and Astrid Laurent-Bellue from the Faculté de Médecine, Université Paris-Saclay, Le Kremlin-Bicêtre, France, and INSERM U1193, Villejuif, France.

1.4 Publications

Accepted or published journal articles

E. Chouzenoux, S. Martin, J.-C. Pesquet, A local MM subspace method for solving constrained variational problems in image recovery, *Journal of Mathematical Imaging and Vision*, 2022.

Submitted journal articles

J. Ajdenbaum, E. Chouzenoux, S. Martin, C. Lefort, J.-C. Pesquet, A novel variational approach for multiphoton microscopy image restoration: from PSF estimation to 3D deconvolution, in *hal-*, 2023.

Conference proceedings

S. Martin, M. Boudiaf, E. Chouzenoux, J.-C. Pesquet, I. Ben Ayed, Towards practical few-shot query sets: Transductive minimum description length inference, *Neural Information Processing Systems (NeurIPS)*, 2022.

M. Kahanam, L. Le-Brusquet, S. Martin, J.-C. Pesquet, A non-convex proximal approach for centroid-based classification, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

S. Martin, E. Chouzenoux, J.-C. Pesquet, A penalized subspace strategy for solving large-scale constrained Optimization problems, *IEEE 29th European Signal Processing Conference (EUSIPCO)*, 2021.

J.-B. Fest, T. Heikkilä, I. Loris, S. Martin, L. Ratti, S. Rebegoldi, G. Sarnighausen, On a fixed-point continuation method for a convex optimization problem, *Advanced Techniques in Optimization for Machine learning and Imaging (ATOMI)*, 2023.

A. Sadraoui, S. Martin, E. Barbot, A. Laurent-Bellue, J.-C. Pesquet, C. Guettier, A Transductive few-shot learning approach for classification of liver cancer histopathology images, *IEEE International Symposium on Biomedical Imaging (ISBI)*, 2024.

Submitted conference proceedings

S. Martin, Y. Huang, F. Shakeri, J.-C. Pesquet, I. Ben Ayed, Transductive zero-shot and few-shot CLIP, 2023.

- Chapter 2 -

An overview of variational approaches

The need to recover or interpret data often translates to solving optimization problems in many scientific and engineering fields, especially in image processing and computer vision. At the heart of these problems lies the quest to find a suitable compromise between fitting the data and ensuring specific desirable properties of the solution. The variational formulation offers a structured approach to articulate such problems.

Section 2.1 introduces the use of variational methods for inverse problems and image classification. Although these two themes might seem unrelated at first glance, we will see that they share a common mathematical framework. The shared tools and algorithmic strategies are addressed in Sections 2.2 and 2.3.

2.1 Introduction to variational problems

2.1.1 General formulation

Given two metric spaces \mathcal{X} and \mathcal{Y} , consider a problem for which one wishes to find a solution $x \in \mathcal{X}$ that aligns with given or observed data $y \in \mathcal{Y}$. Within the scope of this thesis, the nature of the data varies based on the specific context. For instance, in the context of inverse problems, the data consist of an image degraded by optical distortions and noise. Conversely, in the classification context, the data typically comprises numerous entries, some of which are labeled. These data points serve as external inputs that ground the solution in reality. To formulate this problem variationally, one seeks the minimizer of a functional F defined on \mathcal{X} :

$$\begin{aligned} & \underset{x \in \mathcal{X}}{\text{minimize}} \quad F(x) = f(x) + \lambda g(x), \\ & \text{subject to} \quad x \in C, \end{aligned} \tag{\mathcal{P}}$$

where

- $f: \mathcal{X} \rightarrow \mathbb{R} \cup \{+\infty\}$ is the *data-fidelity* term (depending on y), quantifying how well the solution fits the observed or given data.
- $g: \mathcal{X} \rightarrow \mathbb{R} \cup \{+\infty\}$ serves as the *regularization* term. It encodes a prior knowledge or assumptions about the solution. For instance, one might use g to promote smooth or sparse solutions. Often,

another role of g is to counteract the ill-posedness of many inverse problems, offering stability in the solutions.

- $\lambda \in (0, +\infty)$ is a so-called *regularization parameter*. It determines the trade-off between the data fidelity and the regularization.
- $C \subset \mathcal{X}$ delineates the *constraints* the sought solution must satisfy.

The functions f and g , along with the constraint set C , can be adapted to address a broad spectrum of real-world problems, ranging from signal/image processing and machine learning to physics, economics, and many other domains. The sequence of actions for each application, from modeling to the production of the final result, is often as follows:

1. **Model selection.** How should f , g , and C be chosen to accurately represent and address the problem at hand?
2. **Property analysis.** Investigate the intrinsic properties of f , g such as their smoothness, convexity, and coercivity, as well as the properties of C .
3. **Existence and uniqueness.** Determine whether the problem possesses a solution, and if so, establish whether this solution is unique.
4. **Solve the problem.** If a solution exists, explore numerical methods to find a minimizer.
5. **Evaluation of the numerical method.** Assess the efficiency and effectiveness of the numerical method. Is it convergent? How quickly can an accurate solution be computed?

We will address these questions throughout this thesis.

2.1.2 Bayesian interpretation

It is worth noting the links between the formulation in (P) and the Bayesian framework. This perspective frequently offers guidance in proposing a relevant data-fidelity function.

Consider that the observation denoted $y \in \mathcal{Y}$ and the ground truth denoted $x \in \mathcal{X}$ represent realizations of random variables. The Bayesian framework postulates that one has prior beliefs about the ground truth (encoded as a probability distribution, termed the prior) and how the data was generated from the ground truth (embodied in the likelihood). After observing the data y , the ultimate goal is to update our beliefs about the unknown x , leading to the posterior distribution $p(x|y)$.

The *Maximum A Posteriori* (MAP) estimator identifies the mode of this posterior distribution, which is essentially the value where the distribution is maximized. Formally, given a prior distribution $p(x)$ and a likelihood function $p(y|x)$, the MAP estimate \hat{x}_{MAP} is given by

$$\hat{x}_{\text{MAP}} \in \underset{x \in \mathcal{X}}{\operatorname{Argmax}} p(x|y). \quad (2.1.1)$$

Expanding using Bayes' theorem yields

$$\hat{x}_{\text{MAP}} \in \underset{x \in \mathcal{X}}{\operatorname{Argmax}} \frac{p(y|x)p(x)}{p(y)}. \quad (2.1.2)$$

Since $p(y)$ is a constant with respect to the unknown x , the problem reduces to

$$\hat{x}_{\text{MAP}} \in \underset{x \in \mathcal{X}}{\operatorname{Argmax}} p(y|x)p(x). \quad (2.1.3)$$

Introducing the negative logarithm of the above equation transforms our maximization problem into a minimization problem, aligning it with the structure of a variational formulation where one can identify the data-fidelity function and the regularization:

$$\hat{x}_{\text{MAP}} \in \underset{x \in \mathcal{X}}{\operatorname{Argmin}} \underbrace{-\ln p(y|x)}_{\text{data-fidelity}} \underbrace{-\ln p(x)}_{\text{regularization}}. \quad (2.1.4)$$

Thus, a good choice for the data-fidelity function would be $f(x) = -\ln p(y|x)$. Naturally, this mandates some understanding of the distribution $p(y|x)$. Subsequent sections will demonstrate that such knowledge is frequently accessible. Concerning the regularization component, the formulation provided in (2.1.4) suggests that it should encapsulate prior information on the distribution of x .

In the following, we focus on two specific contexts of use of variational problems: image recovery in Section 2.1.3 and image classification in Section 2.1.4.

2.1.3 Inverse problems for image recovery

2.1.3.1 Introduction

Image restoration and reconstruction are essential tasks in the field of image processing and computer vision. The objective is to restore or reconstruct an image that has been degraded by various factors such as noise, blur, missing pixels, or other artifacts. Mathematically, if $\mathbf{y} \in \mathbb{R}^M$ represents the degraded image and $\bar{\mathbf{x}} \in \mathbb{R}^N$ the original image, a generic discrete inverse problem for image recovery consists in estimating $\bar{\mathbf{x}}$ given the relation

$$\mathbf{y} = \mathcal{D}(\mathbf{H}\bar{\mathbf{x}}), \quad (2.1.5)$$

where $\mathbf{H} \in \mathbb{R}^{M \times N}$ is a linear operator which models a degradation such as blur, distortion, projection, etc., and $\mathcal{D}: \mathbb{R}^M \rightarrow \mathbb{R}^M$ accounts for the noise. When the operator \mathbf{H} is unknown, the inverse problem is commonly referred to as *blind*.

Over the years, various variational-based methods have been proposed to solve (2.1.5) and have evolved with advancements in mathematical techniques and computational capabilities. We give a brief overview of the major advancements in the field over the last decades.

- **Early methods.** The most straightforward and famous approach to solving linear inverse problems is probably the least square strategy [Tikhonov, 1963, Helstrom, 1967, Albert, 1972]. Essentially, it works by minimizing the discrepancy between the observed data and the data predicted by a linear model, i.e.,

$$\underset{\mathbf{x} \in \mathbb{R}^N}{\operatorname{minimize}} \|\mathbf{y} - \mathbf{Hx}\|^2. \quad (2.1.6)$$

The solution can be determined using the pseudo-inverse of the matrix \mathbf{H} . However, this method is sensitive to the conditioning of \mathbf{H} . As a result, directly using the least squares can generate artifacts due to the ill-posed nature of many image restoration problems (e.g., when trying to

reverse blur). To counteract this, the regularized least squares approach was developed, adding a quadratic regularization term to the unknown variable \mathbf{x} , a technique also referred to as Tikhonov regularization [Tikhonov, 1963, Golub et al., 1999, Wahba, 1977, Cidade et al., 2000].

- **Non-smooth regularizations.** From the 1990s, there was a significant shift in approaches to image processing with the introduction of non-smooth regularizations. Traditional smooth regularizers often blur edges, leading to a loss of image detail. However, the advent of non-smooth regularizers, especially the ℓ_1 norm and Total Variation (TV), provided solutions that preserved these crucial details. For instance, wavelet-based denoising methods, which capitalize on the sparsity of the decomposition of the signals in the wavelet domain, used the ℓ_1 norm to achieve remarkable noise reduction [Mallat, 1999, Figueiredo and Nowak, 2003, Pizurica et al., 2006, Carrillo et al., 2012, Pustelnik et al., 2016]. On the other hand, TV regularization and its variants emerged as a powerful tool to maintain sharp edges while smoothing homogenous regions [Rudin et al., 1992, Chambolle and Pock, 2011, Condat, 2014, Abergel and Moisan, 2017]. The embrace of these non-smooth regularizations laid the foundations for advanced image restoration methods, setting new benchmarks in the field.
- **Addressing diverse noise models and constraints.** In the same time, significant advancements have been made in tackling a broad spectrum of recovery challenges. During the 1990s and 2000s, a notable change was the move beyond Gaussian noise models to explore non-Gaussian types like Poisson [Dupe et al., 2009, Zanella et al., 2009, Setzer et al., 2010], Poisson-Gaussian [Foi et al., 2008, Li et al., 2015, Chouzenoux et al., 2015], salt-and-pepper and speckle noise [Wu et al., 2011, Bioucas-Dias and Figueiredo, 2009]. These noise models presented difficulties because they often led to non-quadratic and non-smooth data-fidelity terms, complicating traditional optimization formulations. Another shift was the incorporation of constraints in optimization-based approaches for restoration and inverse problems, when earlier models predominantly banked on unconstrained optimization. Improvements in recovery performance occurred by introducing constraints, whether boundary conditions, non-negativity, or physical limits [Youla and Webb, 1982, Combettes and Trussell, 1989, Malgouyres, 2002, Combettes and Pesquet, 2004, Chen et al., 2012, Shibata et al., 2016].

In recent years, traditional methods for image recovery have resulted in relatively small leaps in performance. Contrastingly, deep learning has experienced an impressive rise in the same duration, setting new benchmarks in tasks like denoising [Zhang et al., 2017, Lehtinen et al., 2018] and deconvolution [Xu et al., 2014a]. Deep learning models differ from traditional methods that require assumptions about data characteristics. Through exposure to vast datasets, these models use millions of parameters to learn the transformation from degraded images to their cleaner counterparts, not only by explicitly understanding the degradation, but by discerning the underlying structure of what the clean image should be.

The rise of deep learning does not make variational methods any less important. In fact, they can work well together. Many contemporary deep learning models use ideas from variational methods, drawing from their strengths to enhance both performance and model interpretability [Ulyanov et al., 2018, Bertocchi et al., 2019, Repetti et al., 2022, Huang et al., 2022].

The advantages of variational methods are the following ones:

- They offer a transparent and interpretable framework;
- They do not require a large amount of data for training;
- Their robust algorithmic solutions can operate on desktop computers, making them practical without needing heavy computational resources.

Yet, they leave room for improvement, especially in:

- Refining the choice of regularization parameters. The choice of the regularization parameter remains a prevalent concern within the imaging community. Its selection profoundly impacts the visual quality of the resulting image;
- Accelerating the existing numerical methods and/or designing new ones to achieve faster convergence to the solution.

2.1.3.2 Noise model

In image recovery, understanding the nature of the noise corrupting the observed data is crucial. As we previously discussed, it directly influences the choice of the data-fidelity function. This thesis concentrates on noise models with finite variance, including both homoscedastic and heteroscedastic types, because of their prevalence in real-world imaging scenarios.

Homoscedastic Noise

Homoscedastic noise is characterized by its constant variance, irrespective of the signal intensity. This noise is additive, implying it is added to the actual signal without any dependence on the signal value. A primary example is Gaussian noise, originating from sources like the thermal agitation of electrons. Its prominence in many natural and electronic processes arises due to the Central Limit Theorem, which states that the sum of random independent variables tends to a Gaussian distribution. When modeling Gaussian noise, the data fidelity is typically chosen as the least squares function:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad f(\mathbf{x}) = \frac{1}{2} \|\mathbf{Hx} - \mathbf{y}\|_2^2. \quad (2.1.7)$$

Heteroscedastic Noise

Heteroscedastic noise presents variance that varies with the signal intensity. One of the notable instances of heteroscedastic noise is Poisson noise, which is typical in processes such as photon counting in imaging devices. It is common in low-light conditions, such as astronomical or microscopy imaging [Sarder and Nehorai, 2006, Carlan and Blanc-Féraud, 2011, Dupe et al., 2009]. With Poisson noise, the data fidelity function often takes the form:

$$(\forall \mathbf{x} \in \mathcal{O}) \quad f(\mathbf{x}) = \sum_{n=1}^N (\mathbf{Hx})_n - y_n \ln((\mathbf{Hx})_n), \quad (2.1.8)$$

where $\mathcal{O} = \{\mathbf{x} \in \mathbb{R}^N \mid (\forall n \in \{1, \dots, N\}) \quad (\mathbf{Hx})_n > 0\}$. Since this data-fidelity function can be tedious to manipulate (namely because it is not smooth on the whole space), some works resort to the Anscombe

transform [Anscombe, 1948, Li et al., 2015]. This variance-stabilizing transformation linearizes the Poisson noise, rendering it nearly Gaussian. It will be used in Chapter 3.

Another heteroscedastic model is the Poisson-Gaussian noise, a hybrid that combines the characteristics of both Poisson and Gaussian noise. Tackling this type of noise requires a data fidelity function that fuses terms specific to both Gaussian and Poisson noise types [Chouzenoux et al., 2015]. Another solution to handle this noise is to approximate it with a Gaussian heteroscedastic noise where the noise variance scales linearly with the signal intensity [Foi et al., 2008]. A weighted least squares function often captures this variance behavior:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad f(\mathbf{x}) = \frac{1}{2} \|\mathbf{W}(\mathbf{H}\mathbf{x} - \mathbf{y})\|_2^2, \quad (2.1.9)$$

where $\mathbf{W} \in \mathbb{R}^{M \times M}$ is a diagonal matrix containing weights based on signal standard deviation at each pixel, depending on $\mathbf{H}\mathbf{x}$. To estimate \mathbf{W} , one typically approximates $\mathbf{H}\mathbf{x}$ with a denoised version of \mathbf{y} . This type of noise will be considered in Chapter 4.

2.1.3.3 Regularization

Inverse problems in image processing often demand the incorporation of prior knowledge about the solution. This is typically achieved using regularization functions, which encourage specific properties in the reconstructed image. In this section, we describe some of the most widely employed regularization functions in image recovery.

Sparsity-driven regularization

A vector $\mathbf{x} \in \mathbb{R}^N$ is termed sparse if most of its components are zero. Some signals are sparse in their observation domain, such as seismic signals in the temporal domain, while some other signals may only become sparse after undergoing a suitable linear transformation (e.g., in the Fourier or wavelet domain). For such signals, the regularization function can be formulated as

$$g(\mathbf{x}) = \sum_{s=1}^S g_s(\mathbf{F}_s \mathbf{x}), \quad (2.1.10)$$

where each $\mathbf{F}_s \in \mathbb{R}^{M_s \times N}$ linearly transforms \mathbf{x} , and $g_s : \mathbb{R}^{M_s} \rightarrow \mathbb{R}$ acts as a sparsity-inducing function on the transformed signal. There exist several choices for the functions g_s that promote sparsity. For instance, the ℓ_0 pseudo-norm counts the number of non-zero components of a vector [Nikolova, 2013, Soubies et al., 2015]:

$$\ell_0(\mathbf{x}) = \|\mathbf{x}\|_0 = \sum_{n=1}^N \chi(x_n), \quad (2.1.11)$$

where $\chi(0) = 0$ and, for every $t \neq 0$, $\chi(t) = 1$. More generally, the ℓ_α functions for $0 \leq \alpha < 1$ are regularizations promoting sparsity [Soubies et al., 2017, Chouzenoux et al., 2013a]:

$$\ell_\alpha(\mathbf{x}) = \|\mathbf{x}\|_\alpha = \left(\sum_{n=1}^N |x_n|^\alpha \right)^{\frac{1}{\alpha}}, \quad (2.1.12)$$

for $\alpha \neq 0$. However, these functions are neither differentiable nor convex. To alleviate the optimization challenge, the ℓ_1 norm has been used extensively as a convex approximation of the ℓ_0 pseudo-norm [Fu et al., 2006, Avron et al., 2010, Besson et al., 2017]. The use of the ℓ_1 norm facilitates the application of linear and quadratic programming algorithms [Chen et al., 2001] as well as thresholding algorithms [Donoho, 1995, Daubechies et al., 2004].

Smoothness-driven regularizations

Smoothness-driven regularization functions are crucial in image processing, primarily for tasks where the preservation or enhancement of image structures, such as edges or textures, is desirable.

A prominent smoothness-driven regularization is the Total Variation, introduced for continuous data in [Rudin et al., 1992]. The discrete version, as defined in [Chambolle, 2004], for an image $\mathbf{x} \in \mathbb{R}^{N_1 \times N_2}$ is given as follows:

Definition 2.1. (Total Variation) The discrete Total Variation (TV) of an image $\mathbf{x} \in \mathbb{R}^N$ is given by

$$\text{TV}(\mathbf{x}) = \sum_{n=1}^N \sqrt{(\nabla_v \mathbf{x})_n^2 + (\nabla_h \mathbf{x})_n^2}, \quad (2.1.13)$$

where ∇_v and ∇_h denote respectively the vertical and horizontal gradient linear operators which can be defined according to a variety of discretization schemes.

Though highly efficient in image recovery, the TV regularization presents a computational challenge: it is non-smooth. Minimizing the TV term often requires advanced algorithms for solving the associated variational problem, making it more complex than its smooth counterparts. In addition, one of the well-known limitations of the standard TV regularization is the occurrence of the staircasing effect [Chan et al., 2005]. This refers to the introduction of artificial step-like structures in smooth or homogeneous regions of the image, making the reconstruction appear blocky. To mitigate the staircasing effect and ease the computations while retaining the edge-preserving properties of TV, the smoothed Total Variation may be preferred. In the smoothed Total Variation, the non-smooth absolute function in the TV term is replaced by a smoother function. A typical choice is

$$\text{STV}(\mathbf{x}) = \sum_{n=1}^N \sqrt{\delta + (\nabla_v \mathbf{x})_n^2 + (\nabla_h \mathbf{x})_n^2}, \quad (2.1.14)$$

where δ is a small positive constant ensuring smoothness. However, using the STV has the drawback that the Lipschitz constant of the gradient function explodes as δ tends to 0. Finally, let us mention that non-smooth adaptations of TV regularization have been developed, including Total Generalized Variation (TGV) [Bredies et al., 2010], Semi-Local Total Variation (SLTV) [Condat, 2014] and Staggered Grid TV [Condat, 2017] (TVSG).

2.1.3.4 Examples of constraints

In this section, we explore the typical constraints encountered in image recovery. While this overview is not exhaustive, it emphasizes the constraints we most frequently address in this thesis, progressing from

simpler to more advanced forms.

Linear constraints

Linear constraints are prevalent in imaging and have been thoroughly addressed by the community. Perhaps the most illustrative example of a linear inequality constraint in image recovery is the box constraint applied to pixel values [Beck and Teboulle, 2009, Chan et al., 2013], corresponding to $\mathbf{x} \in [x_{\min}, x_{\max}]^N$. Specific applications may introduce additional linear constraints. For example, in hyperspectral unmixing, the linear mixing model can be formulated as $\mathbf{Ax} = \mathbf{b}$ where the matrix \mathbf{A} stands for the spectral signatures of the endmembers [Chouzenoux et al., 2019a]. Furthermore, given that the abundance fractions for each pixel should add up to one and remain non-negative, this results in supplementary linear constraints on \mathbf{x} [Wang et al., 2017]:

$$\sum_n x_n = 1 \text{ and } x_n \geq 0, \quad (2.1.15)$$

defining a so-called unit simplex.

Ball constraints

These constraints confine solutions to specific bounded geometric regions. For instance, in optical imaging where focal intensity might need to satisfy some energy constraint [Youla and Webb, 1982], the used constraint is

$$\|\mathbf{x}\|_2 \leq \theta, \quad (2.1.16)$$

where θ is a defined threshold. On a broader scale, ℓ_p ball constraints, corresponding to

$$\|\mathbf{x}\|_p \leq \theta, \quad (2.1.17)$$

where $p \in [1, \infty]$, are relevant in image recovery. Specifically, ℓ_1 constraints are traditional choices when promoting sparsity [Mairal et al., 2008, Loris et al., 2009].

Ellipsoidal constraints

Ellipsoidal constraints generally refer to conditions that encapsulate magnitude and orientation of the solution. In this manuscript, we restrict our attention to ellipses in metrics derived from the ℓ_p norms, represented as

$$\|\mathbf{Ax} - \mathbf{b}\|_p \leq \epsilon. \quad (2.1.18)$$

Despite their broad interest, these constraints can be challenging to manage with optimization methods. Such constraints typically arise in contexts where a bound on the data-fidelity function is known. For instance, when addressing independently distributed additive Gaussian noise with variance σ^2 , the data-fidelity term, represented by $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{Hx} - \mathbf{y}\|_2^2$, is expected to satisfy (according to the law of large numbers) the approximation:

$$\|\mathbf{Hx} - \mathbf{y}\|_2^2 \approx N\sigma^2, \quad (2.1.19)$$

assuming the number of data samples is sufficiently large. Consequently, instead of tackling directly Problem (\mathcal{P}) , numerous studies have proposed imposing a limit on the data-fidelity function [Afonso et al., 2010b, Carrillo et al., 2012], leading to the constrained problem

$$\begin{aligned} & \underset{\boldsymbol{x} \in \mathbb{R}^N}{\text{minimize}} \quad g(\boldsymbol{x}), \\ & \text{subject to} \quad \boldsymbol{x} \in C \text{ and } \|\boldsymbol{H}\boldsymbol{x} - \boldsymbol{y}\|_2^2 \leq N\sigma^2. \end{aligned} \tag{2.1.20}$$

This approach notably bypasses the challenging regularization parameter tuning phase. A similar idea has been employed in [Chouzenoux et al., 2019a] in an application to Joint Geometry-Texture Decomposition with the ℓ_∞ norm. Alternatively, one can also consider a bound constraint on the regularization (e.g., TV bound [Combettes and Pesquet, 2004]).

2.1.4 Image classification

This section introduces concepts related to image classification and embeds this thematic in the variational framework. We first provide a brief overview of supervised classification before delving into few-shot classification, which will be at the core of Chapters 5 and 6.

2.1.4.1 Supervised image classification

Definition and first examples

In the context of image processing, supervised classification refers to the process by which an algorithm learns to assign predefined labels to images based on patterns found in a labeled dataset. One typically has access to three distinct datasets: a training set $\mathcal{D}_{\text{train}}$, a validation set \mathcal{D}_{val} , and a test set $\mathcal{D}_{\text{test}}$. Each dataset comprises images, each denoted by $\boldsymbol{x} \in \mathbb{R}^d$ and their corresponding labels $y \in \{1, \dots, K\}$, where K represents the number of classes for the classification task. The goal is to use the training set $\mathcal{D}_{\text{train}}$ to infer a function $\phi : \boldsymbol{x} \mapsto y$ such that for a new image not seen during training, ϕ can predict its associated label accurately. The performance of the classifier is typically refined based on feedback from the validation set \mathcal{D}_{val} and its effectiveness is then evaluated using the test set $\mathcal{D}_{\text{test}}$.

Traditional methods employed for supervised image classification encompass techniques such as Support Vector Machines (SVM) [Cortes and Vapnik, 1995], logistic regression [Cox, 1958], Linear Discriminant Analysis (LDA), and Linear Quadratic Analysis (LQA) [Ghojogh and Crowley, 2019]. However, while these methods have proven their utility in a range of scenarios, they exhibit significant limitations when dealing with large-scale challenges. For instance, when applied to datasets comprising millions of images or those with high resolutions, these methods often struggle in terms of computational efficiency and performance accuracy. As a result, there has been a pressing need for more advanced methods capable of addressing the intricacies of contemporary image datasets.

Dimensionality reduction

Employing the raw images as the classification subjects is an intuitive but often computationally demanding approach as each image lies in a high dimensional space. A widespread alternative is to con-

dense the image data by extracting essential characteristics. Therefore, reducing dimensionality without losing critical features is essential. Linear transforms like Principal Component Analysis (PCA) [Pearson, 1901] provide condensed representations by projecting images into informative vector subspaces, while the wavelet transform [Bruce et al., 2002] captures frequency and spatial details in a multi-resolution manner. The Histogram of Oriented Gradients (HOG) [Dalal and Triggs, 2005] offers another linear approach, focusing on edge orientations. On the nonlinear side, techniques such as Scale-Invariant Feature Transform (SIFT) [Lowe, 1999] identify robust keypoints, while Local Binary Patterns (LBP) [He and Wang, 1990] give texture-based representations. In the last years, neural networks have emerged as powerful nonlinear tools to learn highly informative representations, compressing images into lower-dimensional latent spaces through complex neural architectures [Krizhevsky et al., 2012, He et al., 2016].

Classifying with Neural Networks

At their core, these networks learn representations of input data by passing them through multiple layers of transformations. Especially in architectures like Convolutional Neural Networks (CNNs), the initial layers often detect rudimentary features like edges and textures, while deeper layers capture more abstract and complex patterns, culminating in a recognition of objects or scenes. The terminal part of these networks typically comprises a fully connected layer that, in essence, performs a function analogous to logistic regression, mapping the learned high-dimensional representations to class probabilities. Since their widespread adoption, neural networks have dramatically outperformed traditional classification methods. However, it is worth noting that this enhanced performance comes with a constraint: **deep neural networks necessitate vast amounts of data to be trained effectively, and often require thousands of labeled examples to generalize well.**

2.1.4.2 Few-shot classification

Definition and vocabulary

Few-shot classification refers to a classification challenge where each class is represented by only a handful of labeled examples. Given the sparse nature of the training data, conventional classification techniques are often inadequate, making the task particularly challenging. We now introduce the vocabulary and notation used in this thesis to describe few-shot classification problems. As the vocabulary can be confusing, we also provide a summary in Table 2.1. A few-shot task assumes a *support* set with “few” labeled examples and a *query* set with unlabeled examples. Throughout this thesis, we will systematically denote by $\mathbb{S} \subset \{1, \dots, N\}$, the indexes of images in the support set and its complement by $\mathbb{Q} \subset \{1, \dots, N\}$ the indexes of images in the query set. **The goal is to predict the labels of the query set.** To draw the parallel with supervised classification, the support set would correspond to the training set, and the query set to the test set. Each labeled example in the support set is called a *shot* and the number of shots per class is denoted s . The number of different classes k represented in the support set is referred to as the number of *ways*. For instance, we say the few-shot task is a 5-way 1-shot task if the support set contains 5 classes and each class is represented by one labeled example. An illustration is given in Figure 2.1. It is worth noting that when the support set is empty ($\mathbb{S} = \emptyset$), the tasks reduces to unsupervised

learning. In this context, the objective shifts to clustering the query set into K distinct groups.

Term	Notation	Definition	Typical setting
Shots	s	# of labeled examples per class in the support	5 shots
Ways	k	# of classes in the support set	5 classes
Support set	\mathbb{S}	Set of labeled examples	$ \mathbb{S} = ks$
Query set	\mathbb{Q}	Set of unlabeled examples	$ \mathbb{Q} = 75$

Table 2.1: Summary of the vocabulary used in few-shot classification. In the typical (artificial) tasks considered in the literature, the support set is constituted of an equal number of labeled samples of each class. This number is then called *shot*. We might relax this assumption across the manuscript.

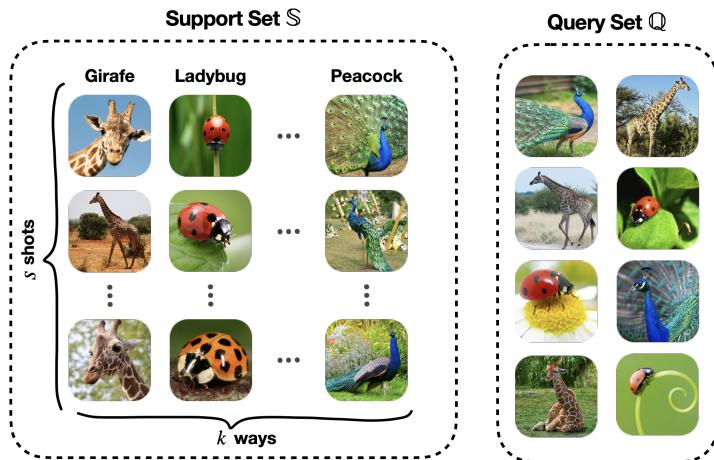


Figure 2.1: Illustration of a k -way s -shot classification task. The goal is to predict the labels of the query set.

Pretraining vs meta-learning paradigm

In the context of few-shot learning, two dominant paradigms emerge: the *meta-learning* paradigm and the *pretraining* paradigm. Both strategies rely on the requisit of having access to a large labeled classification dataset, denoted $\mathcal{D}_{\text{base}}$, distinct from the few-shot dataset. The base dataset is used to train a neural network. The difference between the two paradigms lies in the way the network is trained.

Meta-learning, also known as “learning to learn,” trains the full model on artificial few-shot tasks composed of instances in $\mathcal{D}_{\text{base}}$. The model is trained to learn the structure of the tasks, enabling it to grasp the inherent structure of the tasks and to adapt to new datasets [Snell et al., 2017, Finn et al., 2017, Vinyals et al., 2016, Sung et al., 2018]. However, a notable challenge with meta-learning is its tendency to overfit to the kinds of tasks encountered during the meta-training phase. This can cause the model to become overly tailored to those specific tasks it trained on, potentially underperforming when introduced to new, dissimilar tasks. Specifically, if the model is trained on 5-shot 5-way tasks as it is standard in the literature, it might fail to predict the labels of a 1-shot 10-way task.

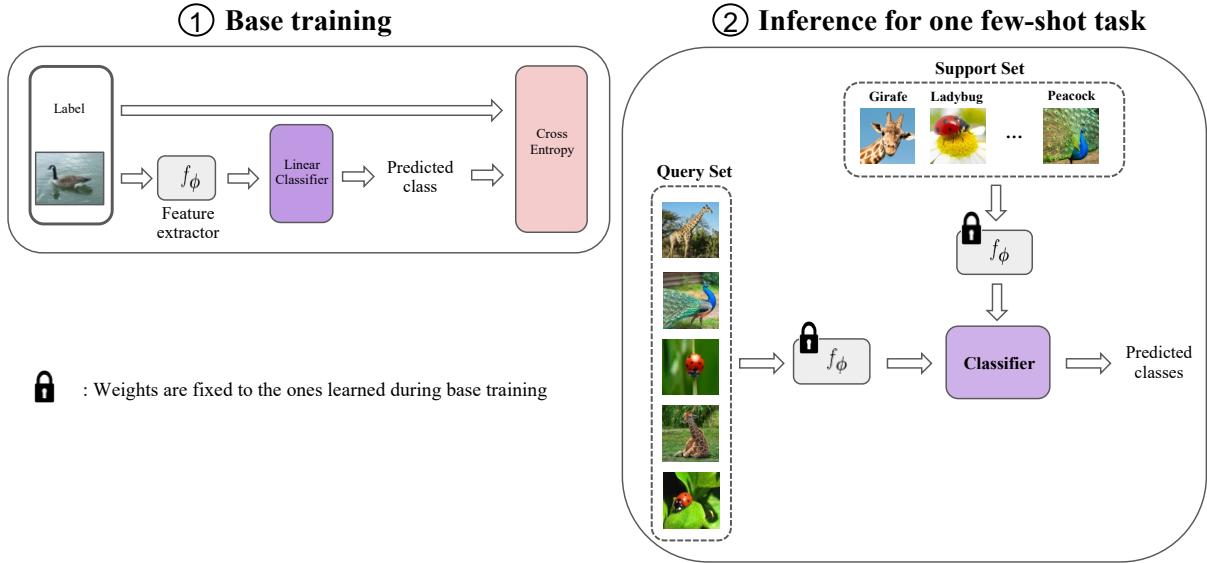


Figure 2.2: Illustration of the pretraining approach in few-shot learning. First, a neural network is trained on a substantial labeled dataset, $\mathcal{D}_{\text{base}}$, where the samples are drawn from classes denoted by $\mathcal{C}_{\text{base}}$. Then, during inference, a few-shot task is defined using samples from a distinct dataset, $\mathcal{D}_{\text{test}}$, which comprises classes $\mathcal{C}_{\text{test}}$. It is crucial that these classes be new, ensuring $\mathcal{C}_{\text{base}} \cap \mathcal{C}_{\text{test}} = \emptyset$. The pretrained feature extractor, whose parameters are kept fixed (or “frozen”), is then employed to extract features from the new dataset, and these features are utilized to train a classifier tailored to the new task.

Conversely, in the pretraining paradigm, a neural network is first trained in the traditional manner using a simple cross-entropy loss on the dataset $\mathcal{D}_{\text{base}}$. Then, at inference, the images of the few-shot task are passed to the pretrained network to extract feature vectors. These features are next given as inputs of a classifier [Chen et al., 2019] which is yet to be trained on the support set. An illustration is given in Figure 2.2. Rooted in transfer learning principles, this method applies knowledge from one problem (supervised classification) to address another related challenge (few-shot classification, in the present case). Its advantages include being practical, fast, and less resource-intensive than meta-learning.

It has been widely established over recent years that pretraining strategies outperform most meta-learning methods [Wang et al., 2019, Tian et al., 2020, Ziko et al., 2021, Boudiaf et al., 2020]. Therefore, we adopt the pretraining framework for the remaining of the manuscript.

Inductive vs transductive setting

One can distinguish between *inductive* and *transductive* few-shot approaches for the forthcoming discussions. In the context of inductive few-shot learning, each instance in the query set is classified independently, without considering any correlation or shared information with other instances. On the other hand, transductive approaches make joint predictions for the batch of query samples of each few-shot task, accounting for the statistics of the query set of a task and leveraging shared information across these instances for better generalization and accuracy. The transduction approach has become

increasingly popular in few-shot learning, and a large body of recent methods focused on this setting, including, for instance, those based on graph regularization [Ziko et al., 2020, Liu et al., 2019], optimal transport [Lazarou et al., 2021, Hu et al., 2021], feature transformations [Liu et al., 2020, Cui and Guo, 2021], information maximization [Veilleux et al., 2021, Boudiaf et al., 2020, Boudiaf et al., 2021] and transductive batch normalization [Bronskill et al., 2020, Finn et al., 2017], among other works [Dhillon et al., 2019, Wang et al., 2020, Qi et al., 2021, Shen et al., 2021]. On standard benchmarks, the gap in classification accuracy between transductive and inductive few-shot methods may reach 10%; see [Boudiaf et al., 2020], for example. This connects with a well-known fact in classical literature on transductive inference [Vapnik, 1999, Joachims, 1999], which prescribes transduction as an effective way to mitigate the difficulty inherent to learning from limited labels. It is worth mentioning that transductive methods inherently depend on the statistical properties of the query sets. For instance, the recent studies in [Lichtenstein et al., 2020, Veilleux et al., 2021] showed that variations in the class balance within the query set may affect the performances of transductive methods.

2.1.4.3 Few-shot classification as a variational problem

Few-shot classification methods are often built on the Maximum Likelihood (ML) formulation. We first introduce the generic ML formulation and then offer an overview of its use in the state-of-the-art. Given:

- N , the total number of samples in the union of the support and query sets.
- $\mathbf{Z} = (\mathbf{z}_n)_{1 \leq n \leq N} \in (\mathbb{R}^d)^N$, the collection of feature vectors extracted with a pretrained network.
- $\mathbf{Y} = (\mathbf{y}_n)_{n \in \mathbb{S}} \in \{0, 1\}^{|\mathbb{S}| \times K}$, representing the one-hot-encoded labels for the samples in the support set. This means that for every $n \in \mathbb{S}$, for every $k \in \{1, \dots, K\}$, $y_{n,k} = 1$ if \mathbf{z}_n belongs to class k , $y_{n,k} = 0$ otherwise.

Following a parametric statistical model, we aim to determine:

- $\mathbf{U} = (\mathbf{u}_n)_{n \in \mathbb{Q}} \in \Delta_K^{|\mathbb{Q}|}$, the soft assignments for the query set samples, where Δ_K denotes the simplex of \mathbb{R}^K :

$$\Delta_K = \left\{ \mathbf{u} = (u_k)_{1 \leq k \leq K} \in \mathbb{R}^K \mid (\forall k \in \{1, \dots, K\}) u_k \geq 0 \text{ and } \sum_{k=1}^K u_k = 1 \right\}. \quad (2.1.21)$$

For every $n \in \mathbb{S}$ and $k \in \{1, \dots, K\}$, $u_{n,k}$ can be understood as the probability that the sample \mathbf{z}_n belongs to class k .

- $\Theta = (\boldsymbol{\theta}_k)_{1 \leq k \leq K}$, the parameters defining each class (e.g., means and covariances for Gaussian distributions).

Inspired from maximum likelihood estimation, a probabilistic approach to estimate both the assignments and the model parameters can be expressed as

$$\begin{aligned} & \underset{\mathbf{U}, \Theta}{\text{maximize}} \quad \mathbf{p}(\mathbf{Z} | \mathbf{U}, \Theta) = \prod_{n=1}^N \prod_{k=1}^K \mathbf{p}(\mathbf{z}_n | \boldsymbol{\theta}_k)^{u_{n,k}}, \\ & \text{subject to} \quad \mathbf{u}_n \in \Delta_K \quad (\forall n \in \mathbb{Q}), \\ & \quad u_{n,k} = y_{n,k} \quad (\forall n \in \mathbb{S}) (\forall k \in \{1, \dots, K\}), \end{aligned} \quad (2.1.22)$$

or equivalently

$$\begin{aligned} & \underset{\boldsymbol{U}, \boldsymbol{\Theta}}{\text{minimize}} \quad - \sum_{n=1}^N \sum_{k=1}^K u_{n,k} \log p(\boldsymbol{z}_n | \boldsymbol{\theta}_k), \\ & \text{subject to} \quad \boldsymbol{u}_n \in \Delta_K \quad (\forall n \in \mathbb{Q}), \\ & \quad u_{n,k} = y_{n,k} \quad (\forall n \in \mathbb{S}) \quad (\forall k \in \{1, \dots, K\}). \end{aligned} \tag{2.1.23}$$

This probabilistic model will be discussed in more details in Chapter 6.

Remark 2.2. Let us emphasize that the problem in (2.1.23) is versatile, suitable for different few-shot scenarios:

- **Inductive setting:** Here, $|\mathbb{Q}| = 0$ when solving (2.1.23). The problem simplifies to a standard supervised classification problem: parameters are derived only from the support set. In a second phase, the model may be used to predict the labels of samples in the test set independently one from another.
- **Transductive setting:** With multiple samples in the query set ($|\mathbb{Q}| \geq 1$), the parameters are influenced by both the support and query sets.
- **Zero-shot setting (akin to clustering):** No support set exists ($|\mathbb{S}| = 0$). Class parameters are deduced solely from the query set.

A majority of few-shot classification approaches use the ML formulation as depicted in Equation (2.1.23) as a starting point. Interestingly, most of these methods consider the same class-conditional distributions $p(\boldsymbol{z}_n | \boldsymbol{\theta}_k)$, namely Gaussian distributions with identity covariance matrices. A discussion on the validity of this assumption in the context of vision-language models will be conducted in Chapter 6.

2.1.4.4 Examples of few-shot classification methods

We now present a selection of methods, distinguishing them based on their inductive or transductive nature. Our upcoming numerical experiments in Chapter 5 will compare several of these methods. Inductive methods are typically simple and computationally efficient. Key examples are:

- **Simple Shot** [Wang et al., 2019] is a straightforward approach. It calculates the means $(\boldsymbol{w}_k)_{1 \leq k \leq K} \in (\mathbb{R}^d)^K$ -commonly referred to as *prototypes*-of the K classes by averaging the features of the support set, i.e.

$$\boldsymbol{w}_k = \frac{1}{|\mathbb{S}|} \sum_{n \in \mathbb{S}} y_{n,k} \boldsymbol{z}_n. \tag{2.1.24}$$

For any sample \boldsymbol{z}_n in the query set, the predicted class is then determined by

$$\underset{1 \leq k \leq K}{\operatorname{argmin}} \| \boldsymbol{z}_n - \boldsymbol{w}_k \|^2. \tag{2.1.25}$$

In this method, the class-conditional distributions are assumed to be Gaussian with identity covariance matrices.

- **Baseline** [Chen et al., 2019] performs a basic fine-tuning of the network final layer using the support set. This process is akin to performing a logistic regression on the support. The method then applies the learned classifier on the query set.

Transductive methods, which are central to this thesis, are generally more intricate than their inductive counterparts. Such methods can necessitate iterative algorithms and, depending on the objective function, efficient optimization algorithms capable of handling high-dimensional data. Key transductive methods explored in this thesis are:

- **Probabilistic K-means** [Bezdek, 2013] is a method that acts as a bridge between traditional clustering approaches and few-shot learning techniques. In Chapter 5, we propose a few-shot method which has close links with probabilistic K-means. In essence, it extends the traditional K-means clustering algorithm to handle labeled and unlabeled data simultaneously. The method tackles the following optimization problem:

$$\underset{\mathbf{U}, \mathbf{W}}{\text{minimize}} \quad \sum_{k=1}^K \sum_{n \in \mathbb{Q}} u_{n,k} \|\mathbf{z}_n - \mathbf{w}_k\|^2 + \sum_{k=1}^K \sum_{n \in \mathbb{S}} y_{n,k} \|\mathbf{z}_n - \mathbf{w}_k\|^2, \quad (2.1.26)$$

where $\mathbf{W} = (\mathbf{w}_k)_{1 \leq k \leq K}$ are the centroid for the K classes. The algorithm proceeds by iteratively, updating alternatively the centroids and the assignment variables based on the query and support set.

- **BD-CSPN** [Liu et al., 2020] mitigates biases in the query set due to domain shifts (i.e., distribution shifts) that arise in few-shot when following the pretraining paradigm. After normalizing features, class prototypes $(\mathbf{w}_k)_{1 \leq k \leq K}$ are computed with the support set. The following method is used:

1. Shifts features of the query, ensuring that the average of these query features matches the mean of the prototypes (cross-class bias reduction).
2. Sets initial predictions by identifying the closest prototype for each instance as in (2.1.25), updates prototypes as

$$\mathbf{w}_k = \frac{1}{|\mathbb{S}| + |\mathbb{Q}|} \sum_{n \in \mathbb{S} \cup \mathbb{Q}} u_{n,k} \mathbf{z}_n, \quad (2.1.27)$$

and updates predictions again with the new prototypes.

- **PT-MAP** [Hu et al., 2021] uses optimal transport theory in two stages:
 1. Feature vectors from both the query and support sets are transformed so that their distributions resemble Gaussian-like distributions, by using a Power Transform (PT).
 2. The few-shot problem is viewed as a discrete optimal transport problem based on Kantorovich's formulation with entropic regularization. The method iteratively updates the assignment matrix $\mathbf{M} \in \mathbb{R}^{|\mathbb{Q}| \times K}$ with a Sinkhorn-style algorithm, and the class centers $(\mathbf{w}_k)_{1 \leq k \leq K}$.
- **Laplacian-Shot** [Ziko et al., 2020] can be understood as a continuous relaxation of the K-modes Laplacian problem. The K-modes Laplacian problem targets the minimization of a binary-assignment function composed of two terms: (i) A term that allocates query instances to the

most proximate class prototype, and (*ii*) another pairwise Laplacian term that advocates for label coherence amongst adjacently situated query instances. This is mathematically formulated as

$$\underset{(\mathbf{u}_n)_{n \in \mathbb{Q}} \in (\nabla_K)^{|\mathbb{Q}|}}{\text{minimize}} \sum_{n \in \mathbb{Q}} \sum_{k=1}^K u_{n,k} \|\mathbf{z}_n - \mathbf{w}_k\|^2 + \frac{\lambda}{2} \sum_{m,n \in \mathbb{Q}} a(\mathbf{z}_m, \mathbf{z}_n) \|\mathbf{u}_m - \mathbf{u}_n\|^2, \quad (2.1.28)$$

where, for every $n \in \mathbb{Q}$, $\mathbf{u}_n = (u_{n,k})_{1 \leq k \leq K} \in \mathbb{R}^K$ symbolizes a binary assignment variable. For every $k \in \{1, \dots, K\}$, \mathbf{w}_k is the prototype of class k , deduced from the support set. The function $a: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ captures the similarity between two feature vectors. The set ∇_K corresponds to the simplex with integer restrictions imposed on $(\mathbf{u}_n)_{n \in \mathbb{Q}}$.

- **TIM (Task-Information Maximization)** [Boudiaf et al., 2020] aims at minimizing the following objective with respect to the classifier weights $\mathbf{W} = (\mathbf{w}_k)_{1 \leq k \leq K} \in (\mathbb{R}^d)^K$:

$$\underset{\mathbf{W}}{\text{minimize}} \underbrace{-\frac{1}{|\mathbb{S}|} \sum_{n \in \mathbb{S}} \sum_{k=1}^K y_{n,k} \ln(u_{n,k})}_{\text{Cross-Entropy}} - \underbrace{\frac{\alpha}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} \sum_{k=1}^K u_{n,k} \ln(u_{n,k}) + \sum_{k=1}^K \hat{u}_k \ln(\hat{u}_k)}_{-\mathcal{I}_\alpha}, \quad (2.1.29)$$

where \mathcal{I}_α is called the weighted mutual information, the posterior posterior probabilities are given by

$$(\forall n \in \{1, \dots, N\})(\forall k \in \{1, \dots, K\}) \quad u_{n,k} \propto \exp\left(-\frac{\tau}{2} \|\mathbf{w}_k - \mathbf{z}_n\|^2\right), \quad (2.1.30)$$

and the marginal distribution by

$$(\forall k \in \{1, \dots, K\}) \quad \hat{u}_k = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}. \quad (2.1.31)$$

The first two terms of the cost seek to reduce the uncertainty in the posterior probabilities of unlabeled query samples. Meanwhile, the last term, the label-marginal entropy regularizer, promotes a uniform distribution of labels.

- **α -TIM** [Veilleux et al., 2021]. α -TIM is an adaptation of TIM for few-shot scenarios that are not perfectly balanced (meaning the query set comprises $K \times m$ instances, with m instances from each of the K classes present in the support set). In (2.1.29), the label-marginal entropy term, whose minimization is equivalent to minimizing a Kullback Leibler distance to the uniform distribution, is substituted by an α -divergence, with α ranging within $(0, +\infty)$. Essentially, this applies a Dirichlet distribution prior on the class proportions. However, this approach necessitates knowledge of the anticipated class proportions, as the parameter α is determined by the user.

2.2 Variational analysis tools

In this section, we outline mathematical concepts essential to the discussions in the later chapters of this thesis. While most of the definitions – including lower semi-continuity, sub-differentiability, and the proximal operator – are presented in the context of the finite-dimensional space \mathbb{R}^N to align with the scope of this thesis, it is worth noting that these definitions can be extended to arbitrary Hilbert spaces.

2.2.1 Proper lower semi-continuous functions

Properness and lower semi-continuity are properties central to functional analysis. These attributes constitute a basis for building other mathematical concepts.

Definition 2.3. (Domain, proper) Let $f: \mathbb{R}^N \rightarrow \mathbb{R} \cup \{-\infty, +\infty\}$. The *domain* of f is defined as $\text{dom } f = \{\mathbf{x} \in \mathbb{R}^N \mid f(\mathbf{x}) < +\infty\}$. Moreover, we say f is *proper* if

$$-\infty \notin f(\mathbb{R}^N) \text{ and } \text{dom } f \neq \emptyset. \quad (2.2.1)$$

Definition 2.4. (Lower semi-continuity) Let $f: \mathbb{R}^N \rightarrow \mathbb{R} \cup \{-\infty, +\infty\}$. We say f is *lower semi-continuous* at \mathbf{x} if, for every sequence $(\mathbf{x}_k)_{k \in \mathbb{N}}$ such that $\mathbf{x}_k \xrightarrow[k \rightarrow +\infty]{} \mathbf{x}$,

$$f(\mathbf{x}) \leq \underline{\lim} f(\mathbf{x}_k).$$

Moreover, f is said lower semi-continuous if it is lower semi-continuous at every point in \mathbb{R}^N .

In particular, continuity implies lower semi-continuity. Next, we give an important example of a function which is lower semi-continuous but not continuous.

Definition 2.5. (Indicator function) Let C be a subset of \mathbb{R}^N . We denote $\iota_C: \mathbb{R}^N \rightarrow \mathbb{R} \cup \{+\infty\}$ the function *indicator function* defined as

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \iota_C(\mathbf{x}) = \begin{cases} 0 & \text{if } \mathbf{x} \in C \\ +\infty & \text{otherwise.} \end{cases}$$

Example 2.6. Let $C \subset \mathbb{R}^N$. Then ι_C is lower semi-continuous if and only if C is closed.

Notation 2.7. The set of proper, lower semi-continuous, convex functions on \mathbb{R}^N is denoted by $\Gamma_0(\mathbb{R}^N)$.

2.2.2 Gradient and sub-gradients

For functions in $\Gamma_0(\mathbb{R}^N)$, we can define a notion of subdifferentiability that coincides with classical differentiability when the function is differentiable.

Definition 2.8. (Moreau subdifferential) [Bauschke and Combettes, 2011, Def. 16.1] Let $f \in \Gamma_0(\mathbb{R}^N)$. The *Moreau subdifferential* of f , denoted by ∂f , is defined as

$$\begin{aligned} \partial f : \mathbb{R}^N &\longrightarrow 2^{\mathbb{R}^N} \\ \mathbf{x} &\longmapsto \{\mathbf{u} \in \mathbb{R}^N \mid (\forall \mathbf{y} \in \mathbb{R}^N) \langle \mathbf{y} - \mathbf{x} | \mathbf{u} \rangle + f(\mathbf{x}) \leq f(\mathbf{y})\}. \end{aligned} \quad (2.2.2)$$

The notion of sub-differential extends beyond functions in $\Gamma_0(\mathbb{R}^N)$ through what is termed the *limiting sub-differential* [Rockafellar and Wets, 2009]. However, since the Moreau and limiting sub-differentials coincide for functions in $\Gamma_0(\mathbb{R}^N)$, and given that Definition (2.8) suffices for the scope of this thesis, we will exclusively rely on it.

In short, as illustrated in Figure 2.3, the sub-differential of $f \in \Gamma_0(\mathbb{R}^N)$ at \mathbf{x} can be identified through the set of all hyperplanes that minorize f and have the same value at \mathbf{x} as f .

The following proposition establishes the link between Fréchet differentiability and sub-differentiability.

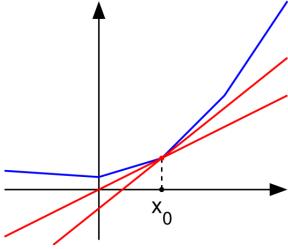


Figure 2.3: The function is in $\Gamma_0(\mathbb{R}^N)$ but is not Fréchet differentiable at \mathbf{x}_0 . Its sub-differential is composed of all the slopes of the red lines.

Proposition 2.9. (Link with Fréchet differentiability) *Let $f \in \Gamma_0(\mathbb{R}^N)$. Suppose f is Fréchet differentiable. Then,*

$$\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}. \quad (2.2.3)$$

Let $f: \mathbb{R}^N \rightarrow \mathbb{R}$. If f is differentiable, the necessary condition for a local minimum is $\nabla f(\mathbf{x}) = 0$. It is well-known that this condition becomes sufficient when f is convex. The following proposition provides the analogous condition for subdifferentiable functions.

Proposition 2.10. (Fermat's rule) *Let $f \in \Gamma_0(\mathbb{R}^N)$. The following equivalence stands:*

$$\mathbf{x} \in \operatorname{Argmin} f \iff 0 \in \partial f(\mathbf{x}). \quad (2.2.4)$$

Finally, we conclude this brief overview of subdifferentiability by presenting a few essential properties to the works exposed in this thesis.

Proposition 2.11. (Useful properties) *Let $f \in \Gamma_0(\mathbb{R}^N)$. Let $\mathbf{x} \in \mathbb{R}^N$.*

- (i) *For every $\lambda \in (0, +\infty)$, $\partial(\lambda f)(\mathbf{x}) = \lambda \partial f(\mathbf{x})$.*
- (ii) *Let $g \in \Gamma_0(\mathbb{R}^N)$ and assume that $\operatorname{int}(\operatorname{dom} g - \operatorname{dom} f) \neq \emptyset$, then*

$$\partial(f + g)(\mathbf{x}) = \partial f(\mathbf{x}) + \partial g(\mathbf{x}).$$

2.2.3 Proximal operator

Over the past decade, the proximal operator has become increasingly significant in the field of inverse problems, especially in signal processing and data science [Combettes and Pesquet, 2011, Combettes and Pesquet, 2021]. It acts as a powerful tool for tackling minimization problems with possibly nonsmooth functions.

Definition 2.12. For $f \in \Gamma_0(\mathbb{R}^N)$, the operator given by

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \operatorname{prox}_f(\mathbf{x}) = \operatorname{argmin}_{\mathbf{u} \in \mathbb{R}^N} \left(f(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \mathbf{x}\|^2 \right),$$

is well-defined and is referred to as the *proximal operator* of f .

Example 2.13. A proximal operator that frequently arises in this thesis is the one of the indicator function ι_C , where C denotes a non-empty, closed convex set. It holds that:

$$\text{prox}_{\iota_C} = \text{proj}_C,$$

where proj_C is the Euclidean projection onto C .

Example 2.14. Another important proximal operator for the upcoming chapters on classification corresponds to $f: \mathbb{R}^N \rightarrow \mathbb{R}$ defined by:

$$(\forall \mathbf{x} = (x_n)_{1 \leq n \leq N}) \quad f(\mathbf{x}) = \sum_{n=1}^N \left(\varphi(x_n) - \frac{x_n^2}{2} \right), \quad (2.2.5)$$

where

$$(\forall u \in \mathbb{R}) \quad \varphi(u) = \begin{cases} u \ln(u) & \text{if } u > 0, \\ 0 & \text{if } u = 0, \\ +\infty & \text{otherwise.} \end{cases} \quad (2.2.6)$$

The function f is proper, lower semi-continuous, and convex. Moreover,

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \text{prox}_f(\mathbf{x}) = \text{softmax}(\mathbf{x}), \quad (2.2.7)$$

with the softmax function defined as, for every $\mathbf{x} \in \mathbb{R}^N$,

$$\text{softmax}(\mathbf{x}) = \left(\frac{\exp(x_n)}{\sum_{j=1}^N \exp(x_j)} \right)_{1 \leq n \leq N} \in \mathbb{R}^N. \quad (2.2.8)$$

This function is particularly useful in the context of classification, as it maps a vector of real numbers to a discrete probability distribution.

For a comprehensive collection of proximal operators, we recommend exploring the Proximity Repository, available online at <http://proximity-operator.net/>. Proximal operators possess appealing properties that render them especially adapted to iterative minimization algorithms, and in particular the following one.

Proposition 2.15. Let $f \in \Gamma_0(\mathbb{R}^N)$. The proximal operator of f is firmly nonexpansive, i.e.,

$$(\forall (\mathbf{x}, \mathbf{y}) \in (\mathbb{R}^N)^2) \quad \|\text{prox}_f(\mathbf{x}) - \text{prox}_f(\mathbf{y})\|^2 + \|(\mathbf{x} - \text{prox}_f(\mathbf{x})) - (\mathbf{y} - \text{prox}_f(\mathbf{y}))\|^2 \leq \|\mathbf{x} - \mathbf{y}\|^2. \quad (2.2.9)$$

The subsequent property will help in computing proximal operators throughout the thesis.

Proposition 2.16. For $f \in \Gamma_0(\mathbb{R}^N)$, given $a \in \mathbb{R}$ and $\mathbf{b} \in \mathbb{R}^N$, let

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad g(\mathbf{x}) = f(\mathbf{x}) + \langle \mathbf{b}, \mathbf{x} \rangle + a\|\mathbf{x}\|^2.$$

Then, it follows that:

$$(\forall \mathbf{x} \in \mathbb{R}^N)(\forall \gamma > 0) \quad \text{prox}_{\gamma g}(\mathbf{x}) = \text{prox}_{\gamma f/(2a\gamma+1)} \left(\frac{\mathbf{x} - \gamma \mathbf{b}}{2a\gamma + 1} \right). \quad (2.2.10)$$

Lastly, the upcoming result establishes the connection between the proximal operator and the subdifferential.

Proposition 2.17. [Bauschke and Combettes, 2011, Chap.16] *For $f \in \Gamma_0(\mathbb{R}^N)$, the following equivalence holds:*

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{p} = \text{prox}_f(\mathbf{x}) \iff \mathbf{x} - \mathbf{p} \in \partial f(\mathbf{p}). \quad (2.2.11)$$

2.2.4 Kurdyka-Łojasiewicz's inequality

We next introduce the *Kurdyka-Łojasiewicz's inequality*. In recent years, this property has become a fundamental tool for convergence analysis in a nonconvex setting [Bolte et al., 2006, Bolte et al., 2014]. This property will be often assumed on the functions employed in the manuscript to ensure the convergence of the algorithms.

Definition 2.18. (Kurdyka-Łojasiewicz's inequality) Let $f: \mathbb{R}^N \rightarrow \mathbb{R} \cup \{+\infty\}$. Function f is said to possess the Kurdyka-Łojasiewicz property at $\bar{\mathbf{x}}$ in $\text{dom } \partial f$ if there exist η in $(0, +\infty)$, a neighborhood U of $\bar{\mathbf{x}}$, and a continuous concave function $\varphi: [0, \eta) \rightarrow \mathbb{R}^+$ such that:

- $\varphi(0) = 0$,
- φ is continuously differentiable on $(0, \eta)$,
- $(\forall s \in (0, \eta)) \varphi'(s) > 0$,
- For every $\mathbf{x} \in U$ with $f(\bar{\mathbf{x}}) < f(\mathbf{x}) < f(\bar{\mathbf{x}}) + \eta$, the following inequality is satisfied:

$$\varphi'(f(\mathbf{x}) - f(\bar{\mathbf{x}})) \text{dist}(0, \partial f(\mathbf{x})) \geq 1. \quad (2.2.12)$$

The definition of the Kurdyka-Łojasiewicz (KL) property for a critical point suggests that, up to a local reparametrization of f around \mathbf{x}^* , the function f is not flat at \mathbf{x}^* . Indeed, the reparametrization function φ can be seen as a desingularizing function, in the sense that it transforms the gradients of f that are arbitrarily close to zero (termed ‘singular’) to elements outside of the unit ball (termed ‘regular’).

Example 2.19. The KL inequality is satisfied by a wide range of functions, particularly the vast majority of functions encountered in the context of image processing or computer vision.

- A lower semi-continuous function that is proper exhibits the KL property at every non-critical point.
- In 1963, S. Łojasiewicz demonstrated in [Łojasiewicz, 1963] that real-analytic functions conform to an inequality of the given type with $\varphi(s) = s^{1-\theta}$ for some θ in $[\frac{1}{2}, 1)$. Hence, if f is real analytic at \mathbf{x} , then f is KL at \mathbf{x} . In particular, this case includes exponentials, logarithms, Gamma function etc.
- If f is semi-algebraic at \mathbf{x} , then f is KL at \mathbf{x} [Bolte et al., 2006]. Note that the set of semi-algebraic functions is stable under various operations (sum, product, composition...) and contains polynomial functions, the square root function and the distance to a semi-algebraic set.
- More generally, f is KL at \mathbf{x} if f is definable in an o-minimal structure at \mathbf{x} [Van den Dries and Miller, 1996, Kurdyka, 1998, Bolte et al., 2007].

2.3 Algorithms

The last section of this overview chapter is dedicated to the presentation of the algorithms that will appear throughout this thesis.

2.3.1 Smooth optimization

2.3.1.1 Line-search and trust region methods

Let us begin by presenting a succinct overview of standard methods employed for solving unconstrained differentiable optimization problems. These techniques primarily fall into two distinct categories: the *line-search* methods and the *trust region* methods. A comprehensive survey on this can be found in [Conn et al., 2000].

Let $F: \mathbb{R}^N \rightarrow \mathbb{R}$ be a differentiable function one aims at minimizing. Both of these categories of methods produce sequences $(\mathbf{x}_k)_{k \in \mathbb{N}}$ following the relation:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k, \quad (2.3.1)$$

where $\mathbf{s}_k \in \mathbb{R}^N$. The core distinction between the two arises from their respective strategies in computing the *increment* \mathbf{s}_k for each iteration.

In line-search methods, a direction $\mathbf{d}_k \in \mathbb{R}^N$ is selected at each iteration. The algorithm then searches along this predetermined direction, starting from the current point \mathbf{x}_k , aiming to identify a new point that possesses a lower function value. A step-size $\alpha_k \in \mathbb{R}$ is deduced by solving (exactly or non-exactly) the scalar minimization problem:

$$\underset{\alpha \in \mathbb{R}}{\text{minimize}} \quad F(\mathbf{x}_k + \alpha \mathbf{d}_k). \quad (2.3.2)$$

The increment is subsequently defined as $\mathbf{s}_k = \alpha_k \mathbf{d}_k$.

In contrast, in the trust region strategy, increment \mathbf{s}_k is computed by solving an inner optimization problem on the whole space \mathbb{R}^N as follows. First, a quadratic model m_k of F around \mathbf{x}_k is constructed from the information available at the current k -th iterate. This model is expected to provide a satisfactory approximation to the objective in a neighborhood of \mathbf{x}_k , and \mathbf{s}_k is defined as an inexact solution to the *trust region subproblem*:

$$\begin{aligned} & \underset{\mathbf{s} \in \mathbb{R}^N}{\text{minimize}} \quad m_k(\mathbf{s}) \\ & \text{subject to} \quad \|\mathbf{s}\|^2 \leq \delta_k, \end{aligned} \quad (2.3.3)$$

where the radius $\delta_k > 0$ of the trust region follows predefined rules.

It is worth noting that for large-scale optimization problems, the conventional line-search and trust region strategies may face challenges. In the context of line-search, determining the best direction \mathbf{d}_k could be challenging (e.g., Newton direction), or the decrease of the objective function too slow along iterations (e.g., gradient direction). For trust region strategies, solving the optimization problem (2.3.3) at each step may be computationally demanding.

2.3.1.2 Subspace methods

An effective approach for overcoming the curse of dimensionality amounts to search for a compromise between a one-dimensional line-search and the resolution of an inner problem on the entire space. The idea is to select \mathbf{s}_k within a restricted low-dimensional subspace \mathcal{S}_k of \mathbb{R}^N , so that, for every $k \in \mathbb{N}$,

$$\mathbf{s}_k \in \mathcal{S}_k, \quad (2.3.4)$$

leading to the introduction of *subspace methods* [Chouzenoux et al., 2013a, Elad et al., 2007]. Specifically, one defines the multidimensional step-size $\mathbf{u}_k \in \mathbb{R}^{M_k}$, where M_k is the dimension of \mathcal{S}_k , as an approximate solution to

$$\underset{\mathbf{u} \in \mathbb{R}^{M_k}}{\text{minimize}} \quad F(\mathbf{x}_k + \mathbf{D}_k \mathbf{u}), \quad (2.3.5)$$

where $\mathbf{D}_k = [\mathbf{d}_k^{(1)} | \dots | \mathbf{d}_k^{(M_k)}] \in \mathbb{R}^{N \times M_k}$ is the matrix of directions delineating \mathcal{S}_k with, for every $i \in \{1, \dots, M_k\}$, $\mathbf{d}_k^{(i)} \in \mathbb{R}^N$. One then updates \mathbf{x}_k as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_k. \quad (2.3.6)$$

An efficient technique for determining u_k , which will be discussed subsequently, is the Majorization–Minimization method proposed in [Chouzenoux et al., 2010]. Particularly, formulations like (2.3.6) have demonstrated their efficiency regarding computational time in various image processing tasks [Florescu et al., 2014, Zibulevsky and Elad, 2010].

It is noteworthy that advanced line-search methods can be considered as a specific instance of these subspace methods. For instance, the well-known nonlinear conjugate gradient (NLCG) [Fletcher and Reeves, 1964] and the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) [Liu and Nocedal, 1989] algorithms can be viewed as subspace techniques, where \mathbf{u}_k is computed with predefined rules and the subspace \mathcal{S}_k is defined as

$$(\text{NLCG}): \quad \mathcal{S}_k = \text{Span}\{\mathbf{g}_k, \mathbf{x}_k - \mathbf{x}_{k-1}\}, \quad (2.3.7)$$

and

$$(\text{L-BFGS}): \quad \mathcal{S}_k = \text{Span}\{\mathbf{g}_k, \mathbf{x}_k - \mathbf{x}_{k-1}, \dots, \mathbf{x}_{k-m+1} - \mathbf{x}_{k-m}, \\ \mathbf{g}_k - \mathbf{g}_{k-1}, \dots, \mathbf{g}_{k-m+1} - \mathbf{g}_{k-m}\}, \quad (2.3.8)$$

for a given $m \in \mathbb{N}^*$, and $\mathbf{g}_k = \nabla F(\mathbf{x}_k)$.

The subspace methodology can also apply to the trust region techniques. In such scenarios, the trust region subproblem (2.3.3) evolves to

$$\begin{aligned} & \underset{\mathbf{s} \in \mathbb{R}^N}{\text{minimize}} \quad m_k(\mathbf{s}) \\ & \text{subject to} \quad \|\mathbf{s}\|^2 \leq \delta_k \text{ and } \mathbf{s} \in \mathcal{S}_k, \end{aligned} \quad (2.3.9)$$

hence shrinking the inner problem dimensionality from N to M_k . This approach has proven effective across various challenges, including image processing tasks [Shultz et al., 1985, Wang et al., 2004, Wang

and Yuan, 2006, Yu et al., 2017].

2.3.1.3 Majorization-Minimization approaches

Majorization-minimization (MM) procedures have been widely used in various applications including signal and image processing and statistics. The appeal of these methods can be attributed to their genericity, their ability to yield algorithms with a minimal number of parameters requiring manual tuning, and their guarantees of robustness, even within non-convex frameworks.

The core idea of these methods revolves around creating a simple upper model (e.g., a quadratic form) of the objective at each point and iteratively updating this model to produce minimizing sequences. In this manuscript, we focus on MM strategies for differentiable functions, therefore we will stick to this setting in the following discussion. Note however that MM methods can be extended to non-differentiable functions, see for instance [Xu et al., 2016].

Definition 2.20. Let $F : \mathbb{R}^N \rightarrow \mathbb{R}$ be a differentiable function and $\mathbf{x}' \in \mathbb{R}^N$. A function $\mathbf{x} \mapsto Q(\mathbf{x}, \mathbf{x}')$ is said to be a *global tangent majorant* of F at \mathbf{x}' if, for every $\mathbf{x} \in \mathbb{R}^N$,

$$F(\mathbf{x}) \leq Q(\mathbf{x}, \mathbf{x}'), \quad (2.3.10)$$

$$F(\mathbf{x}') = Q(\mathbf{x}', \mathbf{x}'). \quad (2.3.11)$$

The Majorization-Minimization procedure to minimize F corresponds to the following iterative scheme:

$$\left\{ \begin{array}{l} \mathbf{x}_0 \in \mathbb{R}^N, \\ (\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} \in \underset{\mathbf{x} \in \mathbb{R}^N}{\operatorname{Argmin}} Q(\mathbf{x}, \mathbf{x}_k). \end{array} \right. \quad (2.3.12)$$

An illustration is given in Figure 2.4.

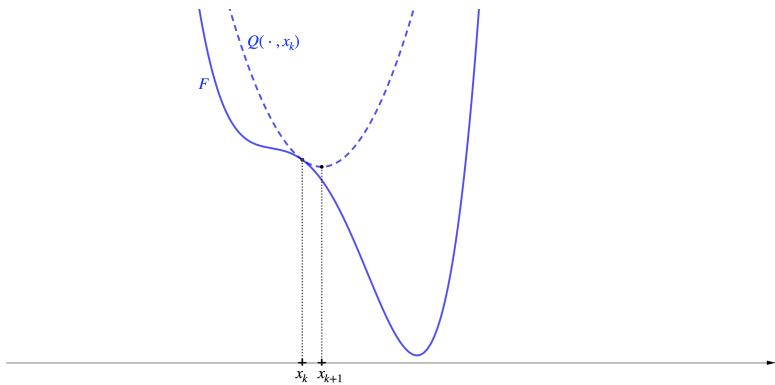


Figure 2.4: The function $\mathbf{x} \mapsto Q(\mathbf{x}, \mathbf{x}_k)$ is a majorant of F at the point \mathbf{x}_k . The next iterate \mathbf{x}_{k+1} is obtained by minimizing $Q(\cdot, \mathbf{x}_k)$.

In Theorem 2.21, we provide a sequential convergence result for the MM algorithm in its generic form (2.3.12). It is also worth noting that for many algorithms based on the MM principle, stronger convergence guarantees have been obtained using specific forms of majorants (e.g., quadratic) and the Kurdyka-Łojasiewicz property [Chouzenoux et al., 2013a, Chouzenoux et al., 2014, Phan and Gillis, 2023].

Theorem 2.21. (Convergence of the MM algorithm [Sun et al., 2017])

Assume the following conditions are satisfied:

- (A1) The sublevel set $\{\mathbf{x} \in \mathbb{R}^N \mid f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ is compact,
- (A2) F and $Q(\cdot, \mathbf{x})_{\mathbf{x} \in \mathbb{R}^N}$ are continuously differentiable,
- (A3) Q is continuous.

Then any limit point of $(\mathbf{x}_k)_{k \in \mathbb{N}}$ is a stationary point of F .

The effectiveness of the MM approach is largely determined by the quality of the approximation provided by the majorant and the ability to minimize it efficiently. Therefore, the choice of majorant at each iteration is crucial. We next review some of the most common majorants used in the literature and how to build them.

Affine Majorants

Affine majorants are the simplest type of majorants. They are defined as the first order Taylor's expansion of the function F at a given point \mathbf{x}' . In particular differentiable concave functions admit tangent affine majorants at any point \mathbf{x}' since the following inequality stands:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad f(\mathbf{x}) \leq f(\mathbf{x}') + \nabla f(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}'). \quad (2.3.13)$$

Quadratic Majorants

A majorizing strategy that applies to a larger class of functions is to use tangent quadratic majorants. A quadratic function satisfying conditions (2.3.10) and (2.3.11) at a point $\mathbf{x}' \in \mathbb{R}^N$ necessarily takes the form

$$Q(\mathbf{x}, \mathbf{x}') = F(\mathbf{x}') + \nabla F(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \mathbf{A}(\mathbf{x}')(\mathbf{x} - \mathbf{x}'), \quad (2.3.14)$$

with $\mathbf{A}(\mathbf{x}') \in \mathcal{S}_N^+$. In that case, $\mathbf{A}(\mathbf{x}')$ is called the *curvature matrix* of function Q at point \mathbf{x}' . When $\mathbf{A}(\mathbf{x}')$ is invertible, the minimization is explicitly and efficiently performed as

$$\operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^N} Q(\mathbf{x}, \mathbf{x}') = \mathbf{x}' - \mathbf{A}(\mathbf{x}')^{-1} \nabla F(\mathbf{x}'). \quad (2.3.15)$$

The resulting iterative MM scheme is known as the *Half Quadratic algorithm*, first introduced in [Geman and Reynolds, 1992]. When the curvature matrix is not invertible, one can resort to the pseudo-inverse of $\mathbf{A}(\mathbf{x}')$ to obtain a solution. A very large body of work have been devoted to the half quadratic algorithm and to variants of the quadratic Majorization-Minimization procedure, both in the field of image processing [Idier, 2001, Nikolova and Ng, 2005, Charbonnier et al., 1997, Chouzenoux et al., 2013b, Florescu et al., 2014] and machine learning [Du et al., 2012, He et al., 2013, Mairal, 2015].

Higher-Order Majorants

MM algorithms with higher-order majorants, particularly of order 3, have also been considered in the literature [Nesterov and Polyak, 2006]. These majorants take the form

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad Q(\mathbf{x}, \mathbf{x}') = F(\mathbf{x}') + \nabla F(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + (\mathbf{x} - \mathbf{x}')^\top \nabla^2 F(\mathbf{x}') (\mathbf{x} - \mathbf{x}') + \frac{L}{6} \|\mathbf{x} - \mathbf{x}'\|^3, \quad (2.3.16)$$

where L is the Lipschitz constant of $\nabla^2 F$. However, using cubic majorants may not always be advantageous due to the computational cost of minimizing them. In particular, the minimization of cubic majorants is not always tractable. For instance, the minimization of the majorant (2.3.16) is not tractable when F is the squared Euclidean norm. In that case, the minimization of the majorant (2.3.16) is equivalent to solving a cubic equation.

Techniques to build quadratic majorants

Here-below, we present some classical techniques to construct quadratic tangent majorants as these will be at the core of the thesis, particularly in Chapter 3.

We first observe that it is not always possible to construct a quadratic upper bound for any differentiable function. For instance, the function $x \mapsto x^4$ cannot be bounded by a quadratic function over the whole space. In the following, we delve into the construction of quadratic upper bounds for certain classes of functions.

Functions with a Lipschitz gradient form a category for which quadratic upper bounds can be constructed at any point. The construction for this type of function is based on the following well known lemma.

Lemma 2.22. (Descent Lemma [Bauschke and Combettes, 2011]) *Assume that $\psi : \mathbb{R}^N \rightarrow \mathbb{R}$ is a continuously differentiable function and that its gradient $\nabla \psi$ is Lipschitz continuous with constant L . For every $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{x}' \in \mathbb{R}^N$, the following inequality holds:*

$$\psi(\mathbf{x}') \leq \psi(\mathbf{x}) + \nabla \psi(\mathbf{x})^\top (\mathbf{x}' - \mathbf{x}) + \frac{L}{2} \|\mathbf{x}' - \mathbf{x}\|^2. \quad (2.3.17)$$

Example 2.23. Many functions used in optimization for image processing have a Lipschitz gradient, such as linear functions, squared Euclidean norm $\|\cdot\|^2$, or the smooth approximation of the ℓ^1 norm given by

$$\mathbf{x} = (x_n)_{1 \leq n \leq N} \mapsto \sum_{n=1}^N \sqrt{\delta + x_n^2},$$

where $0 \leq \delta \ll 1$. An illustration for this last example is given in Figure 2.5.

We have already discussed how concavity can be used to derive linear upper bounds. This property can also be used to find quadratic upper bounds for a specific class of functions. The following Lemma offers the advantage of providing majorants with non-constant, locally adapted curvature.

Lemma 2.24. [Chouzenoux et al., 2013b, Chan and Mulet, 1999] *Let $\psi : \mathbb{R} \rightarrow \mathbb{R}$. Assume that ψ satisfies the following hypotheses:*

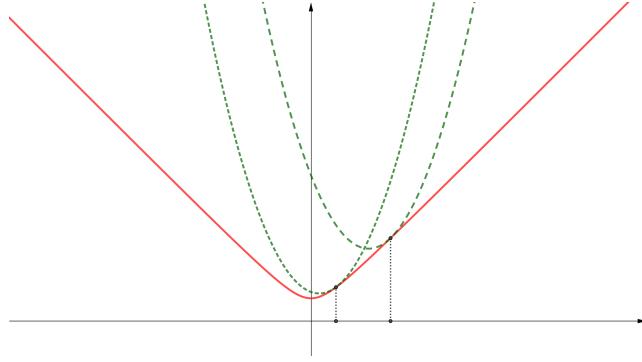


Figure 2.5: 1D illustration of the upper bound provided by Lemma 2.22 for the function $\psi(x) = \sqrt{1 + x^2}$. The function ψ (red curve) and its upper bounds at different points (green dotted lines) are shown.

- (i) ψ is differentiable,
- (ii) $\psi(\sqrt{\cdot})$ is concave,
- (iii) There exists $\bar{\omega} \geq 0$ such that, for every $t \in]0, +\infty[$, $0 \leq \psi'(t) \leq \bar{\omega}t$. Moreover

$$\lim_{\substack{t \rightarrow 0 \\ t \neq 0}} \psi'(t)/t \in \mathbb{R}. \quad (2.3.18)$$

Then the function defined for every $\mathbf{x} \in \mathbb{R}^N$ by $\Psi(\mathbf{x}) = \psi(\|\mathbf{x}\|)$ admits

$$x \mapsto \Psi(\mathbf{x}') + \nabla \Psi(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2} (\mathbf{x} - \mathbf{x}')^\top \mathbf{A}(\mathbf{x}') (\mathbf{x} - \mathbf{x}') \quad (2.3.19)$$

as a quadratic upper bound at the point $\mathbf{x}' \neq \mathbf{0}$, with

$$\mathbf{A}(\mathbf{x}') = \frac{\psi'(\|\mathbf{x}'\|)}{\|\mathbf{x}'\|} \mathbf{I}_N. \quad (2.3.20)$$

Example 2.25. The hypotheses of Lemma 2.24 are in particular verified for the functions of the form

$$(\forall t \in \mathbb{R}) \quad \psi(t) = (\delta + t^2)^{\beta/2}, \quad \delta > 0, \quad \beta \in [1, 2], \quad (2.3.21)$$

which provide a smoothed versions of $t \mapsto |t|^\beta$. Figure 2.6 illustrates the obtained upper bound. This example is particularly useful to derive a majorant of the smooth TV function.

We end this section with the following fundamental lemma, which will be used extensively in this manuscript. Given a function ψ for which we know a quadratic upper bound, this lemma allows us to provide an upper bound for the function ψ composed with a linear operator. This is a situation often encountered in image restoration where ψ is a simple function (e.g., quadratic) and the linear operator is the degradation operator (e.g., blur).

Lemma 2.26. (Composition with a linear operator) Let $\Psi: \mathbb{R}^M \rightarrow \mathbb{R}$ be of the form $\Psi(\mathbf{x}) = \psi(\mathbf{H}\mathbf{x} - \mathbf{y})$ with $\psi: \mathbb{R}^N \rightarrow \mathbb{R}$, $\mathbf{H} \in \mathbb{R}^{N \times M}$ and $\mathbf{y} \in \mathbb{R}^N$. We assume that, for every $\mathbf{u}' \in \mathbb{R}^N$, there exists

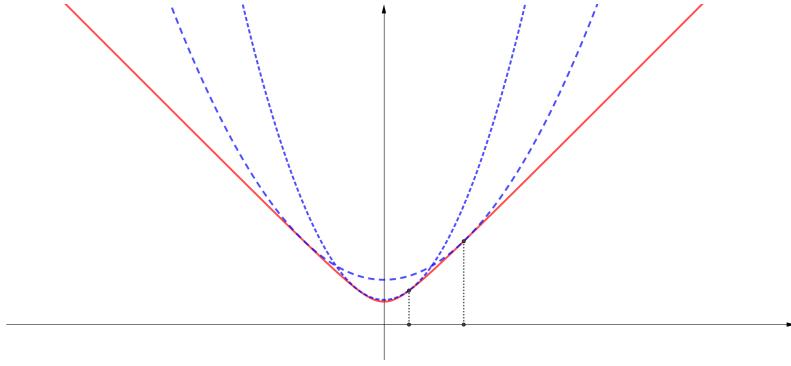


Figure 2.6: 1D Illustration of the upper bound provided by Lemma 2.24, for the function $\psi(x) = \sqrt{1+x^2}$. We plotted ψ (red curve) and its upper bound at different points (blue dotted lines). This upper bound is “tighter” than the one observed in Figure 2.5.

$\mathbf{A}(\mathbf{u}') \in \mathcal{S}_N^+$ such that

$$\mathbf{u} \mapsto \psi(\mathbf{u}') + \nabla\psi(\mathbf{u}')^\top(\mathbf{u} - \mathbf{u}') + (\mathbf{u} - \mathbf{u}')^\top \mathbf{A}(\mathbf{u}')(\mathbf{u} - \mathbf{u}') \quad (2.3.22)$$

is a quadratic upper bound for ψ . Then, for every $\mathbf{x}' \in \mathbb{R}^M$,

$$\mathbf{x} \mapsto \Psi(\mathbf{x}') + \nabla\Psi(\mathbf{x}')^\top(\mathbf{x} - \mathbf{x}') + \frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \mathbf{B}(\mathbf{x}')(\mathbf{x} - \mathbf{x}') \quad (2.3.23)$$

is a quadratic upper bound for Ψ at point \mathbf{x}' , with

$$\mathbf{B}(\mathbf{x}') = \mathbf{H}^\top \mathbf{A}(\mathbf{H}\mathbf{x}' - \mathbf{y}) \mathbf{H} \in \mathbb{R}^{M \times M}. \quad (2.3.24)$$

To conclude our introduction to MM methods, we spotlight the renowned Expectation-Maximization (EM) algorithm as an example of MM algorithm. Notably, the EM algorithm will be the centerpiece of Chapter 6.

An example of MM algorithm: the Expectation-Maximisation algorithm

The Expectation-Minimization (EM) algorithm is an iterative optimization technique widely employed in the context of statistical models, particularly when dealing with latent variables or incomplete data. We describe it here in the case of a mixture model. Assume that $(\mathbf{z}_n)_{1 \leq n \leq N} \in (\mathbb{R}^d)^N$ are N independent observations. The posterior distribution corresponds to

$$p(\mathbf{z}_n \mid \boldsymbol{\pi}, \boldsymbol{\Theta}) = \sum_{k=1}^K \pi_k p(\mathbf{z}_n \mid \boldsymbol{\theta}_k), \quad (2.3.25)$$

where mixing coefficients are represented by the vector $\boldsymbol{\pi} = (\pi_k)_{1 \leq k \leq K} \in \Delta_K$ and $(\boldsymbol{\theta}_k)_{1 \leq k \leq K} \in \mathbb{R}^m$ are K parameters to be estimated. The goal is to maximize the log-likelihood function:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\pi}, \boldsymbol{\Theta}) &= \ln p(\mathbf{Z} \mid \boldsymbol{\pi}, \boldsymbol{\Theta}) \\ &= \sum_{n=1}^N \ln \left(\sum_{k=1}^K \pi_k p(\mathbf{z}_n \mid \boldsymbol{\theta}_k) \right).\end{aligned}\quad (2.3.26)$$

Directly optimizing this log-likelihood can be computationally intractable. Instead, the EM algorithm iteratively maximizes a lower bound on the log-likelihood function, generating sequences $(\boldsymbol{\pi}^{(\ell)})_{\ell \in \mathbb{N}}$ and $(\boldsymbol{\Theta}^{(\ell)})_{\ell \in \mathbb{N}}$ that converge in terms of log-likelihood values. The lower bound at iteration ℓ is derived from the Jensen's inequality:

$$\mathcal{L}(\boldsymbol{\pi}, \boldsymbol{\Theta}) \geq S\left((\boldsymbol{\pi}, \boldsymbol{\Theta}); (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\Theta}^{(\ell)})\right), \quad (2.3.27)$$

where

$$S\left((\boldsymbol{\pi}, \boldsymbol{\Theta}); (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\Theta}^{(\ell)})\right) = \sum_{n=1}^N \sum_{k=1}^K u_{n,k}^{(\ell)} \ln \left(\frac{\pi_k p(\mathbf{z}_n \mid \boldsymbol{\theta}_k)}{u_{n,k}^{(\ell)}} \right), \quad (2.3.28)$$

and $(u_{n,k}^{(\ell)})_{\substack{1 \leq n \leq N \\ 1 \leq k \leq K}}$ are defined as

$$u_{n,k}^{(\ell)} = \frac{\pi_k^{(\ell)} p(\mathbf{z}_n \mid \boldsymbol{\theta}_k^{(\ell)})}{\sum_{j=1}^K \pi_j^{(\ell)} p(\mathbf{z}_n \mid \boldsymbol{\theta}_j^{(\ell)})}. \quad (2.3.29)$$

The algorithm alternates between the following two steps:

Expectation step. In the E-step, ones computes the “responsibilities” $(u_{n,k}^{(\ell)})_{\substack{1 \leq n \leq N \\ 1 \leq k \leq K}}$. These responsibilities represent the probability that a given data point belongs to a particular cluster, based on the current parameter estimates. The definition in (2.3.29) ensures the tightness of the lower bound (2.3.27), i.e., $S\left((\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\Theta}^{(\ell)}); (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\Theta}^{(\ell)})\right) = \mathcal{L}(\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\Theta}^{(\ell)})$.

Maximization step. The M-step involves maximizing the function obtained in the E-step with respect to the model parameters, $\boldsymbol{\Theta}$ and $\boldsymbol{\pi}$. The update for the mixing coefficients is

$$(\forall k \in \{1, \dots, K\}) \quad \pi_k^{(\ell+1)} = \frac{1}{N} \sum_{n=1}^N u_{n,k}^{(\ell)}. \quad (2.3.30)$$

The update for the model parameters, $\boldsymbol{\Theta}$, will then depend on the specific nature of the posteriors.

2.3.2 Splitting algorithms

In this section, we present the class of so-called splitting algorithms, which are applicable to non-smooth and constrained optimization. These methods will appear regularly along the manuscript, as comparative algorithms in Chapter 3 or as subroutines for the algorithms we propose in Chapter 4.

In our previous discussions, it was observed that variational formulations frequently comprise multiple terms, each representing specific attributes of the desired solution. These components in the objective can exhibit a range of mathematical characteristics, such as smoothness, convexity, or the existence of an analytical expression of their proximity operator. Attempting a direct minimization of the objective function using methods like gradient descent is often infeasible.

The essence of proximal splitting methods lies in their ability to decompose the objective function into simpler, more manageable subcomponents. By addressing each term individually, these strategies pave the way to addressing a wide range of variational problems both accurately and efficiently.

2.3.2.1 Forward-Backward algorithm

Consider the following problem:

Problem 2.27. Let $f: \mathbb{R}^N \rightarrow \mathbb{R}$ be a differentiable function having a β -Lipschitzian gradient with $\beta \in (0, +\infty)$, and $g \in \Gamma_0(\mathbb{R}^N)$ be such that $f(\mathbf{x}) + g(\mathbf{x}) \rightarrow +\infty$ as $\|\mathbf{x}\| \rightarrow +\infty$. The problem is then

$$\underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad f(\mathbf{x}) + g(\mathbf{x}). \quad (2.3.31)$$

It is easy to see that the solutions to Problem (2.3.31) are characterized by the fixed point equation, for any $\gamma > 0$:

$$\mathbf{x} = \text{prox}_{\gamma g}(\mathbf{x} - \gamma \nabla f(\mathbf{x})). \quad (2.3.32)$$

This suggests performing the following iterations:

$$(\forall n \in \mathbb{N}) \quad \mathbf{x}_{n+1} = \underbrace{\text{prox}_{\gamma_n g}}_{\text{backward step}} \underbrace{(\mathbf{x}_n - \gamma_n \nabla f(\mathbf{x}_n))}_{\text{forward step}} \quad (2.3.33)$$

for values of the step-size parameter γ_n within a suitable bounded interval. This corresponds to the Forward-Backward algorithm, summarized in Algorithm 1, where a relaxation parameter λ_n has been introduced.

Algorithm 1: Forward-backward algorithm [Chen and Rockafellar, 1997]

```

Set sequences  $(\gamma_n)_{n \in \mathbb{N}}$  and  $(\lambda_n)_{n \in \mathbb{N}}$ ,
Initialize  $\mathbf{x} \in \mathbb{R}^N$ ,
for  $n = 0, 1, 2, \dots$  do
   $\mathbf{y}_n = \mathbf{x}_n - \gamma_n \nabla f(\mathbf{x}_n)$ ,
   $\mathbf{x}_{n+1} = \mathbf{x}_n + \lambda_n (\text{prox}_{\gamma_n g}(\mathbf{y}_n) - \mathbf{x}_n)$ .

```

Remark 2.28.

- When $g = 0$, Equation (2.3.32) becomes the *descent gradient method* for minimizing a function with a Lipschitz continuous gradient:

$$(\forall n \in \mathbb{N}) \quad \mathbf{x}_{n+1} = \mathbf{x}_n - \gamma_n \nabla f(\mathbf{x}_n) \quad (2.3.34)$$

- When $g = \iota_C$ with C a nonempty, closed, convex set, Equation (2.3.32) becomes the *projected gradient method*:

$$(\forall n \in \mathbb{N}) \quad \mathbf{x}_{n+1} = \text{proj}_C(\mathbf{x}_n - \gamma_n \nabla f(\mathbf{x}_n)). \quad (2.3.35)$$

- Conversely, when $f = 0$, Equation (2.3.32) simplifies to the so-called *proximal point algorithm*:

$$(\forall n \in \mathbb{N}) \quad \mathbf{x}_{n+1} = \text{prox}_{\gamma_n g} \mathbf{x}_n, \quad (2.3.36)$$

mainly used for minimizing a nondifferentiable function.

Thus, the Forward-Backward algorithm can be viewed as a fusion of these foundational methods. In addition, the Forward-Backward update is reminiscent of the Majorization-Minimization principle. Indeed, when the step-size satisfies for every $n \in \mathbb{N}$, $\gamma_n \in (0, \frac{1}{L})$, the descent Lemma 2.22 yields the following majorant of $f + g$ at \mathbf{x}_n :

$$\mathbf{x} \mapsto f(\mathbf{x}_n) + \nabla f(\mathbf{x}_n)^\top (\mathbf{x} - \mathbf{x}_n) + \frac{1}{2\gamma_n} \|\mathbf{x} - \mathbf{x}_n\|^2 + g(\mathbf{x}). \quad (2.3.37)$$

Minimizing (2.3.37) is equivalent to the Forward-Backward step in (2.3.33).

The convergence properties of the Forward-Backward algorithm are stated in the following proposition.

Proposition 2.29. (Convergence of the Forward-Backward algorithm [Combettes and Wajs, 2005, Thm. 3.4]) Assume that the sequence of parameters (γ_n) satisfies $\gamma_n \in [\varepsilon, \frac{2}{\beta} - \varepsilon]$ and that $\lambda_n \in [\varepsilon, 1]$ for some $\varepsilon \in (0, \min\{1, 1/\beta\})$. Then, every sequence $(\mathbf{x}_n)_{n \in \mathbb{N}}$ generated by the Forward-Backward algorithm converges to a solution to Problem (2.3.31).

It is worth noting that the scope of the Forward-Backward scheme application goes beyond what is covered in this thesis. Expanded versions of the FB scheme have introduced a variable metric into the proximal operator [Chen and Rockafellar, 1997, Combettes and Vũ, 2014] and allow for inexact computation of this operator [Villa et al., 2013]. Furthermore, building on the KL inequality, studies have established the method convergence even in nonconvex scenarios [Chouzenoux et al., 2014].

2.3.2.2 Dual Forward-Backward algorithm

We now turn our attention to the Dual Forward-Backward algorithm, which will appear in Chapter 3 as a subroutine for our comparisons. This method is particularly efficient when evaluating the proximal operator of a sum comprising two possibly nonsmooth functions.

Problem 2.30. Let $h \in \Gamma_0(\mathbb{R}^N)$, $g \in \Gamma_0(\mathbb{R}^M)$, and $\mathbf{L} \in \mathbb{R}^{M \times N}$ be such that $\text{int}(\text{dom } g - \mathbf{L}(\text{dom } h)) \neq \emptyset$. Let $\mathbf{x}' \in \mathbb{R}^N$. The problem is then

$$\underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad g(\mathbf{L}\mathbf{x}) + h(\mathbf{x}) + \frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2}. \quad (2.3.38)$$

The unique solution to (2.3.38) can be deduced from its Fenchel-Moreau-Rockafellar dual, which can be solved through forward-backward splitting. The Dual Forward-Backward algorithm is given in Algorithm 2

Proposition 2.31. (Convergence of the Dual Forward-Backward algorithm [Combettes et al., 2011]) Set $\varepsilon \in (0, \min\{1, \frac{1}{\|\mathbf{L}\|^2}\})$. Assume that the sequence of parameters (γ_n) satisfies $\gamma_n \in [\varepsilon, \frac{2}{\|\mathbf{L}\|^2} - \varepsilon]$ and that $\lambda_n \in [\varepsilon, 1]$. Then, the sequence $(\mathbf{x}_n)_{n \in \mathbb{N}}$ generated by the Dual Forward-Backward Algorithm converges to the unique solution to (2.3.38).

Algorithm 2: Dual Forward-Backward Algorithm [Combettes et al., 2011]

```

Initialize  $\mathbf{u}_0 \in \mathbb{R}^M$ 
for  $n = 0, 1, 2, \dots$  do
     $\mathbf{x}_n = \text{prox}_h(\mathbf{x}' - \mathbf{L}^\top \mathbf{u}_n),$ 
     $\mathbf{u}_{n+1} = \mathbf{u}_n + \lambda_n \text{prox}_{\gamma_n g^*}(\mathbf{u}_n + \gamma_n \mathbf{L} \mathbf{x}_n) - \mathbf{u}_n.$ 

```

2.3.2.3 Primal-dual Splitting Algorithm

Condat and Vũ introduced a primal-dual splitting method in their respective works [Condat, 2012] and [Vũ, 2013]. A comprehensive overview of primal-dual methods is provided in [Komodakis and Pesquet, 2015].

Problem 2.32. Let $f \in \Gamma_0(\mathbb{R}^N)$ be differentiable with a Lipschitzian gradient, $g \in \Gamma_0(\mathbb{R}^M)$, $h \in \Gamma_0(\mathbb{R}^N)$, and let $\mathbf{L} \in \mathbb{R}^{M \times N}$ be such that $\text{int}(\text{dom } g - \mathbf{L}(\text{dom } h)) \neq \emptyset$. Let $\mathbf{x}' \in \mathbb{R}^N$. The problem is then:

$$\underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad f(\mathbf{x}) + g(\mathbf{L}\mathbf{x}) + h(\mathbf{x}). \quad (2.3.39)$$

This method applies a forward-backward scheme to dissociate the differentiable term f from the nonsmooth function h . Furthermore, duality is employed to decouple g and \mathbf{L} . Algorithm 3 presents a version of the primal-dual algorithm, where one makes the assumption of error-free computations for the proximal operators. Notably, when f is a zero function, this algorithm coincides with the one proposed by Chambolle and Pock [Chambolle and Pock, 2011]. [Condat, 2012, Theorem 3.1] provides a convergence result for the primal sequence in Algorithm 3.

Algorithm 3: Primal-dual Splitting Algorithm [Condat, 2012][Vũ, 2013]

```

Parameters:  $\sigma \in (0, +\infty)$ ,  $\tau \in (0, +\infty)$ 
Initialize:  $(\mathbf{x}_0, \mathbf{v}_0) \in \mathbb{R}^M \times \mathbb{R}^N$ 
for  $k = 0, 1, \dots$  do
     $\mathbf{y}_k = \text{prox}_{\tau h}(\mathbf{x}_k - \tau \nabla f(\mathbf{x}_k) + \mathbf{L}^\top \mathbf{v}_k)$ 
     $\mathbf{u}_k = \text{prox}_{\sigma g^*}(\mathbf{v}_k + \sigma \mathbf{L}(2\mathbf{y}_k - \mathbf{x}_k))$ 
     $\mathbf{x}_{k+1} = \lambda_k \mathbf{y}_k + (1 - \lambda_k) \mathbf{x}_k$ 
     $\mathbf{v}_{k+1} = \lambda_k \mathbf{u}_k + (1 - \lambda_k) \mathbf{v}_k$ 

```

2.3.3 Alternating minimization

Finally, we provide an overview of alternating minimization algorithms, ranging from the classical Gauss-Seidel method to recent advancements. These methods are especially suited for solving optimization problems with a composite structure, that is, problems that can be broken down into simpler subproblems. We will delve into this class of methods on multiple occasions throughout this manuscript, specifically in Chapters 4, 5, and 6.

Problem 2.33. Let $m \in \mathbb{N}^*$, $(n_1, \dots, n_m) \in (\mathbb{N}^*)^m$ and $N = \sum_{i=1}^m n_i$. Let $F: \mathbb{R}^{n_1} \times \dots \times \mathbb{R}^{n_m} \rightarrow \mathbb{R} \cup \{+\infty\}$ be proper. The problem is then:

$$\underset{\boldsymbol{x} \in \mathbb{R}^{n_1} \times \dots \times \mathbb{R}^{n_m}}{\text{minimize}} F(\boldsymbol{x}). \quad (2.3.40)$$

2.3.3.1 Block-coordinate method

The *nonlinear Gauss-Seidel* algorithm, also known as the *Block Coordinate* method, generates a sequence $(\boldsymbol{x}^{(\ell)})_{\ell \in \mathbb{N}} = (\boldsymbol{x}_1^{(\ell)}, \dots, \boldsymbol{x}_m^{(\ell)}) \in (\mathbb{R}^{n_1} \times \dots \times \mathbb{R}^{n_m})^{\mathbb{N}}$ corresponding to Algorithm 4. The idea is to update the variables one after the other, in a cyclic fashion, through an exact minimization of the partial function at the current iterate. The subsequential convergence of this algorithm is guaranteed under restrictive assumptions on the partial functions of F . The result is stated in Theorem 2.34.

Algorithm 4: Block Coordinate algorithm

```

Initialization: choose initial point  $\boldsymbol{x}^{(0)} = (\boldsymbol{x}_1^{(0)}, \dots, \boldsymbol{x}_m^{(0)})$ 
for  $\ell = 0, 1, \dots$  do
  for  $i = 1, \dots, m$  do
     $\boldsymbol{x}_i^{(\ell+1)} \in \underset{\boldsymbol{\xi} \in \mathbb{R}^{n_i}}{\text{argmin}} F(\boldsymbol{x}_1^{(\ell+1)}, \dots, \boldsymbol{x}_{i-1}^{(\ell+1)}, \boldsymbol{\xi}, \boldsymbol{x}_{i+1}^{(\ell)}, \dots, \boldsymbol{x}_m^{(\ell)})$ 

```

Theorem 2.34. (Subsequential convergence [Tseng, 2001, Theorem 4.1])

Assume the following conditions are satisfied:

- (A1) The level set $X^{(0)} = \{\boldsymbol{x} \in \mathbb{R}^n \mid f(\boldsymbol{x}) \leq f(\boldsymbol{x}^{(0)})\}$ is compact and F is continuous on $X^{(0)}$.
- (A2) For every $i \in \{1, \dots, m\}$, for every $\boldsymbol{x} \in \mathbb{R}^N$, the partial function

$$\boldsymbol{\xi} \mapsto F(\boldsymbol{x}_1, \dots, \boldsymbol{x}_{i-1}, \boldsymbol{\xi}, \boldsymbol{x}_{i+1}, \dots, \boldsymbol{x}_m)$$

has a unique minimizer.

Then the sequence $(\boldsymbol{x}^{(\ell)})_{\ell \in \mathbb{N}}$ generated by Algorithm 4 is bounded and any of its cluster points is a coordinate-wise minimizer of F on \mathbb{R}^n , i.e. any cluster point $\bar{\boldsymbol{x}} = (\bar{\boldsymbol{x}}_1, \dots, \bar{\boldsymbol{x}}_m)$ satisfies

$$(\forall \boldsymbol{x} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_m) \in \mathbb{R}^n), (\forall i \in \{1, \dots, m\}) \\ F(\bar{\boldsymbol{x}}_0, \dots, \bar{\boldsymbol{x}}_{i-1}, \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{x}}_{i+1}, \dots, \bar{\boldsymbol{x}}_m) \leq F(\bar{\boldsymbol{x}}_0, \dots, \bar{\boldsymbol{x}}_{i-1}, \boldsymbol{x}_i, \bar{\boldsymbol{x}}_{i+1}, \dots, \bar{\boldsymbol{x}}_m). \quad (2.3.41)$$

2.3.3.2 Proximal alternating minimization

While the Block Coordinate method has been demonstrated as an effective tool, it might not always be feasible to minimize the partial functions in an exact manner. Moreover, the convergence result presented in Theorem 2.34 is limited to subsequential convergence with restrictive assumptions on the cost function. However, with advancements in the field of optimization, especially with the introduction of proximal tools and convergence under the KL assumption, remedies to these shortcomings have been proposed.

The *Proximal Alternating Minimization* (PAM) algorithm, introduced in [Attouch et al., 2010], can be perceived as a regularized iteration of the Block Coordinate algorithm in which the variables are updated using a proximal point scheme rather than an exact minimization. PAM addresses Problem 2.33 for a specific decomposition of the objective function F ,

$$\underset{\mathbf{x} \in \mathbb{R}^n}{\text{minimize}} \quad F(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^m g_i(\mathbf{x}_i). \quad (2.3.42)$$

where $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is the coupling term and, for every $i \in \{1, \dots, m\}$, $g_i: \mathbb{R}^{n_i} \rightarrow \mathbb{R} \cup \{+\infty\}$ are the separable terms. The PAM algorithm generates iterates $(\mathbf{x}^{(\ell)})_{\ell \in \mathbb{N}}$ according to Algorithm 5, where we adopt the short notation

$$(\forall i \in \{1, \dots, m\}) \quad (\forall \ell \in \mathbb{N}^*) \quad f_i^{(\ell)}(\mathbf{x}_i) = f(\mathbf{x}_1^{(\ell+1)}, \dots, \mathbf{x}_{i-1}^{(\ell+1)}, \mathbf{x}_i, \mathbf{x}_{i+1}^{(\ell)}, \dots, \mathbf{x}_m^{(\ell)}). \quad (2.3.43)$$

The convergence result derived in [Attouch et al., 2010] is stated in Theorem 2.35.

Algorithm 5: Proximal Alternating Minimization (PAM)

Initialization: choose initial point $(\mathbf{x}_1^{(0)}, \dots, \mathbf{x}_m^{(0)})$ and sequences
 $(\gamma_i^{(\ell)})_{1 \leq i \leq m} \in ((0, +\infty)^m)^{\mathbb{N}}$
for $\ell = 0, 1, \dots$ **do**
 for $i = 1, \dots, m$ **do**
 $\mathbf{x}_i^{(\ell+1)} \in \underset{\xi \in \mathbb{R}^{n_i}}{\operatorname{argmin}} f_i^{(\ell)}(\xi) + g_i(\xi) + \frac{1}{2\gamma_i^{(\ell)}} \|\xi - \mathbf{x}_i^{(\ell)}\|^2$

Theorem 2.35. (Convergence of the PAM algorithm [Attouch et al., 2010])

Assume the following conditions are satisfied:

- (A1) The objective function F is lower semi-continuous and coercive.
- (A2) f is continuously differentiable and ∇f is Lipschitz continuous on bounded subsets of \mathbb{R}^n .
- (A3) For every $i \in \{1, \dots, m\}$, g_i is a proper, lower semi-continuous function.
- (A4) There exist $\gamma_- > 0$ and $\gamma_+ > 0$ such that, for every $\ell \in \mathbb{N}$, for every $i \in \{1, \dots, m\}$,
 $\gamma_i^{(\ell)} \in [\gamma_-, \gamma_+]$.
- (A5) The objective function F satisfies the Kurdyka-Łojasiewicz property at each point of its domain.

Then any sequence $(\mathbf{x}^{(\ell)})_{\ell \in \mathbb{N}}$ generated by Algorithm 5 is of finite length and converges to a critical point of F . Moreover the sequence converges locally to a global minimizer of F .

In a similar spirit as the PAM algorithm, the *Proximal Alternating Linearized Minimization* (PALM) algorithm, introduced in [Bolte et al., 2014] and considered in [Chouzenoux et al., 2014] with a variable metric, is a variant of the Block Coordinate method in which the partial functions are minimized using a Forward-Backward step (also referred in the following as prox-linearized step).

2.3.3.3 Mixed alternating minimization algorithm

In [Xu and Yin, 2013], the authors propose a new block coordinate method for minimizing the objective function in (2.3.42). We refer to this methods as the *Mixed Alternating Minimization* (MAM) algorithm. Their main contribution is to allow different types of steps on each block with respect to the properties of the partial function: the minimization of the partial function may be direct, proximal or prox-linear. A similar strategy is conducted in [Foare et al., 2019].

At each step, we consider three different possible updates:

$$(Original) \quad \mathbf{x}_i^{(\ell+1)} = \operatorname{argmin}_{\mathbf{x}_i} f_i^{(\ell)}(\mathbf{x}_i) + g_i(\mathbf{x}_i); \quad (a)$$

$$(Proximal) \quad \mathbf{x}_i^{(\ell+1)} = \operatorname{argmin}_{\mathbf{x}_i} f_i^{(\ell)}(\mathbf{x}_i) + \frac{\gamma_i^{(\ell)}}{2} \|\mathbf{x}_i - \mathbf{x}_i^{(\ell)}\|^2 + g_i(\mathbf{x}_i); \quad (b)$$

$$(Prox-linear) \quad \mathbf{x}_i^{(\ell+1)} = \operatorname{argmin}_{\mathbf{x}_i} \langle \nabla f_i^{(\ell)}, \mathbf{x}_i - \mathbf{x}_i^{(\ell)} \rangle + \frac{\gamma_i^{(\ell)}}{2} \|\mathbf{x}_i - \mathbf{x}_i^{(\ell)}\|^2 + g_i(\mathbf{x}_i), \quad (c)$$

where, for every $i \in \{1, \dots, m\}$, $(\gamma_i^{(\ell)})_{\ell \in \mathbb{N}} \in (0, +\infty)^{\mathbb{N}}$. The corresponding procedure is given in Algorithm 6. Strong convergence guarantees are established in [Xu and Yin, 2013] relying on the KL inequality. The result is stated in Theorem 2.36. Note that the convergence requires the objective function to be multi-convex, which is a stronger assumption than the one required for the PAM algorithm. While the assumptions of Theorem 2.36 will be satisfied in the works presented in this thesis, it is worth mentionning more recent works that have relaxed some of these. In [Chouzenoux et al., 2023], the authors propose a variant of the MAM algorithm that does not require the objective function to be multi-convex. They also consider the case when the block-separable terms are not necessarily convex.

In [Phan and Gillis, 2023], a unifying framework for the MAM algorithm is introduced. By noting that exact, proximal, and prox-linearized updates can be viewed as specific instances of the Majorization-Minimization scheme (where, at each iteration $\ell \in \mathbb{N}$ and for each block $i \in \{1, \dots, m\}$, a majorant of the differentiable function $f_i^{(\ell)}$ is utilized), the authors present a more versatile algorithm which offers enhanced flexibility in constructing updates for each block. Fundamentally, the authors demonstrate that any MM update which satisfies a specific descent condition (for instance satisfied when using strongly convex majorants) ensures the convergence of the algorithm.

Algorithm 6: Mixed Alternating Minimization (MAM)

```

Initialization: choose initial point  $(\mathbf{x}_1^{(0)}, \dots, \mathbf{x}_m^{(0)})$ 
for  $\ell = 0, 1, \dots$  do
  for  $i \in \{1, \dots, s\}$  do
     $\mathbf{x}_i^{(\ell+1)} \leftarrow$  (a), (b) or (c).

```

Theorem 2.36. (Convergence of the MAM algorithm [Xu and Yin, 2013])

Assume that, for a fixed $i \in \{1, \dots, m\}$, block i is always updated using the same scheme among (a), (b) and (c), at every iteration $\ell \in \mathbb{N}^*$. Let \mathcal{I}_a , \mathcal{I}_b and \mathcal{I}_c denote the set of blocks updated by (a), (b) and (c) respectively. Moreover, assume the following conditions are satisfied:

- (A1) The objective function F is lower semi-continuous and coercive.
- (A2) f is multi-convex, i.e., convex with respect to each block.
- (A3) f is continuously differentiable, ∇f is Lipschitz continuous on bounded subsets of \mathbb{R}^n and, for every $\ell \in \mathbb{N}^*$,

- if $i \in \mathcal{I}_a$, $f_i^{(\ell)} + g_i$ is strongly convex;
- if $i \in \mathcal{I}_c$, $\nabla f_i^{(\ell)}$ is Lipschitz continuous with constant $L_i^{(\ell)}$ satisfying

$$\underline{L}_i \leq L_i^{(\ell)} \leq \overline{L}_i,$$

where $(\underline{L}_i, \overline{L}_i) \in (0, +\infty)^2$.

- (A4) For every $i \in \{1, \dots, m\}$, g_i is a proper, lower semi-continuous function.

- (A5) The sequences of parameters are such that

- There exist $(\gamma_-, \gamma_+) \in (0, +\infty)^2$ such that, for every $\ell \in \mathbb{N}$, for every $i \in \mathcal{I}_b$, $\gamma_i^{(\ell)} \in [\gamma_-, \gamma_+]$.
- There exist $\gamma_- \in (0, +\infty)$ such that, for every $\ell \in \mathbb{N}$, for all $i \in \mathcal{I}_c$, $\gamma_i^{(\ell)} \in \left[\gamma_-, \frac{2}{L_i^{(\ell)}}\right]$.

- (A6) The objective function F satisfies the Kurdyka-Łojasiewicz property at each point of its domain.

Then any sequence $(\mathbf{x}^{(\ell)})_{\ell \in \mathbb{N}}$ generated by Algorithm 6 is of finite length and converges to a critical point of F . Moreover the sequence converges locally to a global minimizer of F .

Conclusion

In summary, this chapter has established the foundations for understanding the principles of image restoration and classification through variational approaches. As we have observed, depending on the application and the specific modeling techniques employed, the functions to be minimized may exhibit distinct mathematical characteristics. This requires the design of specialized algorithms capable of handling these various properties. The forthcoming chapters will expand upon these topics, employing the introduced tools to further explore the mathematical and algorithmic aspects of image restoration and classification.

- Chapter 3 -

A local MM subspace method for solving constrained variational problems in image recovery

This chapter is based on the following article:

E. Chouzenoux, S. Martin, J.-C. Pesquet, A local MM subspace method for solving constrained variational problems in image recovery, *Journal of Mathematical Imaging and Vision*, 2022.

We introduce a new Penalized Majorization-Minimization Subspace algorithm (P-MMS) for solving smooth, constrained optimization problems. In short, our approach consists of embedding a subspace algorithm in an inexact exterior penalty procedure. The subspace strategy, combined with a Majorization-Minimization step-size search, takes great advantage of the smoothness of the penalized cost function, while the penalty method allows to handle a wide range of constraints. The main drawback of exterior penalty approaches, namely ill-conditioning for large values of the penalty parameter, is overcome by using a trust-region-like technique. The convergence of the resulting algorithm is analyzed. Numerical experiments carried out on two large-scale image recovery applications demonstrate that, compared with state-of-the-art algorithms, the proposed method performs well in terms of computational time.

3.1 Introduction

Many challenges in image processing can be addressed by solving constrained optimization problems. These problems may be difficult to solve numerically in reasonable times because of their high dimension and of the involved constraints. These constraints may play diverse and crucial roles as they enforce prior knowledge about the solution. For example, constraints may have a regularization effect [Combettes and Pesquet, 2004], promote sparsity [Bahmani et al., 2013], ensure consistency with the noise model through a data-fit term [Afonso et al., 2010a, Harizanov et al., 2013], or impose the fulfillment of geometrical properties as in deformable image matching [Musse et al., 2001].

In this chapter, we propose a novel method for solving large-scale differentiable constrained problems formulated as

$$\underset{\boldsymbol{x} \in C}{\text{minimize}} \quad F(\boldsymbol{x}), \tag{3.1.1}$$

where $F: \mathbb{R}^N \rightarrow \mathbb{R}$ is a differentiable cost function and $C \subseteq \mathbb{R}^N$. In particular, our approach allows to tackle image recovery problems of high dimension (typically, at least 10^9 variables) efficiently. Although

such problems are often formulated as the minimization of a nondifferentiable objective function – being the sum of a differentiable data-fidelity term and a nondifferentiable regularization term, e.g. ℓ_1 or total variation semi-norms – satisfying (and sometimes better) results can be obtained by adopting a smooth formulation, yielding problems expressed as (3.1.1). Let us mention in particular two common methods to circumvent the non-smoothness possibly arising in the regularization term. The first one consists of using a smoothed approximation to the regularization function, which usually does not alter the recovery performance and can sometimes avoid undesirable effects such as staircasing [Nikolova, 2004]. The second possibility is to re-express the regularization term in the objective as a constraint, in the spirit of Ivanov approach [Ivanov et al., 2013].

As discussed in the overview Chapter in Section 2.3.1.2, differentiability assumption gives us access to a wide range of algorithms for solving Problem (3.1.1) when $C = \mathbb{R}^N$, in particular subspace methods. Interestingly, subspace methods can also be used to tackle *constrained* differentiable optimization problems. In particular, the case of linear constraints can be handled efficiently by relying on specific techniques, similar to the ones used in full-space trust region methods [Murtagh and Saunders, 1978] or unidirectional line-search methods [Coleman and Li, 1996]. For instance, active-set methods [Nocedal and Wright, 2006, Chap. 16] have been used for line-search strategies, defining \mathbf{x}_{k+1} via a direction computed from a reduced problem associated only with the inactive variables. Based on these ideas, Shanno and Marsten [Shanno and Marsten, 1982] proposed a conjugate gradient method for solving problems with linear equality constraints and bound constraints. Byrd et al. [Byrd et al., 1995] adapted the L-BFGS method [Liu and Nocedal, 1989] to minimize a smooth function under bound constraints leading to the extensively used L-BFGS-B algorithm. Furthermore, rescaling strategies [Conn et al., 2000, Sec. 6.7] have been applied to subspace trust region methods for bound-constrained optimization. For instance, in [Branch et al., 1999], Branch et al. performed a change of variables to eliminate the bound constraint in the trust region subproblem and their algorithm was applied to image deblurring in [Wang and Ma, 2009].

In contrast, there are fewer works on how to use subspace algorithms for handling nonlinear constraints. In the literature, attempts to use line-search or trust region methods (without a subspace strategy) for problems with general constraints of the form

$$C = \{\mathbf{x} \in \mathbb{R}^N \mid c_i(\mathbf{x}) \leq 0, \forall i \in \{1, \dots, r\}\}, \quad (3.1.2)$$

where, for every $i \in \{1, \dots, r\}$, $c_i : \mathbb{R}^N \rightarrow \mathbb{R}$ is continuously differentiable, usually involve the setup of sophisticated procedures. These procedures are essentially based on barrier procedure [Byrd et al., 2000], exterior penalty [Conn et al., 1991] strategies, Sequential Quadratic Programming (SQP) approaches [Gould and Toint, 2010], or mixed penalty-SQP strategies [Fletcher, 1985, Curtis et al., 2017]. To our knowledge, mostly SQP methods have been combined with a subspace strategy. SQP consists of solving a sequence of quadratic programming subproblems. Each of them aims at minimizing a quadratic model of the objective subject to a linearization of the constraints. A typical subspace SQP line-search

[Nocedal and Wright, 2006, Sec. 18.4] approach solves the subproblem:

$$\begin{aligned} & \underset{\mathbf{d} \in \mathbb{R}^N}{\text{minimize}} \quad m_k(\mathbf{d}) \\ & \text{subject to} \quad \nabla c_i(\mathbf{x}_k)^\top \mathbf{d} + c_i(\mathbf{x}_k) \leq 0, \quad i \in \{1, \dots, r\}, \\ & \quad \mathbf{d} \in \mathcal{S}_k, \end{aligned} \tag{3.1.3}$$

where m_k is a quadratic model of F at the current iterate \mathbf{x}_k . The next iterate is then computed as $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$, where \mathbf{d}_k is a solution to (3.1.3) and $\alpha_k \in \mathbb{R}$ is determined via classical rules. Lee et al. [Lee et al., 2019] analyzed this subspace SQP line-search algorithm for the particular choice of subspace:

$$\mathcal{S}_k = \text{Span}\{-\mathbf{g}_k, \mathbf{d}_1, \dots, \mathbf{d}_{k-1}, -\nabla c_i(\mathbf{x}_k)\}, \tag{3.1.4}$$

with $i = \text{argmax}\{|c_i(\mathbf{x}_k)|, i \in \{1, \dots, r\}\}$. On the other hand, in a subspace SQP trust region approach [Nocedal and Wright, 2006, Sec. 18.5], the solution \mathbf{s}_k to the following subproblem is computed:

$$\begin{aligned} & \underset{\mathbf{s} \in \mathbb{R}^N}{\text{minimize}} \quad m_k(\mathbf{s}) \\ & \text{subject to} \quad \nabla c_i(\mathbf{x}_k)^\top \mathbf{s} + c_i(\mathbf{x}_k) \leq 0, \quad i \in \{1, \dots, r\}, \\ & \quad \|\mathbf{s}\| \leq \delta_k, \\ & \quad \mathbf{s} \in \mathcal{S}_k, \end{aligned} \tag{3.1.5}$$

where the radius $\delta_k > 0$ of the trust region follows predefined rules. The next iterate corresponds to $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$. However the trust region strategy (3.1.5) is more complex to implement than the line-search one (3.1.3) because the approximate feasible region and the trust region may be disjoint. Celis et al. [Celis et al., 1984] proposed to relax problem (3.1.5), in the case $\mathcal{S}_k = \mathbb{R}^N$, into the so-called CDT problem, by replacing the linear inequalities with

$$\|(\nabla c_i(\mathbf{x}_k)^\top \mathbf{s} + c_i(\mathbf{x}_k))_-\|^2 \leq \zeta_{k,i}, \quad i \in \{1, \dots, r\}, \tag{3.1.6}$$

where $(\zeta_{k,i})_{1 \leq i \leq r} \in (\mathbb{R}^+)^r$. Grapiglia et al. [Grapiglia et al., 2013] studied a subspace version of CDT, but one drawback of their method is that the subspace dimension increases with iterations. In general, although SQP methods converge locally at a superlinear rate, they remain complex to implement as the subproblems may be infeasible. Well-known undesirable practical convergence behaviors [Powell, 1983] must also be circumvented. In addition, even in the special case when the constraint set C is closed convex, these approaches do not take advantage of the projection onto C when it is available. To our knowledge, the only subspace methods to do so are projected NLCG [Sun et al., 2006, Zhang et al., 2014] in the same spirit as projected gradient algorithms [Bonettini et al., 2008]. However, such algorithms remain limited to this category of constraints and cannot cope with simple precomposition with linear operators.

The method we investigate in this chapter is an adaptation of the Majorization-Minimization Subspace algorithm (MMS) proposed in [Chouzenoux et al., 2013a] to constrained problems. In a nutshell, our method is a subspace method following the scheme (2.3.6), coupled with techniques reminiscent of trust region techniques, embedded into an exterior penalty framework. Our approach offers several advantages

over the previously mentioned ones: (i) it can take into account a wide variety of constraints, and in particular, constraints for which the projection has closed-form are easily handled; (ii) the choice of the subspace is very flexible; (iii) the step search at each iteration is easy to implement, in particular it does not require solving a constrained problem thus avoiding feasibility issues. Our contribution is twofold. We first propose a modification of the MMS algorithm for solving efficiently penalized problems. The new MMS algorithm uses a local majorization strategy similar to trust region approaches, to overcome the ill-conditioning that usually arises in penalty methods. The convergence of the algorithm is analyzed. Secondly, the penalty method in which the new MMS algorithm is nested is allowed to be inexact, each subproblem being solved with a fixed tolerance. The convergence of the global method towards a solution to the constrained problem (3.1.1) is proved.

The rest of the chapter is organized as follows: Section 3.2 presents the optimization problem tackled through the chapter. In Section 3.3, we describe the original MMS algorithm, and we propose a new accelerated variant of it in Section 3.4. Section 3.5 is dedicated to the inexact exterior penalty method. Finally, in Section 3.6, comparisons between our approach and state-of-the-art algorithms are conducted on two image recovery applications.

3.2 Considered optimization problem

3.2.1 Notation and definitions

In this chapter, \mathbb{R}^N denotes the N -dimensional Euclidean space endowed with the standard scalar product $\langle \cdot, \cdot \rangle$ and the norm $\|\cdot\|$. We denote by \mathcal{S}^N the set of symmetric matrices of $\mathbb{R}^{N \times N}$ and \mathcal{S}_+^N the set of symmetric semi-definite positive matrices of $\mathbb{R}^{N \times N}$. The Loewner order relation is denoted by \preceq , i.e., for every $\mathbf{A} \in \mathcal{S}^N$ and $\mathbf{B} \in \mathcal{S}^N$, $\mathbf{A} \preceq \mathbf{B}$ if, for every $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{x}^\top \mathbf{A} \mathbf{x} \leq \mathbf{x}^\top \mathbf{B} \mathbf{x}$. We refer to the identity matrix of $\mathbb{R}^{N \times N}$ as \mathbf{I}_N . For every nonempty set $C \subseteq \mathbb{R}^N$, d_C is the distance function to C with respect to the Euclidean norm, and ι_C is the indicator function of C , namely $\iota_C(\mathbf{x}) = 0$ if $\mathbf{x} \in C$, $\iota_C(\mathbf{x}) = +\infty$ otherwise.

The following definition will be at the core of the developments in this chapter:

Definition 3.1. (Exterior penalty function) Let C be a nonempty subset of \mathbb{R}^N . We say $R: \mathbb{R}^N \rightarrow [0, +\infty)$ is an *exterior penalty function for constraint $\mathbf{x} \in C$* if

$$\operatorname{Argmin} R = C \quad \text{and} \quad \min R = 0. \quad (3.2.1)$$

As their name suggests it, such penalty functions assign a nonnegative cost to every point which is exterior to C .

Example 3.2. Let $\psi: [0, +\infty) \rightarrow [0, +\infty)$ be a strictly increasing function such that $\psi(0) = 0$. Let the constraint set C be the lower zero-level set of a function $f: \mathbb{R}^N \rightarrow \mathbb{R}$. Then functions of the form

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad R(\mathbf{x}) = \psi(\max(0, f(\mathbf{x}))), \quad (3.2.2)$$

are exterior penalty functions for the constraint $f(\mathbf{x}) \leq 0$ [Bertsekas, 2014, Luenberger, 1997]. In addition,

- if f and ψ are convex, then R is convex;
- if f and ψ are differentiable and $\psi'(0) = 0$, then R is differentiable.

In particular, when $\psi = (\cdot)^2$, we recover the widely used quadratic penalty [Fiacco and McCormick, 1990, Castillo, 2019].

Remark 3.3. One could think of using exact penalization methods. However, these techniques require more restrictive assumption since either they use non-differentiable penalties, or the objective and the constraint functions of the problem are assumed to be twice differentiable [Di Pillo and Grippo, 1989, Bertsekas, 2014].

A useful property of exterior penalty functions is the following one:

Proposition 3.4. Let $r \in \mathbb{N}^*$ and let $(C_i)_{1 \leq i \leq r}$ be subsets of \mathbb{R}^N . For every $i \in \{1, \dots, r\}$, let $R_i: \mathbb{R}^N \rightarrow \mathbb{R}$ be an exterior penalty function for constraint $\mathbf{x} \in C_i$. Then $\sum_{i=1}^r R_i$ is an exterior penalty function for the constraint $\mathbf{x} \in \bigcap_{i=1}^r C_i$.

3.2.2 Optimization problem

Throughout this chapter, we will use functions F and R satisfying the following hypotheses.

Assumption 3.1.

- $F: \mathbb{R}^N \rightarrow \mathbb{R}$ is a differentiable function.
- C is a nonempty subset of \mathbb{R}^N , which is decomposed as $C = \bigcap_{i=1}^r C_i$, where $r \in \mathbb{N}^*$ and $(C_i)_{1 \leq i \leq r}$ are subsets of \mathbb{R}^N .
- $R = \sum_{i=1}^r R_i$ where, for every $i \in \{1, \dots, r\}$, $R_i: \mathbb{R}^N \rightarrow \mathbb{R}$ is a differentiable exterior penalty function for constraint $\mathbf{x} \in C_i$.

Our main goal will be to solve the following constrained optimization problem.

Problem 3.5. Suppose that Assumption 3.1 is satisfied. We want to find

$$\hat{\mathbf{x}} \in \operatorname{Argmin}_{\mathbf{x} \in C} F(\mathbf{x}). \quad (3.2.3)$$

Following the framework of exterior penalty methods, we will tackle this problem by considering the penalized problem hereunder defined for every $\gamma \in (0, +\infty)$.

Problem 3.6. Suppose that Assumption 3.1 is satisfied. We want to find

$$\hat{\mathbf{x}}_\gamma \in \operatorname{Argmin}_{\mathbf{x} \in \mathbb{R}^N} F_\gamma(\mathbf{x}), \quad (3.2.4)$$

where

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad F_\gamma(\mathbf{x}) = F(\mathbf{x}) + \gamma R(\mathbf{x}). \quad (3.2.5)$$

3.3 Majorization-Minimization subspace algorithm

Let the penalty parameter $\gamma > 0$ be fixed in the whole section, and let F and $(R_i)_{1 \leq i \leq N}$ be functions fulfilling Assumption 3.1. We seek to solve the penalized Problem 3.6. Among available methods, the Majorization-Minimization (MM) principle offers a generic framework to solve this problem and can be efficiently combined with a subspace acceleration [Wald and Schuster, 2017, Elad et al., 2007, Yuan, 2008]. Hereafter, we describe the use of the Majorization-Minimization subspace algorithm, initially proposed in [Chouzenoux et al., 2010], for this particular problem.

3.3.1 Majorization-Minimization principle

Definition 3.7. Let $f : \mathbb{R}^N \rightarrow \mathbb{R}$ be a differentiable function and $\mathbf{x}' \in \mathbb{R}^N$. A function $\mathbf{x} \mapsto Q(\mathbf{x}, \mathbf{x}')$ is said to be a *global tangent majorant* of f at \mathbf{x}' if, for every $\mathbf{x} \in \mathbb{R}^N$,

$$f(\mathbf{x}) \leq Q(\mathbf{x}, \mathbf{x}') \quad \text{and} \quad f(\mathbf{x}') = Q(\mathbf{x}', \mathbf{x}'). \quad (3.3.1)$$

Moreover $\mathbf{x} \mapsto Q(\mathbf{x}, \mathbf{x}')$ is said to be a global tangent *quadratic* majorant of f at \mathbf{x}' if

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad Q(\mathbf{x}, \mathbf{x}') = f(\mathbf{x}') + \nabla f(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2} (\mathbf{x} - \mathbf{x}')^\top A(\mathbf{x}') (\mathbf{x} - \mathbf{x}'), \quad (3.3.2)$$

with $A(\mathbf{x}') \in \mathcal{S}_+^N$. In that case, $A(\mathbf{x}')$ is called the *curvature matrix* of function Q at point \mathbf{x}' .

In the following, we will assume that the functions F and $(R_i)_{1 \leq i \leq r}$, involved in Problem (\mathcal{P}_γ) , have quadratic majorants:

Assumption 3.2.

- (i) For every $\mathbf{x} \in \mathbb{R}^N$, F has a quadratic tangent majorant at \mathbf{x} with curvature matrix $A_F(\mathbf{x})$.
- (ii) For every $i \in \{1, \dots, r\}$ and for every $\mathbf{x} \in \mathbb{R}^N$, R_i has a quadratic tangent majorant at \mathbf{x} with curvature matrix $A_{R_i}(\mathbf{x})$.

As a consequence, for every $\mathbf{x} \in \mathbb{R}^N$, the penalty function R has a quadratic tangent majorant at \mathbf{x} with curvature

$$A_R(\mathbf{x}) = \sum_{i=1}^r A_{R_i}(\mathbf{x}), \quad (3.3.3)$$

and F_γ defined in (3.2.5) has a quadratic tangent majorant at \mathbf{x} with curvature

$$\mathbf{A}_{F_\gamma}(\mathbf{x}) = \mathbf{A}_F(\mathbf{x}) + \gamma \mathbf{A}_R(\mathbf{x}). \quad (3.3.4)$$

Example 3.8. Let us consider a wide class of penalty functions that we will subsequently use in our numerical experiments (see Section 3.6). Let $\mathbf{L} \in \mathbb{R}^{P \times N}$ and let C' be a nonempty closed convex set of \mathbb{R}^P . Suppose that the constraint set is

$$C = \mathbf{L}^{-1}(C') = \{\mathbf{x} \in \mathbb{R}^N \mid \mathbf{L}\mathbf{x} \in C'\}. \quad (3.3.5)$$

Let us define the exterior penalty function relative to the constraint set (3.3.5) as

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad R(\mathbf{x}) = d_{C'}^2(\mathbf{Lx}). \quad (3.3.6)$$

Since $\psi = d_{C'}^2$ has a 2-Lipschitz gradient, the quadratic function with curvature defined by, for every $\mathbf{y}' \in \mathbb{R}^P$, $\mathbf{A}(\mathbf{y}') = 2\mathbf{I}_P$ is a majorant of ψ . It then follows from Lemma 2.26 that a quadratic majorant of function R can be built with (constant) curvature

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{A}_R(\mathbf{x}) = 2\mathbf{L}^\top \mathbf{L}. \quad (3.3.7)$$

MM approaches are iterative methods for the minimization of an objective function $f: \mathbb{R}^N \rightarrow \mathbb{R}$ based on a surrogate such as (3.3.2). At each iteration of the algorithm, a global tangent majorant of the objective at the current iterate is minimized, yielding the following update rule:

$$\begin{cases} \mathbf{x}_0 \in \mathbb{R}^N, \\ (\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} \in \underset{\mathbf{x} \in \mathbb{R}^N}{\operatorname{Argmin}} Q(\mathbf{x}, \mathbf{x}_k). \end{cases} \quad (3.3.8)$$

When applied to the objective F_γ , the iterative procedure (3.3.8) corresponds to the half-quadratic algorithm. The convergence guarantees for this algorithm have been established for instance in [Nikolova and Ng, 2005, Allain et al., 2006] under mild boundedness assumptions on the curvature matrices sequence $(\mathbf{A}_{F_\gamma}(\mathbf{x}_k))_k$. However, the computational complexity for (3.3.8) is rather high, as it involves the inversion of an $N \times N$ matrix at each iteration. We thus propose to resort to a subspace accelerated version of it that we describe next.

3.3.2 Majorization-Minimization subspace algorithm

The MM subspace algorithm (MMS) takes advantage of both the MM principle, introduced in Sec. 3.3.1 and subspace acceleration strategies. In short, the MMS method that we describe in this section is a subspace minimization algorithm coupled with a quadratic MM step-size search [Chouzenoux et al., 2010, Chouzenoux et al., 2013a]. This allows us to avoid tedious $N \times N$ matrix inversions involved in the update of (3.3.8), while still preserving fast practical convergence [Chouzenoux and Pesquet, 2016].

3.3.2.1 Subspace minimization

A general subspace method moves the current point \mathbf{x}_k at each iteration $k \in \mathbb{N}$ along a subspace of dimension $M_k \in \{1, \dots, N\}$, by generating

$$(\forall k \in \mathbb{N}) \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_k, \quad (3.3.9)$$

where

$$(\forall k \in \mathbb{N}) \quad \mathbf{D}_k = [\mathbf{d}_k^1, \dots, \mathbf{d}_k^{M_k}] \quad (3.3.10)$$

and, for every $j \in \{1, \dots, M_k\}$, $\mathbf{d}_k^j \in \mathbb{R}^N$. \mathbf{D}_k is the search direction matrix and $\mathbf{u}_k \in \mathbb{R}^{M_k}$ is a multivariate step-size. In the present study, we will adopt the same requirement on \mathbf{D}_k as in [Chouzenoux et al., 2010], that is \mathbf{d}_k^1 is assumed to be *gradient-related* (see more details in Sec. 3.4.2, Assumption 3.5).

The following example describes relevant choices for \mathbf{D}_k encompassed by our analysis.

Example 3.9. Let $k \in \mathbb{N}$ and let us introduce the shorter notation $\mathbf{g}_k = \nabla F_\gamma(\mathbf{x}_k)$.

- (i) If $\mathbf{d}_k^1 = -\mathbf{g}_k$, then we retrieve the classical gradient descent direction.
- (ii) We can also use a preconditioned gradient direction, $\mathbf{d}_k^1 = -\mathbf{H}_k \mathbf{g}_k$, where \mathbf{H}_k is a symmetric positive definite matrix of $\mathbb{R}^{N \times N}$. For instance, if the Hessian of the objective function is invertible at the current iterate, \mathbf{H}_k can be chosen equal to the inverse of this Hessian (or an approximation of it). We then obtain the classical Newton (or quasi-Newton) direction.
- (iii) One can also use for \mathbf{d}_k^1 , the truncated Newton direction obtained by solving approximately the linear system $\nabla^2 F_\gamma(\mathbf{x}_k) \mathbf{d} + \mathbf{g}_k = 0$ with the Conjugate Gradient algorithm. Such choice was adopted, for instance, in the SESOP-TN method from [Zibulevsky, 2008].
- (iv) When the subspace is spanned by the gradient direction and the difference between two past iterates (also called *memory term*), i.e.

$$\mathbf{D}_k = \begin{cases} -\mathbf{g}_k & k = 0 \\ [-\mathbf{g}_k, \mathbf{x}_k - \mathbf{x}_{k-1}] & \text{elsewhere,} \end{cases} \quad (3.3.11)$$

we recover the *memory gradient* subspace introduced in [Miele and Cantrell, 1969], which is closely related to the search direction employed in the NLGC algorithm [Fletcher and Reeves, 1964, Zibulevsky, 2008]. This is also the subspace used in [Chouzenoux et al., 2010].

- (v) A generalization of the latter consists in adding further past directions by setting for some $m \in \mathbb{N}^*$, and $k \geq m$:

$$\mathbf{D}_k = [-\mathbf{g}_k, \mathbf{x}_k - \mathbf{x}_{k-1}, \dots, \mathbf{x}_{k-m+1} - \mathbf{x}_{k-m}], \quad (3.3.12)$$

yielding the *supermemory gradient subspace* from [Cragg and Levy, 1969].

- (vi) Matrix

$$\mathbf{D}_k = [-\mathbf{g}_k, \mathbf{x}_k - \mathbf{x}_{k-1}, \dots, \mathbf{x}_{k-m+1} - \mathbf{x}_{k-m}, \mathbf{g}_k - \mathbf{g}_{k-1}, \dots, \mathbf{g}_{k-m+1} - \mathbf{g}_{k-m}], \quad (3.3.13)$$

with $m \in \mathbb{N}^*$, and $k \geq m$, spans the subspace inherent to the limited memory quasi-Newton L-BFGS algorithm [Liu and Nocedal, 1989], when setting the memory size to m . L-BFGS method amounts to approximate the Newton step by a step in the above subspace, composed of $(2m + 1)$ directions including the m last differences between the passed iterates and the m last differences between the past gradients.

Once the method for building the subspace \mathbf{D}_k has been selected, one must define a suitable step-size strategy. Two approaches are typically employed. In the first approach, used in NLCG and L-BFGS for instance, a vector $\mathbf{s}_k \in \mathbb{R}^N$ belonging to the subspace spanned by the columns of \mathbf{D}_k is selected in a deterministic way (e.g., conjugation rules in NLCG). Then, the next iterate is $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{s}_k$ where $\alpha_k > 0$ is obtained through a standard one-dimensional linesearch procedure using rules such as Armijo, Goldstein, or Wolfe ones [Nocedal and Wright, 2006]. This approach is appealing because of its low complexity cost, but it requires an ad-hoc procedure for determining \mathbf{s}_k . Moreover, a one-dimensional step-size search might not allow to exploit optimally the whole subspace generated by \mathbf{D}_k , which can limit the performance of the method. In contrast, a multidimensional step-size search consists of finding

a step $\mathbf{u}_k \in \mathbb{R}^{M_k}$ minimizing f_k defined by

$$(\forall \mathbf{u} \in \mathbb{R}^M) \quad f_k(\mathbf{u}) = F_\gamma(\mathbf{x}_k + \mathbf{D}_k \mathbf{u}). \quad (3.3.14)$$

Solving the aforementioned minimization problem can be time-consuming, so a compromise must be reached between accuracy, convergence stability, and complexity. Typical strategies in the literature of subspace optimization involve a Newton or truncated Newton method [Narkiss and Zibulevsky, 2005, Zibulevsky, 2008], however they go with limited convergence guarantees.

As shown in [Chouzenoux et al., 2010], the MM framework allows to design a practical, fast step search, while providing better theoretical properties for convergence than Newton-based methods. The use of MM procedure for the step-size search in (3.3.9) yields MMS method, that we describe hereafter.

3.3.2.2 MMS algorithm

MMS defines the stepsize \mathbf{u}_k in (3.3.9) by minimizing a quadratic tangent majorant for f_k , defined hereabove, at $\mathbf{u}' = 0$. According to Assumption 3.2, such a majorant reads for every $k \in \mathbb{N}$,

$$(\forall \mathbf{u} \in \mathbb{R}^{M_k}) \quad q_k(\mathbf{u}) = F_\gamma(\mathbf{x}_k) + \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{D}_k \mathbf{u} + \frac{1}{2} \mathbf{u}^\top \mathbf{B}_k \mathbf{u}, \quad (3.3.15)$$

where $\mathbf{B}_k \in \mathbb{R}^{M_k \times M_k}$ is defined as

$$\mathbf{B}_k = \mathbf{D}_k^\top \mathbf{A}_{F_\gamma}(\mathbf{x}_k) \mathbf{D}_k, \quad (3.3.16)$$

and $\mathbf{A}_{F_\gamma}(\mathbf{x}_k)$ is the majorant curvature matrix introduced in (3.3.4). The MM step search strategy then reduces to minimizing the quadratic function q_k over \mathbb{R}^{M_k} , therefore leading to a low-complexity closed form computation for the step-size \mathbf{u}_k at each iteration. The MMS method for minimizing F_γ is detailed in Algorithm 7, where \dagger stands for the pseudo-inverse operation. The convergence properties of MMS are studied for instance in [Chouzenoux and Pesquet, 2016, Chouzenoux et al., 2010, Chouzenoux et al., 2013a]. Its good practical performance, in comparison with NLCG and L-BFGS in particular, have been illustrated on various image restoration problems in [Chouzenoux et al., 2010, Chouzenoux et al., 2013a].

Algorithm 7: MMS($\mathbf{x}_0, \gamma, \varepsilon$)

```

Inputs:  $(\gamma, \varepsilon) \in (0, +\infty)^2$ ,  $\mathbf{x}_0 \in \mathbb{R}^N$ .
for  $k = 0, 1, \dots$  do
    Construct  $\mathbf{D}_k$  according to (3.3.10),
    Compute  $\mathbf{B}_k$  according to (3.3.16),
     $\mathbf{u}_k = -[\mathbf{B}_k]^\dagger \mathbf{D}_k^\top \nabla F_\gamma(\mathbf{x}_k)$ ,
     $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_k$ ,
    if  $\|\nabla F_\gamma(\mathbf{x}_{k+1})\| \leq \varepsilon$  then
        exit loop // stop algorithm if given precision  $\varepsilon$  on the norm of the
        gradient is reached
        return  $\mathbf{x}_{k+1}$ .
    end
end

```

3.4 Proposed local variant of MMS

The penalty function present in Problem 3.6 may have a large curvature, which can jeopardize the good accuracy of the majorant function inherent to the MM strategy and thus significantly slowdown the convergence of the algorithm [Sghaier et al., 2022, Cherni et al., 2020]. This issue had been overcome by the authors of [Sghaier et al., 2022], in the particular case of the MM memory gradient (3MG) method, by empirically substituting local majorizations for global majorizations, in a similar spirit as trust-region approaches [Yuan, 2015, Toint, 1988, Branch et al., 1999]. In what follows, we formalize and generalize such a local approach to accelerate any MM subspace algorithm, and we demonstrate the convergence of the resulting iterative approach to a critical point of Problem 3.6.

3.4.1 Description of the algorithm

According to theoretical results concerning penalty methods, large values of the penalty parameter γ should be chosen for the solutions to Problem 3.6 to lie close to the constraint set. However, as stressed out in [Conn et al., 2000, Dussault, 1995, Konnov, 2019], large values of γ tend to overemphasize the role of the penalty function R with respect to the objective function F . Consequently, the penalized problem becomes ill-conditioned, hence difficult to solve. This harshly impacts the convergence profile of MMS as observed in [Sghaier et al., 2022]. This slowdown is directly related to a large spectral norm of the curvature matrix \mathbf{B}_k , defined by (3.3.16) at each iteration k , which leads to small update steps.

We thus propose a novel approach to cope with the aforementioned issue by making use of a local majorant for F_γ — by opposition to a global one. We define a local majorant as follows.

Definition 3.10. Let $f : \mathbb{R}^N \rightarrow \mathbb{R}$ be a differentiable function, let $\mathcal{D} \subseteq \mathbb{R}^N$, and let $\mathbf{x}' \in \mathcal{D}$. A function $\mathbf{x} \mapsto Q(\mathbf{x}, \mathbf{x}')$ is said to be a *local tangent majorant* of f at \mathbf{x}' on the *trust region* \mathcal{D} if, for every $\mathbf{x} \in \mathcal{D}$, the majorizing property (3.3.1) is satisfied.

The local MM framework shares similarities with trust region methods [Conn et al., 2000], the objective model identifying here with the quadratic majorant and the trust region with the domain over which the majorant is valid. This kind of local majorization strategy has already been successfully used for the proximal-based MM algorithm in [Cherni et al., 2020], leading to a significant speed up. In the case of penalized problems of the form of Problem 3.6, the use of local majorants is promising. Indeed, when γ is large enough, constraints are likely to be satisfied after some iterations. Then, the penalty term has no more effect locally and therefore the majorant of F_γ can be reduced to the sole contribution of F .

At each iteration k and given \mathbf{x}_k , we thus propose to make a finite number of curvature trials for our local majorants of F_γ . For each trial, an MMS update is performed using the corresponding curvature. If the resulting iterate lies in the local domain where the majorizing property (3.3.1) is verified, it is accepted and defined as \mathbf{x}_{k+1} . Otherwise the curvature of the local majorant is strictly increased (in the sense of Loewner order). Designing this test allows us to guarantee the same descent properties than in the global case as will be shown in next subsection. To construct the sequence of increasing curvatures, we rely on the following definitions.

Definition 3.11. Let $\mathcal{I} \subseteq \{1, \dots, r\}$. For every $\mathbf{x} \in \mathbb{R}^N$, we define the curvature $\mathbf{A}_{F_\gamma}^{(\mathcal{I})}(\mathbf{x}) \in \mathcal{S}_+^N$ as

$$\mathbf{A}_{F_\gamma}^{(\mathcal{I})}(\mathbf{x}) = A_F(\mathbf{x}) + \gamma \sum_{i \in \mathcal{I}} \mathbf{A}_{R_i}(\mathbf{x}), \quad (3.4.1)$$

and the associated local domain as

$$\mathcal{D}^{(\mathcal{I})} = \{\mathbf{x} \in \mathbb{R}^N \mid (\forall i \notin \mathcal{I}) \ \mathbf{x} \in C_i\}. \quad (3.4.2)$$

Note that, for every $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{A}_{F_\gamma}^{(\mathcal{I})}(\mathbf{x}) \preceq \mathbf{A}_{F_\gamma}(\mathbf{x})$, where $\mathbf{A}_{F_\gamma}(\mathbf{x})$ is defined in (3.3.4). In particular, when $\mathcal{I} = \{1, \dots, r\}$, we have, for every $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{A}_{F_\gamma}^{(\mathcal{I})}(\mathbf{x}) = \mathbf{A}_{F_\gamma}(\mathbf{x})$ and $\mathcal{D}^{(\mathcal{I})} = \mathbb{R}^N$. In other words, we recover a global majorant.

Let us now explain how to build local majorants of F_γ at \mathbf{x}_k , for a given $k \in \mathbb{N}$. Let

$$\mathcal{I}_k = \{i \in \{1, \dots, r\} \mid \mathbf{x}_k \notin C_i\}. \quad (3.4.3)$$

A suitable strategy for majorizing F_γ at \mathbf{x}_k then consists of choosing a local majorant which only penalizes the remaining constraints $(C_i)_{i \in \mathcal{I}_k}$. Such a local majorant can be given by

$$Q_k^{(\mathcal{I}_k)} : \mathbf{x} \mapsto F_\gamma(\mathbf{x}_k) + \nabla F_\gamma(\mathbf{x}_k)^\top (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_k)^\top \mathbf{A}_{F_\gamma}^{(\mathcal{I}_k)}(\mathbf{x}_k) (\mathbf{x} - \mathbf{x}_k), \quad (3.4.4)$$

on the trust region $\mathcal{D}^{(\mathcal{I}_k)}$, that is

$$(\forall \mathbf{x} \in \mathcal{D}^{(\mathcal{I}_k)}) \quad F_\gamma(\mathbf{x}) \leq Q_k^{(\mathcal{I}_k)}(\mathbf{x}). \quad (3.4.5)$$

Consequently, $q_k^{(\mathcal{I}_k)}$ defined as

$$(\forall \mathbf{u} \in \mathbb{R}^M) \quad q_k^{(\mathcal{I}_k)}(\mathbf{u}) = F_\gamma(\mathbf{x}_k) + \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{D}_k \mathbf{u} + \frac{1}{2} \mathbf{u}^\top \mathbf{B}_k^{(\mathcal{I}_k)} \mathbf{u}, \quad (3.4.6)$$

with

$$\mathbf{B}_k^{(\mathcal{I}_k)} = \mathbf{D}_k^\top \mathbf{A}_{F_\gamma}^{(\mathcal{I}_k)}(\mathbf{x}_k) \mathbf{D}_k, \quad (3.4.7)$$

is a local majorant of f_k defined in (3.3.14), at $\mathbf{u}' = 0$, on the trust region

$$\left\{ \mathbf{u} \in \mathbb{R}^{M_k} \mid \mathbf{x}_k + \mathbf{D}_k \mathbf{u} \in \mathcal{D}^{(\mathcal{I}_k)} \right\}. \quad (3.4.8)$$

Remark 3.12. From the definition of $\mathbf{A}_{F_\gamma}^{(\mathcal{I}_k)}(\mathbf{x}_k)$ in (3.4.1), for every $\mathbf{u} \in \mathbb{R}^{M_k}$, $q_k^{(\mathcal{I}_k)}(\mathbf{u}) \leq q_k(\mathbf{u})$, which means that the local majorants are upper bounded by the global one.

Our resulting local MMS procedure is detailed in Algorithm 8. At each iteration $k \in \mathbb{N}$ and, at each MMS trial $\ell \in \mathbb{N}$ of Algorithm 8, a local majorant of f_k is constructed on a trust region which depends on the constraints satisfied at the previous trials $\mathbf{x}_{k,0}, \dots, \mathbf{x}_{k,\ell-1}$. The procedure stops when $\mathbf{x}_{k,\ell}$ violates no new constraints compared to the previous trials $(\mathbf{x}_{k,i})_{0 \leq i \leq \ell-1}$. In other words, $\mathbf{x}_{k,\ell}$ belongs to the trust region of the local majorant from which it was computed. The fact that the sequence $(\mathcal{I}_{k,\ell})_{\ell \in \mathbb{N}}$ is monotonically increasing ensures that the method is well defined: in the worst case we end up using the global majorant (i.e., $\mathcal{I}_{k,\ell} = \{1, \dots, r\}$) and thus the standard MMS iteration is retrieved.

Algorithm 8: MMS^{loc}($\mathbf{x}_0, \gamma, \varepsilon$)

Inputs: $(\gamma, \varepsilon) \in (0, +\infty)^2$, $\mathbf{x}_0 \in \mathbb{R}^N$.

```

for  $k = 0, 1, \dots$  do
    Construct  $\mathbf{D}_k$  according to (3.3.10),
     $\mathbf{x}_{k,0} = \mathbf{x}_k$ , // initialize the current point to  $\mathbf{x}_k$ 
     $\mathcal{I}_{k,0} = \{i \in \{1, \dots, r\} \mid \mathbf{x}_{k,0} \notin C_i\}$ , // initialize index set with the indices
        of constraints not satisfied by  $\mathbf{x}_{k,0}$ 
    for  $\ell = 1, \dots$  do
        Compute  $\mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})}$  according to (3.4.7),
         $\mathbf{u}_{k,\ell} = -[\mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})}]^\dagger \mathbf{D}_k^\top \nabla F_\gamma(\mathbf{x}_k)$ ,
         $\mathbf{x}_{k,\ell} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_{k,\ell}$ ,
         $\mathcal{I}_{k,\ell} = \mathcal{I}_{k,\ell-1} \cup \{i \in \{1, \dots, r\} \mid \mathbf{x}_{k,\ell} \notin C_i\}$ , // increment index set with the
            indices of constraints not satisfied by  $\mathbf{x}_{k,\ell}$ 
        if  $\mathcal{I}_{k,\ell} = \mathcal{I}_{k,\ell-1}$  then
            | exit loop, // accept the update  $\mathbf{x}_{k,\ell}$  if the local majorization
                property is valid
        end
    end
     $\mathbf{x}_{k+1} = \mathbf{x}_{k,\ell}$ , // define  $\mathbf{x}_{k+1}$  as  $\mathbf{x}_{k,\ell}$ 
    if  $\|\nabla F_\gamma(\mathbf{x}_{k+1})\| < \varepsilon$  then
        | exit loop // stop algorithm if given precision  $\varepsilon$  on the norm of the
            gradient is reached
        | return  $\mathbf{x}_{k+1}$ .
    end
end

```

3.4.2 Convergence analysis of the proposed algorithm

We analyse here the convergence of Algorithm 8. Our main convergence result corresponds to Theorem 3.4.2.2 and does not require any convexity assumption on F_γ .

3.4.2.1 Assumptions

In addition to Assumption 3.1-3.2, our convergence analysis of Algorithm 8 relies on similar assumptions as the ones commonly adopted for MMS [Chouzenoux et al., 2013a].

We first make the following assumption ensuring the boundedness of the sequence generated by Algorithm 8.

Assumption 3.3. F_γ is coercive on \mathbb{R}^N .

The next assumption controls the eigenvalues of the curvature function $\mathbf{x} \mapsto \mathbf{A}_{F_\gamma}(\mathbf{x})$.

Assumption 3.4. We have the following boundedness properties on the curvature functions $\mathbf{x} \mapsto \mathbf{A}_F(\mathbf{x})$ and $\mathbf{x} \mapsto \mathbf{A}_{F_\gamma}(\mathbf{x})$:

- (i) There exists $\eta > 0$ such that, for every $\mathbf{x} \in \mathbb{R}^N$,

$$(\forall \mathbf{v} \in \mathbb{R}^N) \quad \eta \|\mathbf{v}\|^2 \leq \mathbf{v}^\top \mathbf{A}_F(\mathbf{x}) \mathbf{v}. \quad (3.4.9)$$

- (ii) For every nonempty compact set $\mathcal{B} \subseteq \mathbb{R}^N$, there exists $\nu > 0$ such that for every $\mathbf{x} \in \mathcal{B}$,

$$(\forall \mathbf{v} \in \mathbb{R}^N) \quad \mathbf{v}^\top \mathbf{A}_{F_\gamma}(\mathbf{x}) \mathbf{v} \leq \nu \|\mathbf{v}\|^2. \quad (3.4.10)$$

Note that this last assumption is mild, since Assumption (i) can be satisfied by adding an elastic term to F , and Assumption (ii) is straightforwardly satisfied when $\mathbf{x} \mapsto \mathbf{A}_{F_\gamma}(\mathbf{x})$ is continuous, which is often the case. Let us also remark that Assumption (i) ensures that for every $\mathcal{I} \subseteq \{1, \dots, r\}$ and for every $\mathbf{x} \in \mathbb{R}^N$,

$$(\forall \mathbf{v} \in \mathbb{R}^N) \quad \eta \|\mathbf{v}\|^2 \leq \mathbf{v}^\top \mathbf{A}_{F_\gamma}^{(\mathcal{I})}(\mathbf{x}) \mathbf{v}. \quad (3.4.11)$$

In particular for $\mathcal{I} = \{1, \dots, r\}$ we obtain

$$(\forall \mathbf{v} \in \mathbb{R}^N) \quad \eta \|\mathbf{v}\|^2 \leq \mathbf{v}^\top \mathbf{A}_{F_\gamma}(\mathbf{x}) \mathbf{v}, \quad (3.4.12)$$

which is a standard assumption in the analysis of MM and MMS methods [Allain et al., 2006, Chouzenoux et al., 2013a, Chouzenoux and Pesquet, 2016].

Moreover, we assume that the set of directions $(\mathbf{D}_k)_{k \in \mathbb{N}}$ fulfills the following condition.

Assumption 3.5. There exist $(\mu_0, \mu_1) \in (0, +\infty)^2$ such that for every $k \in \mathbb{N}$, the first column \mathbf{d}_k^1 of direction matrix $\mathbf{D}_k \in \mathbb{R}^{N \times M_k}$ (with $1 \leq M_k \leq N$) satisfies:

$$\nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1 \leq -\mu_0 \|\nabla F_\gamma(\mathbf{x}_k)\|^2, \quad (3.4.13)$$

$$\|\mathbf{d}_k^1\| \leq \mu_1 \|\nabla F_\gamma(\mathbf{x}_k)\|. \quad (3.4.14)$$

Assumption 3.5 is satisfied by classical subspace directions, including all those listed in Example 3.9 (see also [Chouzenoux et al., 2010] for a discussion on this assumption).

We finally assume that the penalized function F_γ verifies the following Kurdyka-Łojasiewicz (KL) property.

Assumption 3.6. For all $\tilde{\mathbf{x}} \in \mathbb{R}^N$, there exist $\beta \in (0, +\infty)$, $\zeta \in (0, +\infty)$ and $\theta \in [0, 1)$ (all depending of γ) such that for every $\mathbf{x} \in \mathbb{R}^N$ verifying $|F_\gamma(\mathbf{x}) - F_\gamma(\tilde{\mathbf{x}})| \leq \zeta$,

$$\|\nabla F_\gamma(\mathbf{x})\| \geq \beta |F_\gamma(\mathbf{x}) - F_\gamma(\tilde{\mathbf{x}})|^\theta. \quad (3.4.15)$$

In recent years, this property has become a fundamental tool for convergence analysis in a nonconvex setting [Bolte et al., 2006, Bolte et al., 2014]. Functions satisfying the KL property include semi-algebraic functions and real analytic functions. In particular, the set of semi-algebraic functions is stable under various operations (sum, product, composition...) and contains polynomials, square root, distance to a semi-algebraic set, etc. The KL assumption on F_γ is not restrictive for standard choices of F and R . As we will see in the applications in Section 3.6, this assumption is met for a wide range of functions used in the context of image recovery. For instance, the standard Gaussian log-likelihood $\mathbf{x} \mapsto \|\mathbf{H}\mathbf{x} - \mathbf{y}\|^2$ with $\mathbf{H} \in \mathbb{R}^{N \times M}$ and $\mathbf{y} \in \mathbb{R}^M$, and classical regularizations such as ℓ_1 or total variation are semi-algebraic. Concerning the penalty function R , if we suppose it to be of the form given in Example 3.2, it is semi-algebraic as soon as ψ and f are. Therefore, constraints that can be taken into account include polynomial constraints and distance to balls or ellipsoids.

3.4.2.2 Convergence proof

Our proof follows the same structure as the convergence proof of MMS [Chouzenoux et al., 2013a]. We adapt the descent lemmas to the local case and conclude using [Chouzenoux et al., 2013a, Theorem 3].

Lemma 3.13. *Suppose that Assumptions 3.1, 3.2, and 3.4 are satisfied. Let $\mathbf{x}_0 \in \mathbb{R}^N$ be given. Then, the sequence $(\mathbf{x}_k)_{k \in \mathbb{N}}$ produced by Algorithm 8 with $\varepsilon = 0$ satisfies for every $k \in \mathbb{N}$,*

$$F_\gamma(\mathbf{x}_k) - F_\gamma(\mathbf{x}_{k+1}) \geq \frac{\eta}{2} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2, \quad (3.4.16)$$

where η is defined in (3.4.9).

Proof. Since Algorithm 8 is well-defined, there exists $\ell \in \mathbb{N}^*$ such that $\mathbf{x}_{k+1} = \mathbf{x}_{k,\ell} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_{k,\ell}$, $\mathbf{u}_{k,\ell} \in \mathbb{R}^{M_k}$. By definition of $\mathbf{u}_{k,\ell}$ we have

$$\mathbf{D}_k^\top \nabla F_\gamma(\mathbf{x}_k) = -\mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})} \mathbf{u}_{k,\ell}, \quad (3.4.17)$$

for some index set $\mathcal{I}_{k,\ell-1} \subseteq \{1, \dots, r\}$ built from the previous trials $\mathbf{x}_k = \mathbf{x}_{k,0}, \dots, \mathbf{x}_{k,\ell-1}$. Using the

expression (3.4.6) of $q_k^{(\mathcal{I}_{k,\ell-1})}$,

$$\begin{aligned}
 F_\gamma(\mathbf{x}_k) - q_k^{(\mathcal{I}_{k,\ell-1})}(\mathbf{u}_{k,\ell}) &= -\nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{D}_k^\top \mathbf{u}_{k,\ell} - \frac{1}{2}(\mathbf{u}_{k,\ell})^\top \mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})} \mathbf{u}_{k,\ell} \\
 &= (\mathbf{u}_{k,\ell})^\top \mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})} \mathbf{u}_{k,\ell} - \frac{1}{2}(\mathbf{u}_{k,\ell})^\top \mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})} \mathbf{u}_{k,\ell} \\
 &= \frac{1}{2}(\mathbf{u}_{k,\ell})^\top \mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})} \mathbf{u}_{k,\ell}.
 \end{aligned} \tag{3.4.18}$$

We now use the fact that $q_k^{(\mathcal{I}_{k,\ell-1})}$ is a local tangent majorant of f_k at $\mathbf{u}' = 0$ on the trust region

$$\left\{ \mathbf{u} \in \mathbb{R}^M \mid \mathbf{x}_k + \mathbf{D}_k \mathbf{u} \in \mathcal{D}^{(\mathcal{I}_{k,\ell-1})} \right\}.$$

By definition of ℓ , $\mathbf{u}_{k,\ell}$ belongs to the above set, hence

$$\begin{aligned}
 q_k^{(\mathcal{I}_{k,\ell-1})}(\mathbf{u}_{k,\ell}) &\geq f_k(\mathbf{u}_{k,\ell}) \\
 &= F_\gamma(\mathbf{x}_k + \mathbf{D}_k \mathbf{u}_{k,\ell}) \\
 &= F_\gamma(\mathbf{x}_{k+1}).
 \end{aligned} \tag{3.4.19}$$

Finally, from Assumption (i), (3.4.18), and (3.4.19),

$$\begin{aligned}
 F_\gamma(\mathbf{x}_k) - F_\gamma(\mathbf{x}_{k+1}) &\geq \frac{1}{2}(\mathbf{u}_{k,\ell})^\top \mathbf{B}_k^{(\mathcal{I}_{k,\ell-1})} \mathbf{u}_{k,\ell} \\
 &= \frac{1}{2}(\mathbf{D}_k \mathbf{u}_{k,\ell})^\top \mathbf{A}_{F_\gamma}^{(\mathcal{I}_{k,\ell-1})}(\mathbf{x}_k) \mathbf{D}_k \mathbf{u}_{k,\ell} \\
 &\geq \frac{\eta}{2} \|\mathbf{D}_k \mathbf{u}_{k,\ell}\|^2 \\
 &= \frac{\eta}{2} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2.
 \end{aligned} \tag{3.4.20}$$

□

Corollary 3.14. Suppose that Assumptions 3.1-3.4 are satisfied. Let $\mathbf{x}_0 \in \mathbb{R}^N$ be given. Then the sequence $(\mathbf{x}_k)_{k \in \mathbb{N}}$ generated by Algorithm 8 with $\varepsilon = 0$ is bounded.

Proof. From Lemma 3.13, $(F_\gamma(\mathbf{x}_k))_{k \in \mathbb{N}}$ is nonincreasing, thus for every $k \in \mathbb{N}$, $F_\gamma(\mathbf{x}_k) \leq F_\gamma(\mathbf{x}_0)$. We deduce that $(F_\gamma(\mathbf{x}_k))_{k \in \mathbb{N}}$ is bounded from above and the boundedness of $(\mathbf{x}_k)_{k \in \mathbb{N}}$ follows from Assumption 3.3. □

Lemma 3.15. Suppose that Assumptions 3.1, 3.2, 3.4 and 3.5 hold. Let $\mathbf{x}_0 \in \mathbb{R}^N$ be given. Then, there exists $\nu > 0$ such that the sequence $(\mathbf{x}_k)_{k \in \mathbb{N}}$ produced by Algorithm 8 with $\varepsilon = 0$ satisfies, for every $k \in \mathbb{N}$,

$$F_\gamma(\mathbf{x}_k) - F_\gamma(\mathbf{x}_{k+1}) \geq \frac{1}{2} \frac{\mu_0^2}{\nu \mu_1^2} \|\nabla F_\gamma(\mathbf{x}_k)\|^2, \tag{3.4.21}$$

where μ_0 and μ_1 are defined in Assumption 3.5.

Proof. Fix $k \in \mathbb{N}$. First, note that if the first direction in \mathbf{D}_k satisfies $\mathbf{d}_k^1 = 0$, then Assumption 3.5 implies that $\nabla F_\gamma(\mathbf{x}_k) = 0$. Thus $\mathbf{x}_{k+1} = \mathbf{x}_k$ and (3.4.21) is satisfied. We shall now assume that $\mathbf{d}_k^1 \neq 0$.

In the same way as in the previous proof, there exists $\ell \in \mathbb{N}^*$ such that $\mathbf{x}_{k+1} = \mathbf{x}_{k,\ell} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_{k,\ell}$. Let

$$(\forall t \in \mathbb{R}) \quad h_k(t) = q_k([t, 0, \dots, 0]), \quad (3.4.22)$$

where q_k is defined in (3.3.15). Expanding h_k yields

$$h_k(t) = F_\gamma(\mathbf{x}_k) + t \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1 + \frac{t^2}{2} (\mathbf{d}_k^1)^\top \mathbf{A}_{F_\gamma}(\mathbf{x}_k) \mathbf{d}_k^1. \quad (3.4.23)$$

Because of Assumption (i), h_k is a strictly convex quadratic function which achieves its minimum at

$$\hat{t}_k = \frac{-\nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1}{(\mathbf{d}_k^1)^\top \mathbf{A}_{F_\gamma}(\mathbf{x}_k) \mathbf{d}_k^1}, \quad (3.4.24)$$

with minimum value

$$h_k(\hat{t}_k) = F_\gamma(\mathbf{x}_k) + \frac{1}{2} \hat{t}_k \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1. \quad (3.4.25)$$

Recall that by definition, $\mathbf{u}_{k,\ell}$ minimizes $q_k^{(\mathcal{I}_{k,\ell-1})}$, defined in (3.4.6). It follows from Remark 3.12 that

$$\begin{aligned} q_k^{(\mathcal{I}_{k,\ell-1})}(\mathbf{u}_{k,\ell}) &= \min_{\mathbf{u} \in \mathbb{R}^{M_k}} q_k^{(\mathcal{I}_{k,\ell-1})}(\mathbf{u}) \\ &\leq \min_{\mathbf{u} \in \mathbb{R}^{M_k}} q_k(\mathbf{u}) \\ &\leq h_k(\hat{t}_k) \\ &= F_\gamma(\mathbf{x}_k) + \frac{1}{2} \hat{t}_k \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1, \end{aligned} \quad (3.4.26)$$

which can be rewritten as

$$F_\gamma(\mathbf{x}_k) - q_k^{(\mathcal{I}_{k,\ell-1})}(\mathbf{u}_{k,\ell}) \geq -\frac{1}{2} \hat{t}_k \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1. \quad (3.4.27)$$

In addition, since $q_k^{(\mathcal{I}_{k,\ell-1})}$ is a local tangent majorant of f_k at $\mathbf{u}' = 0$ on a trust region containing $\mathbf{u}_{k,\ell}$,

$$\begin{aligned} q_k^{(\mathcal{I}_{k,\ell-1})}(\mathbf{u}_{k,\ell}) &\geq f_k(\mathbf{u}_{k,\ell}) \\ &= F_\gamma(\mathbf{x}_k + \mathbf{D}_k \mathbf{u}_{k,\ell}) \\ &= F_\gamma(\mathbf{x}_{k+1}). \end{aligned} \quad (3.4.28)$$

Coupling (3.4.27) and (3.4.28), we obtain:

$$F_\gamma(\mathbf{x}_k) - F_\gamma(\mathbf{x}_{k+1}) \geq -\frac{1}{2} \hat{t}_k \nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1. \quad (3.4.29)$$

Let us now use Assumptions 3.4 and 3.5. From Lemma 3.13, $(\mathbf{x}_k)_{k \in \mathbb{N}}$ belongs to a compact set $\mathcal{B} \subseteq \mathbb{R}^N$.

Let ν be given by Assumption (ii) for this compact set. Firstly, according to (3.4.14),

$$\hat{t}_k \leq -\frac{\nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1}{\nu \|\mathbf{d}_k^1\|^2} \leq -\frac{\nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1}{\nu \mu_1^2 \|\nabla F_\gamma(\mathbf{x}_k)\|^2}. \quad (3.4.30)$$

Moreover, according to (3.4.13),

$$(\nabla F_\gamma(\mathbf{x}_k)^\top \mathbf{d}_k^1)^2 \geq \mu_0^2 \|\nabla F_\gamma(\mathbf{x}_k)\|^4. \quad (3.4.31)$$

Combining (3.4.29), (3.4.30), and (3.4.31), the desired inequality is obtained

$$F_\gamma(\mathbf{x}_k) - F_\gamma(\mathbf{x}_{k+1}) \geq \frac{1}{2} \frac{\mu_0^2}{\nu \mu_1^2} \|\nabla F_\gamma(\mathbf{x}_k)\|^2. \quad (3.4.32)$$

□

Finally the following theorem can be deduced as a consequence of the previous lemmas and [Chouzenoux et al., 2013a, Theorem 3], which was established using the KL Assumption 3.6.

Theorem 3.16

Suppose that Assumptions 3.1-3.6 are satisfied. Let $\mathbf{x}_0 \in \mathbb{R}^N$ be given. Then, for every $\varepsilon \in (0, +\infty)$ Algorithm 8 stops after a finite number of iterations K_ε . In addition, there exists a critical point $\hat{\mathbf{x}}$ of F_γ such that $\mathbf{x}_{K_\varepsilon} \rightarrow \hat{\mathbf{x}}$ as $\varepsilon \rightarrow 0$.

3.5 Embedding the proposed algorithm in a sequential penalty framework

In the previous section, we have presented a new MM-based subspace algorithm for the resolution of penalized problems of the form 3.6. Coming back to our main objective of solving the constrained problem 3.5, we propose to embed Algorithm 8 in an exterior penalty method by progressively increasing the penalty parameter. This section describes the proposed iterative approach and provides a convergence proof for it.

3.5.1 Inexact exterior penalty method

Exterior penalty methods solve Problem 3.5 by recasting it into a sequence of unconstrained subproblems of the form 3.6, for $\gamma > 0$ growing to infinity [Bertsekas, 2014, Bonnans et al., 2006]. The motivation is the following. Let $(\gamma_j)_{j \in \mathbb{N}}$ be a real sequence of positive reals such that $\lim_{j \rightarrow +\infty} \gamma_j = +\infty$. Then, for every $\mathbf{x} \in \mathbb{R}^N$,

$$\gamma_j R(\mathbf{x}) \xrightarrow{j \rightarrow +\infty} \iota_C(\mathbf{x}). \quad (3.5.1)$$

Thus, for some j sufficiently large, one may expect any solution to (\mathcal{P}_{γ_j}) to be close to a solution of 3.5. This intuition turns out to be valid under the assumption that F and R are lower semi-continuous, which is covered by Assumption 3.1 (see [Bertsekas, 2014, Bonnans et al., 2006, Luenberger, 1997]). More precisely, let $(\gamma_j)_{j \in \mathbb{N}}$ be set as stated before, and denote by \mathbf{x}_j a solution to (\mathcal{P}_{γ_j}) for $j \in \mathbb{N}$. Then, $(\mathbf{x}_j)_{j \in \mathbb{N}}$ is bounded and any of its cluster points is a solution to 3.5 [Luenberger, 1997, Chap. 10, Thm. 1]. However, for obvious practical reasons, the penalized subproblems $(\mathcal{P}_{\gamma_j})_{j \in \mathbb{N}}$ cannot be solved exactly. It is actually not even desirable to solve them with high accuracy before reaching large values of γ_j as this would be of limited interest while inducing a high computation cost.

Following [Gould, 1989, Fiacco and McCormick, 1990], we propose to solve the subproblems $(\mathcal{P}_{\gamma_j})_{j \in \mathbb{N}}$ with an increasing accuracy. More precisely, given $j \in \mathbb{N}$, an initial point $\mathbf{x}_j^{(0)} \in \mathbb{R}^N$, a penalty parameter

$\gamma_j \in (0, +\infty)$ and a precision $\varepsilon_j \in (0, +\infty)$, Algorithm 8 generates a vector $\mathbf{x}_j \in \mathbb{R}^N$ such that

$$\|\nabla F_{\gamma_j}(\mathbf{x}_j)\| < \varepsilon_j. \quad (3.5.2)$$

The loop over $j \in \mathbb{N}$ in Algorithm 9 leads to an inexact exterior penalty method, for which we provide a convergence analysis in the remaining of this section. We denote by P-MMS^{loc} (resp. P-MMS) the penalized MMS algorithm with (resp. without) local acceleration.

Algorithm 9: P-MMS^(loc)

Inputs: $(\gamma_j)_{j \in \mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$, $(\varepsilon_j)_{j \in \mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$.
for $j = 0, 1, \dots$ **do**
 | Set initial point $\mathbf{x}_j^{(0)}$,
 | $\mathbf{x}_j = \text{MMS}^{(\text{loc})}(\mathbf{x}_j^{(0)}, \gamma_j, \varepsilon_j)$, // find an inexact solution to (\mathcal{P}_{γ_j})
end
return \mathbf{x}_j .

3.5.2 Assumptions

We introduce the following classical assumption on the sequence of penalty and precision parameters involved in our method.

Assumption 3.7. The sequences of parameters $(\varepsilon_j)_{j \in \mathbb{N}}$ and $(\gamma_j)_{j \in \mathbb{N}}$ satisfy:

- (i) For every $j \in \mathbb{N}$, $\varepsilon_j > 0$ and $\lim_{j \rightarrow +\infty} \varepsilon_j = 0$.
- (ii) $(\gamma_j)_{j \in \mathbb{N}}$ is nondecreasing sequence of positive reals and $\lim_{j \rightarrow +\infty} \gamma_j = +\infty$.

In addition, the convergence analysis hereafter is led under the following assumption on the involved functions.

Assumption 3.8. There exists some $j^* \in \mathbb{N}$ that such, for every $j \geq j^*$, F_{γ_j} is convex.

Note that the above assumption implies that F is convex on C , but it can be nonconvex on the whole space as shown next and illustrated in Figure 3.1.

Example 3.17. Assume that C is the closed ball of \mathbb{R}^N with center 0 and radius $\rho \in (0, +\infty)$. Let $R = d_C^2/2$ and let F be a continuously differentiable function which is convex on C and twice-differentiable on $\mathbb{R}^N \setminus C$ with its Hessian $\nabla^2 F$ satisfying the following property: there exists $(\alpha, \delta) \in (0, +\infty)^2$ with $\alpha \geq \rho\delta$ such that

$$(\forall \mathbf{x} \in \mathbb{R}^N \setminus C) \quad \nabla^2 F(\mathbf{x}) - \frac{\alpha}{\|\mathbf{x}\|} \left(\mathbf{I}_N - \frac{\mathbf{x}\mathbf{x}^\top}{\|\mathbf{x}\|^2} \right) \succeq -\delta I_N. \quad (3.5.3)$$

Then F_γ is convex if

$$\gamma\rho \geq \alpha. \quad (3.5.4)$$

Indeed (3.5.3) means that G defined as

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad G(\mathbf{x}) = F(\mathbf{x}) - \alpha\|\mathbf{x}\| + \frac{\delta}{2}\|\mathbf{x}\|^2 \quad (3.5.5)$$

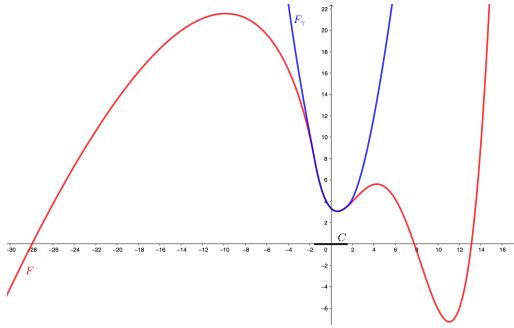


Figure 3.1: Illustration of Example 3.17 in the one-dimensional case. C is the closed ball of \mathbb{R} of center 0 and radius $\rho = 1.5$, and $R = d_C^2/2$. F is defined as $F(x) = 14.4\sqrt{|x|} + \ln(1+|x|)x - x - 10$ if $x \leq -\rho$, and $F(x) = P(x)$ if $x > -\rho$, where $P(x) = ax^4 + bx^3 + cx^2 - x + e$, is the unique polynomial function such that $P(-\rho) = F(-\rho)$, $P'(-\rho) = F'(-\rho)$, $P'(0.6) = 0$ and $P'(11.07) = 0$. On this illustration, F corresponds to the red curve and F_γ with $\gamma = 2$ to the blue curve. One can note that F is convex on C and nonconvex on \mathbb{R} , but for $\gamma \geq 2$, F_γ is convex. Moreover F_γ is coercive for any $\gamma > 0$.

is a convex function on every convex subset of $\mathbb{R}^N \setminus C$. We have then

$$(\forall \mathbf{x} \in \mathbb{R}^N \setminus C) \quad F_\gamma(\mathbf{x}) = G(\mathbf{x}) + \Delta(\mathbf{x}), \quad (3.5.6)$$

where

$$\begin{aligned} \Delta(\mathbf{x}) &= \alpha\|\mathbf{x}\| - \frac{\delta}{2}\|\mathbf{x}\|^2 + \frac{\gamma}{2}d_C^2(\mathbf{x}) \\ &= \alpha\|\mathbf{x}\| - \frac{\delta}{2}\|\mathbf{x}\|^2 + \frac{\gamma}{2}(\|\mathbf{x}\| - \rho)^2. \end{aligned} \quad (3.5.7)$$

For every $\mathbf{x} \in \mathbb{R}^N \setminus C$,

$$\nabla^2 \Delta(\mathbf{x}) = (\gamma - \delta)I_N + \frac{\alpha - \gamma\rho}{\|\mathbf{x}\|} \left(I_N - \frac{\mathbf{x}\mathbf{x}^\top}{\|\mathbf{x}\|^2} \right) \quad (3.5.8)$$

If (3.5.4) holds, the minimum eigenvalue of $\nabla^2 \Delta(\mathbf{x})$ is

$$\begin{aligned} \gamma - \delta + (\alpha - \gamma\rho)/\|\mathbf{x}\| &\geq \gamma - \delta + (\alpha - \gamma\rho)/\rho \\ &= \alpha/\rho - \delta \geq 0, \end{aligned} \quad (3.5.9)$$

which shows that Δ is convex on every convex subset of $\mathbb{R}^N \setminus C$. We deduce from (3.5.6) that F_γ shares the same property. Since F is \mathcal{C}^1 and convex on C and $\nabla d_C^2 = 2(\text{Id} - \text{proj}_C)$ is continuous, F_γ is also \mathcal{C}^1 and convex on C . F_γ is thus convex on \mathbb{R}^N .

Note that a sufficient condition for (3.5.3) to be satisfied with $\alpha = \rho\delta$ is

$$(\forall \mathbf{x} \in \mathbb{R}^N \setminus C) \quad \lambda_{\min}(\nabla^2 F(\mathbf{x})) \geq -\delta \left(1 - \frac{\rho}{\|\mathbf{x}\|} \right), \quad (3.5.10)$$

where $\lambda_{\min}(\nabla^2 F(\mathbf{x}))$ is the minimum eigenvalue of the Hessian of F at \mathbf{x} .

3.5.3 Convergence analysis

Our convergence result is given in Theorem 3.5.3 and exhibits some similarity to the one obtained in [Luenberger, 1997, Chap. 10, Thm. 1] for the classical exact exterior penalty method.

Lemma 3.18. *Suppose that Assumptions 3.1 and 3.2 hold. Let $(\varepsilon_j)_{j \in \mathbb{N}}$ and $(\gamma_j)_{j \in \mathbb{N}}$ be two sequences satisfying Assumption 3.7. Suppose that Assumptions 3.3-3.6 and 3.8 are satisfied by $(F_{\gamma_j})_{j \in \mathbb{N}}$. Then the sequence $(\mathbf{x}_j)_{j \in \mathbb{N}}$ generated by Algorithm 9 is bounded.*

Proof. For every $j \in \mathbb{N}$, since Assumptions 3.1-3.6 are satisfied by F_{γ_j} , the convergence results established in Section 3.4.2.2 apply. Thus Algorithm 8 produces, in a finite number of iterations, a vector $\mathbf{x}_j \in \mathbb{R}^N$ verifying (3.5.2). This shows that Algorithm 9 is well-defined.

Let us now prove the boundedness of sequence $(\mathbf{x}_j)_{j \in \mathbb{N}}$. By definition of sequence $(\mathbf{x}_j)_{j \in \mathbb{N}}$ and because of the convexity of functions $(F_{\gamma_j})_{j \geq j^*}$, the Cauchy-Schwarz inequality yields for every $j \geq j^*$ and $\mathbf{z} \in \mathbb{R}^N$,

$$\begin{aligned} F_{\gamma_j}(\mathbf{x}_j) - F_{\gamma_j}(\mathbf{z}) &\leq \langle \nabla F_{\gamma_j}(\mathbf{x}_j), \mathbf{x}_j - \mathbf{z} \rangle, \\ &< \|\nabla F_{\gamma_j}(\mathbf{x}_j)\| \|\mathbf{x}_j - \mathbf{z}\|, \\ &\leq \varepsilon_j \|\mathbf{x}_j - \mathbf{z}\|. \end{aligned} \quad (3.5.11)$$

We then argue by contradiction. Assume that $(\mathbf{x}_j)_{j \in \mathbb{N}}$ is unbounded. Then there exists a subsequence of nonzero vectors $(\mathbf{x}_{j_q})_{q \in \mathbb{N}}$ and $\bar{\mathbf{x}} \in \mathbb{R}^N$ such that

$$\lim_{q \rightarrow +\infty} \|\mathbf{x}_{j_q}\| = +\infty, \quad (3.5.12)$$

$$\lim_{q \rightarrow +\infty} \frac{\mathbf{x}_{j_q}}{\|\mathbf{x}_{j_q}\|} = \bar{\mathbf{x}}, \quad (3.5.13)$$

$$\|\bar{\mathbf{x}}\| = 1. \quad (3.5.14)$$

Let $\mathbf{z} \in C$. According to (3.5.11), for every $q \in \mathbb{N}$ such that $j_q \geq j^*$,

$$\begin{aligned} \frac{F_{\gamma_{j_q}}(\mathbf{x}_{j_q})}{\|\mathbf{x}_{j_q}\|} &< \frac{F_{\gamma_{j_q}}(\mathbf{z})}{\|\mathbf{x}_{j_q}\|} + \varepsilon_{j_q} \left\| \frac{\mathbf{x}_{j_q}}{\|\mathbf{x}_{j_q}\|} - \frac{\mathbf{z}}{\|\mathbf{x}_{j_q}\|} \right\| \\ &= \frac{F(\mathbf{z})}{\|\mathbf{x}_{j_q}\|} + \varepsilon_{j_q} \left\| \frac{\mathbf{x}_{j_q}}{\|\mathbf{x}_{j_q}\|} - \frac{\mathbf{z}}{\|\mathbf{x}_{j_q}\|} \right\|, \end{aligned} \quad (3.5.15)$$

where we have used the fact that $\mathbf{z} \in C$, hence $R(\mathbf{z}) = 0$, to obtain the equality. Let us take the infimum limit on both sides. It follows from (3.5.12), (3.5.13), and Assumption 3.7(i), that

$$\liminf_{q \rightarrow +\infty} \frac{F_{\gamma_{j_q}}(\mathbf{x}_{j_q})}{\|\mathbf{x}_{j_q}\|} \leq 0. \quad (3.5.16)$$

Let $q_0 \in \mathbb{N}$. By using (3.5.12), (3.5.13) and the convexity (and continuity) of $F_{\gamma_j^*}$,

$$\begin{aligned} & F_{\gamma_j^*}(\mathbf{z} + \|\mathbf{x}_{j_{q_0}}\| \bar{\mathbf{x}}) \\ &= \lim_{q \rightarrow +\infty} F_{\gamma_j^*} \left(\left(1 - \frac{\|\mathbf{x}_{j_{q_0}}\|}{\|\mathbf{x}_{j_q}\|}\right) \mathbf{z} + \frac{\|\mathbf{x}_{j_{q_0}}\|}{\|\mathbf{x}_{j_q}\|} \mathbf{x}_{j_q} \right) \\ &\leq \liminf_{q \rightarrow +\infty} \left[\left(1 - \frac{\|\mathbf{x}_{j_{q_0}}\|}{\|\mathbf{x}_{j_q}\|}\right) F_{\gamma_j^*}(\mathbf{z}) + \frac{\|\mathbf{x}_{j_{q_0}}\|}{\|\mathbf{x}_{j_q}\|} F_{\gamma_j^*}(\mathbf{x}_{j_q}) \right] \\ &\leq \liminf_{q \rightarrow +\infty} \left[\left(1 - \frac{\|\mathbf{x}_{j_{q_0}}\|}{\|\mathbf{x}_{j_q}\|}\right) F(\mathbf{z}) + \frac{\|\mathbf{x}_{j_{q_0}}\|}{\|\mathbf{x}_{j_q}\|} F_{\gamma_{j_q}}(\mathbf{x}_{j_q}) \right]. \end{aligned} \quad (3.5.17)$$

The last inequality (3.5.17) is a consequence of the monotonicity of $(\gamma_j)_{j \in \mathbb{N}}$ and of the equality of $F_{\gamma_j^*}(\mathbf{z})$ and $F(\mathbf{z})$. By using (3.5.16), we deduce that

$$F_{\gamma_j^*}(\mathbf{z} + \|\mathbf{x}_{j_{q_0}}\| \bar{\mathbf{x}}) \leq F(\mathbf{z}). \quad (3.5.18)$$

Thus the sublevel set $\{\mathbf{x} \in \mathbb{R}^N \mid F_{\gamma_j^*}(\mathbf{x}) \leq F(\mathbf{z})\}$ contains the unbounded set $\{\mathbf{z} + \|\mathbf{x}_{j_{q_0}}\| \bar{\mathbf{x}}, q_0 \in \mathbb{N}\}$, which contradicts the coercivity of $F_{\gamma_j^*}$. \square

Theorem 3.19

Under the same assumptions as in Lemma 3.18, the sequence $(\mathbf{x}_j)_{j \in \mathbb{N}}$ generated by Algorithm 9 has at least one cluster point. In addition, every of its cluster points is a solution to Problem 3.5.

Proof. According to Lemma 3.18, $(\mathbf{x}_j)_{j \in \mathbb{N}}$ is bounded. Let $(\mathbf{x}_{j_q})_{q \in \mathbb{N}}$ be a subsequence of $(\mathbf{x}_j)_{j \in \mathbb{N}}$ which converges to a point $\mathbf{x}^* \in \mathbb{R}^N$. We shall show that $\mathbf{x}^* \in C$ and that, for every $\mathbf{z} \in C$, $F(\mathbf{x}^*) \leq F(\mathbf{z})$.

Let us first prove that $\mathbf{x}^* \in C$. Let $\mathbf{z} \in C$ and $q \in \mathbb{N}$ such that $j_q \geq j^*$. Following the same reasoning as for (3.5.11), and using the equality $F_{\gamma_{j_q}}(\mathbf{z}) = F(\mathbf{z})$, we have

$$F_{\gamma_{j_q}}(\mathbf{x}_{j_q}) < F(\mathbf{z}) + \varepsilon_{j_q} \|\mathbf{x}_{j_q} - \mathbf{z}\|. \quad (3.5.19)$$

Since the sequence $(\mathbf{x}_{j_q})_{q \in \mathbb{N}}$ is bounded and Assumption 3.7(i) holds, we deduce that there exists a constant $\nu \in \mathbb{R}$ such that

$$(\forall q \in \mathbb{N}) \quad F_{\gamma_{j_q}}(\mathbf{x}_{j_q}) \leq \nu. \quad (3.5.20)$$

In addition, since F is continuous and $(\mathbf{x}_j)_{j \in \mathbb{N}}$ is bounded, $(F(\mathbf{x}_{j_q}))_{q \in \mathbb{N}}$ is bounded from below by a constant $\zeta \in \mathbb{R}$ and the following inequality holds:

$$\begin{aligned} (\forall q \in \mathbb{N}) \quad F_{\gamma_{j_q}}(\mathbf{x}_{j_q}) &= F(\mathbf{x}_{j_q}) + \gamma_{j_q} R(\mathbf{x}_{j_q}) \\ &\geq \zeta + \gamma_{j_q} R(\mathbf{x}_{j_q}). \end{aligned} \quad (3.5.21)$$

By combining (3.5.20) and (3.5.21), we deduce that

$$(\forall q \in \mathbb{N}) \quad 0 \leq R(\mathbf{x}_{j_q}) \leq \frac{\nu - \zeta}{\gamma_{j_q}}. \quad (3.5.22)$$

Next, passing to the limit when $q \rightarrow +\infty$ in (3.5.22) and using Assumption 3.7(ii), we obtain by continuity

of R ,

$$R(\mathbf{x}^*) = 0, \quad (3.5.23)$$

which, according to Definition 3.1, is equivalent to $\mathbf{x}^* \in C$.

Let us now show that, for every $\mathbf{z} \in C$, $F(\mathbf{x}^*) \leq F(\mathbf{z})$. Equation (3.5.19) implies that

$$\begin{aligned} F(\mathbf{x}_{j_q}) &\leq F(\mathbf{z}) + \varepsilon_{j_q} \|\mathbf{x}_{j_q} - \mathbf{z}\| - \gamma_{j_q} R(\mathbf{x}_{j_q}) \\ &\leq F(\mathbf{z}) + \varepsilon_{j_q} \|\mathbf{x}_{j_q} - \mathbf{z}\|. \end{aligned} \quad (3.5.24)$$

Passing to the limit when $q \rightarrow +\infty$ in (3.5.24), we obtain by continuity of F ,

$$F(\mathbf{x}^*) \leq F(\mathbf{z}), \quad (3.5.25)$$

which completes the proof. \square

Corollary 3.20. *Under the same assumptions as in Lemma 3.18, if Problem 3.5 admits a unique solution $\mathbf{x}^* \in \mathbb{R}^N$, then the sequence $(\mathbf{x}_j)_{j \in \mathbb{N}}$ generated by Algorithm 9 converges to \mathbf{x}^* .*

Remark 3.21.

- (i) Our convergence proof remains valid if, at each iteration $j \in \mathbb{N}$ of the proposed inexact penalty method, the subproblem (\mathcal{P}_{γ_j}) is solved by any method returning an inexact minimizer $\mathbf{x}_j \in \mathbb{R}^N$ such that (3.5.2) holds.
- (ii) The whole theory is conducted here with a single sequence of penalty parameters for the global penalty function R . However, in the case where there are r penalty functions that we wish to weight differently with sequences of parameters $(\gamma_j^{(1)})_{j \in \mathbb{N}}, \dots, (\gamma_j^{(r)})_{j \in \mathbb{N}}$, a similar proof may be carried.

3.5.4 Initial guess at each iteration

The choice of the initial point $\mathbf{x}_j^{(0)}$ at each iteration can influence the convergence speed of the algorithm, so one should define it carefully. Ideally, the latter must be determined in such a way that $\|\nabla F_{\gamma_j}(\mathbf{x}_j^{(0)})\|$ is as small as possible, with

$$\nabla F_{\gamma_j}(\mathbf{x}_j^{(0)}) = \nabla F(\mathbf{x}_j^{(0)}) + \gamma_j \nabla R(\mathbf{x}_j^{(0)}). \quad (3.5.26)$$

We propose a heuristic algorithm initialization, relying on the two previous iterates, \mathbf{x}_{j-1} and \mathbf{x}_{j-2} , and on the penalty parameters γ_{j-2} , γ_{j-1} , γ_j , when $j \geq 2$. In a neighborhood $\mathcal{N}(\mathbf{x}^*)$ of the optimum \mathbf{x}^* for Problem (\mathcal{P}) , we suppose that

$$\begin{aligned} (\forall \mathbf{x} \in \mathcal{N}(\mathbf{x}^*)) \quad \nabla F_{\gamma}(\mathbf{x}) &\simeq \nabla F(\mathbf{x}^*) + \gamma \nabla R(\mathbf{x}) \\ &\simeq \nabla F(\mathbf{x}^*) + 2\gamma(\mathbf{x} - \mathbf{x}^*). \end{aligned} \quad (3.5.27)$$

The latter approximation is valid if \mathbf{x}^* lies on the boundary of C , $\mathbf{x} \notin C$, and

$$(\forall \mathbf{x} \in \mathcal{N}(\mathbf{x}^*) \setminus C) \quad R(\mathbf{x}) \simeq \|\mathbf{x} - \mathbf{x}^*\|^2 \quad (3.5.28)$$

(e.g., $R = d_C^2$). Then, using the approximation (3.5.27) for the values (γ, x) in $\{(\gamma_{j-2}, \mathbf{x}_{j-2}), (\gamma_{j-1}, \mathbf{x}_{j-1})\}$ and assuming that $\nabla F_{\gamma_{j-2}}(\mathbf{x}_{j-2}) \simeq 0$ and $\nabla F_{\gamma_{j-1}}(\mathbf{x}_{j-1}) \simeq 0$, we obtain

$$\mathbf{x}_{j-1} - \mathbf{x}_{j-2} \simeq \frac{1}{2} \nabla F(\mathbf{x}^*) \left(\frac{1}{\gamma_{j-2}} - \frac{1}{\gamma_{j-1}} \right). \quad (3.5.29)$$

Similarly, to get $\nabla F_{\gamma_j}(\mathbf{x}_j^{(0)}) \simeq 0$, we should have

$$\mathbf{x}_j^{(0)} - \mathbf{x}_{j-1} \simeq \frac{1}{2} \nabla F(\mathbf{x}^*) \left(\frac{1}{\gamma_{j-1}} - \frac{1}{\gamma_j} \right). \quad (3.5.30)$$

Gathering (3.5.29) and (3.5.30) leads to the following choice for the initial point:

$$\mathbf{x}_j^{(0)} = \mathbf{x}_{j-1} + \frac{\frac{1}{\gamma_{j-1}} - \frac{1}{\gamma_j}}{\frac{1}{\gamma_{j-2}} - \frac{1}{\gamma_{j-1}}} (\mathbf{x}_{j-1} - \mathbf{x}_{j-2}). \quad (3.5.31)$$

3.6 Numerical experiments

We consider two applicative scenarios with the aim to highlight the computational efficiency of our approach on large-scale image recovery problems, as well as to illustrate the variety of constraints that can be handled. All algorithms are implemented in Python 3 and the computations are performed on a desktop having an Intel Xeon 3.2 GHz processor and 16 GB of RAM.

3.6.1 Frame-based image restoration in presence of heteroscedastic noise

The first numerical scenario we consider concerns image restoration in the presence of pixel-dependent additive Gaussian noise. We emphasize that the optimization problem addressed hereafter is a generalization of a standard wavelet denoising problem [Afonso et al., 2010b].

We consider an image restoration scenario aiming at recovering an original image $\bar{\mathbf{y}} \in \mathbb{R}^M$ from a degraded version of it, $\mathbf{z} = (z_i)_{1 \leq i \leq M} \in \mathbb{R}^M$, given the observation model

$$\mathbf{z} = \mathbf{H}\bar{\mathbf{y}} + \mathbf{w}, \quad (3.6.1)$$

where $\mathbf{H} \in \mathbb{R}^{M \times M}$ is a matrix modelling a blur and $\mathbf{w} = (w_i)_{1 \leq i \leq M} \in \mathbb{R}^M$ is an additive noise. The noise is assumed to be zero-mean Gaussian i.i.d. and heteroscedastic (i.e. with non stationary variance). As discussed in [Foi, 2009, Thai et al., 2013], heteroscedastic noise is commonly encountered when dealing with raw digital images of an under or over-exposed scene, as it is often the case in satellite imaging. A suitable model for the noise variance that was used in [Foi, 2009] is given by

$$(\forall i \in \{1, \dots, M\}) \quad w_i = \sigma((\mathbf{H}\bar{\mathbf{y}})_i) v_i, \quad (3.6.2)$$

where $(\cdot)_i$ designates the i -th component of its vector in argument, $(v_i)_{1 \leq i \leq M}$ are realizations of independent zero-mean, unit-variance Gaussian random variables, and

$$(\forall t \in \mathbb{R}) \quad \sigma(t) = \begin{cases} \sqrt{at + b} & \text{if } at + b \geq 0 \\ \sqrt{b} & \text{otherwise,} \end{cases} \quad (3.6.3)$$

with $(a, b) \in (\mathbb{R}^+)^2$.

3.6.1.1 Tight frame decomposition

Restoring \bar{y} from its degraded version z is an ill-posed inverse problem which resolution requires the introduction of prior knowledge on the sought image. The prior can either be formulated in the original data space of the image, or through an appropriate linear representation (e.g., Fourier, cosine, wavelet domain) of it. In this experiment, we opt for a sparsity enhancing prior in a frame domain [Mallat, 1999]. More precisely, we consider the decomposition of the image on a tight frame [Pustelnik et al., 2016, Carrillo et al., 2012, Durand et al., 2009] i.e., an overcomplete dictionary of vectors $(e_1, \dots, e_N) \in (\mathbb{R}^M)^N$ with $N > M$ such that the operator

$$W: \mathbb{R}^M \mapsto \mathbb{R}^N : y \mapsto (\langle y, e_n \rangle)_{1 \leq n \leq N} \quad (3.6.4)$$

satisfies $W^*W = \mu I_M$ for some parameter $\mu > 0$, where W^* refers to the adjoint of W . Such an overcomplete decomposition leads to a more flexible linear model than traditional orthonormal representations. In particular, if we set $\mu \in \mathbb{N}^*$, one can build a tight frame with $W^*W = \mu I_M$ by concatenating μ orthonormal wavelet bases, associated to the decomposition operator

$$W = [W_1^* | \dots | W_\mu^*]^* \in \mathbb{R}^{N \times M}, \quad (3.6.5)$$

where $N = \mu M$ and $(W_i)_{1 \leq i \leq \mu} \in (\mathbb{R}^{M \times M})^\mu$ are orthogonal operators.

3.6.1.2 Optimization problem

Let $W \in \mathbb{R}^{N \times M}$ be the tight frame decomposition operator introduced above. We propose to follow the so-called synthesis approach for solving the inverse problem (3.6.1). This approach consists of estimating $\bar{x} \in \mathbb{R}^N$, the decomposition of the original image \bar{y} onto the tight frame, by solving an optimization problem in the space of the frame coefficients. A solution to the initial Problem (3.6.1) is then retrieved as $\bar{y} = W^* \bar{x}$.

When the Gaussian noise is identically distributed for every pixel, the minimization problem on the tight frame coefficients usually takes the form

$$\underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad \|HW^*\mathbf{x} - z\|^2 + \lambda F(\mathbf{x}), \quad (3.6.6)$$

where $F: \mathbb{R}^N \rightarrow \mathbb{R}$ is a regularization function on the frame coefficients, and $\lambda > 0$ a regularization parameter. The main drawback of formulation (3.6.6) is the difficulty to adjust the regularization parameter λ . This can be alleviated by considering instead the constrained minimization problem [Harizanov et al., 2013, Carrillo et al., 2012]:

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad F(\mathbf{x}) \\ & \text{subject to} \quad \|HW^*\mathbf{x} - z\|^2 \leq \alpha, \end{aligned} \quad (3.6.7)$$

in which the fidelity to the model is ensured by means of a constraint. By virtue of the law of large numbers, the new parameter $\alpha > 0$ can be estimated as $\alpha \simeq \sigma^2 M$, where σ is the standard deviation of

the noise.

In the present case, the heteroscedastic noise follows the standard deviation rule (3.6.3) and using directly the formulation (3.6.7) might not be appropriate to account for the local properties of such a noise. We thus propose to partition the image into $L \in \mathbb{N}^*$ rectangular blocks of $B \in \mathbb{N}^*$ pixels and to impose a taylored data fidelity constraint on each of these blocks. For every $\ell \in \{1, \dots, L\}$, we denote by $[\cdot]_\ell$ the component corresponding to the ℓ -th block. The minimization problem we propose to solve then reads

$$\begin{aligned} & \underset{\boldsymbol{x} \in \mathbb{R}^N}{\text{minimize}} \quad F(\boldsymbol{x}), \\ & \text{subject to} \quad \|[\boldsymbol{H}\boldsymbol{W}^*\boldsymbol{x} - \boldsymbol{z}]_\ell\|^2 \leq \alpha_\ell \quad (\forall \ell \in \{1, \dots, L\}), \\ & \quad \boldsymbol{W}^*\boldsymbol{x} \in [0, 1]^M. \end{aligned} \quad (3.6.8)$$

Hereabove, F is a smooth ℓ_1 -regularization promoting sparsity of the frame coefficients chosen as

$$(\forall \boldsymbol{x} = (\xi_n)_{1 \leq n \leq N} \in \mathbb{R}^N) \quad F(\boldsymbol{x}) = \sum_{n=1}^N \lambda_n \sqrt{\delta + \xi_n^2}, \quad (3.6.9)$$

with, for every $n \in \{1, \dots, N\}$, $\lambda_n \in (0, +\infty)$. Moreover, $\delta \in (0, +\infty)$ and $(\alpha_\ell)_{1 \leq \ell \leq L} \in (\mathbb{R}^+)^L$ are given hyper-parameters. Note that the optimization problem (3.6.8) is an instance of Problem 3.5 where function F is defined as in (3.6.9) and constraint set C corresponds to

$$\{\boldsymbol{x} \in \mathbb{R}^N \mid (\forall \ell \in \{1, \dots, L\}) \|[\boldsymbol{H}\boldsymbol{W}^*\boldsymbol{x} - \boldsymbol{z}]_\ell\|^2 \leq \alpha_\ell\} \cap \{\boldsymbol{x} \in \mathbb{R}^N \mid \boldsymbol{W}^*\boldsymbol{x} \in [0, 1]^M\}. \quad (3.6.10)$$

Remark 3.22. Regarding the partition of the image, we opted here for a rectangular one as it was the most straightforward for the considered image. However one could also consider using a partition based on a segmentation of the degraded image when such a segmentation is available.

3.6.1.3 Proposed algorithm

We propose to apply the methods P-MMS and P-MMS^{loc} to the minimization problem (3.6.8). To do so, we have to define the penalty functions for the data-fidelity constraints. For every $\ell \in \{1, \dots, L\}$, we set

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad R_{1,\ell}(\boldsymbol{x}) = d_{\mathcal{B}_\ell}^2([\boldsymbol{H}\boldsymbol{W}^*\boldsymbol{x}]_\ell), \quad (3.6.11)$$

where \mathcal{B}_ℓ is the Euclidean ball of \mathbb{R}^B centered at $[\boldsymbol{z}]_\ell$ with radius α_ℓ . Similarly, we define the penalty function for the pixel-range constraint by

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad R_2(\boldsymbol{x}) = \frac{1}{10} d_{[0,1]^M}^2(\boldsymbol{W}^*\boldsymbol{x}). \quad (3.6.12)$$

This latter penalty function will be split into M penalty functions operating on each component of $\boldsymbol{W}^*\boldsymbol{x}$, which naturally provides a suitable applicative context for the local approach described in Section 3.4. In a nutshell, the penalty function for Problem (3.6.8) reads

$$(\forall \boldsymbol{x} \in \mathbb{R}^N) \quad R(\boldsymbol{x}) = \sum_{\ell=1}^L R_{1,\ell}(\boldsymbol{x}) + R_2(\boldsymbol{x}). \quad (3.6.13)$$

The penalized MMS method relies on Assumption 3.2, which requires building explicitly quadratic

majorants of the objective F and the penalty functions $(R_{1,\ell})_{1 \leq \ell \leq L}$ and R_2 . It can be easily shown (see [Chan and Mulet, 1999] for instance) that, for every $\mathbf{x} = (\xi_n)_{1 \leq n \leq N} \in \mathbb{R}^N$, the matrix

$$\mathbf{A}_F(\mathbf{x}) = \text{Diag} \left\{ \left(\frac{\lambda_n}{\sqrt{\delta + \xi_n^2}} \right)_{1 \leq n \leq N} \right\} \in \mathbb{R}^{N \times N} \quad (3.6.14)$$

defines the curvature of a valid quadratic majorant of F at \mathbf{x} . Regarding the penalty functions, according to Example 3.8, quadratic majorants for $(R_{1,\ell})_{1 \leq \ell \leq L}$ are given by the curvature matrices

$$(\forall \ell \in \{1, \dots, L\})(\forall \mathbf{x} \in \mathbb{R}^N) \mathbf{A}_{R_{1,\ell}}(\mathbf{x}) = 2\mathbf{W}\mathbf{H}^\top \mathbf{T}_\ell \mathbf{H}\mathbf{W}^*, \quad (3.6.15)$$

where $\mathbf{T}_\ell = \text{Diag} \left\{ (t_i)_{1 \leq i \leq M} \right\} \in \mathbb{R}^{M \times M}$ with $t_i = 1$ if the i -th pixel belongs to the ℓ -th block, $t_i = 0$ otherwise. Similarly, we define

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{A}_{R_2}(\mathbf{x}) = \frac{1}{5} \mathbf{W}\mathbf{W}^*. \quad (3.6.16)$$

By assuming that set C defined by (3.6.10) is nonempty¹, for both P-MMS and P-MMS^{loc}, the convergence conditions introduced in Section 3.5.3 are met. Indeed, Assumptions 3.1-3.4 and Assumption 3.6 are fulfilled by functions F and R . By choosing the subspace as the memory gradient one in (3.3.11), Assumption 3.5 is satisfied. Finally, penalty parameters $(\gamma_j)_{j \in \mathbb{N}}$ in P-MMS or P-MMS^{loc} were set to $\gamma_j = (3j)^{0.8}$ and the precision parameters $(\varepsilon_j)_{j \in \mathbb{N}}$ were chosen as $\varepsilon_j = 40/(\gamma_j)^{0.25}$, so that the requirements in Assumption 3.7 are satisfied. Assumption 3.8 also holds since F and R are convex.

3.6.1.4 Results

For our numerical experiment, we consider the satellite `zaghouane` image, of size $M = 512 \times 512$, blurred by an 11×11 Gaussian kernel with standard deviation 1.7 using symmetric boundary conditions. The image is corrupted with a Gaussian noise following the standard deviation rule (3.6.3), with $a = 10^{-4}$ and $b = 10^{-3}$ (Figure 3.2 (left)). The tight frame operator \mathbf{W} is defined as a concatenation of $\mu = 3$ orthonormal wavelet bases, each one with 4 resolution levels: Daubechies of order 2, Coiflets of order 3, and Symlets of order 6 (respectively `db2`, `coif3`, `sym6` from the `pywt` Python package). The size of the considered optimization problem is therefore $N = 3M$. In the numerical implementation of the algorithm, the operators \mathbf{W} and \mathbf{W}^* are not stored as matrices; they are defined as functions returning a matrix-vector product relying on the discrete wavelet transform of Python. In (3.6.9), $\lambda_n = 1$ if ξ_n is a detail (i.e. high frequency) wavelet coefficient, whereas $\lambda_n = 10^{-4}$ if ξ_n is an approximation (i.e. low frequency) coefficient. The image is divided into a partition of $L = 256$ squares of size $B = 32 \times 32$ pixels. The hyper-parameters are set to $\delta = 10^{-4}$ and, for every $\ell \in \{1, \dots, L\}$, $\alpha_\ell = 0.9 \hat{\sigma}_\ell^2 B$, where $\hat{\sigma}_\ell$ is the empirical standard deviation of the noise on the ℓ -th block, estimated as

$$\hat{\sigma}_\ell^2 \simeq \frac{a}{B} \sum_{i \in \ell\text{-th block}} z_i + b. \quad (3.6.17)$$

We compare the convergence speed of P-MMS and its local acceleration P-MMS^{loc}, with four state-

¹Since \mathbf{W}^* is surjective, this means that there exists $y \in [0, 1]^M$ such that, for every $\ell \in \{1, \dots, L\}$, $\|[\mathbf{H}y - z]_\ell\|^2 \leq \alpha_\ell$.

of-the-art algorithms, namely the primal-dual Condat-Vũ algorithm (CV) [Condat, 2012, Vũ, 2013], the parallel proximal algorithm (PPXA+) [Combettes and Pesquet, 2011], FISTA [Beck and Teboulle, 2009] and the SQP trust region algorithm of Byrd et al. [Byrd et al., 1999] (trust-constr). FISTA is implemented using the improved scheme [Chambolle and Dossal, 2015] where subiterations for computing the proximity operator are performed with an accelerated dual Forward-Backward algorithm [Abboud et al., 2017]. The trust region method is a solver already implemented in the Python package `scipy.optimize`. Note that PPXA+ is similar to a parallel version of ADMM [Setzer et al., 2010, Boyd et al., 2011].

The restored image using formulation (3.6.8) is displayed in Figure 3.2 (right). We compare, in Figure 3.3, the convergence speed of the algorithms in terms of the relative distance between the outer iterate (corresponding to \mathbf{x}_j) and the limit point x_∞ , computed after a large number of iterations for each method. One can observe that the proposed method P-MMS^{loc} outperforms all the other competitors. The advantage of resorting to the local majorization strategy can be observed when comparing P-MMS and P-MMS^{loc} curves.

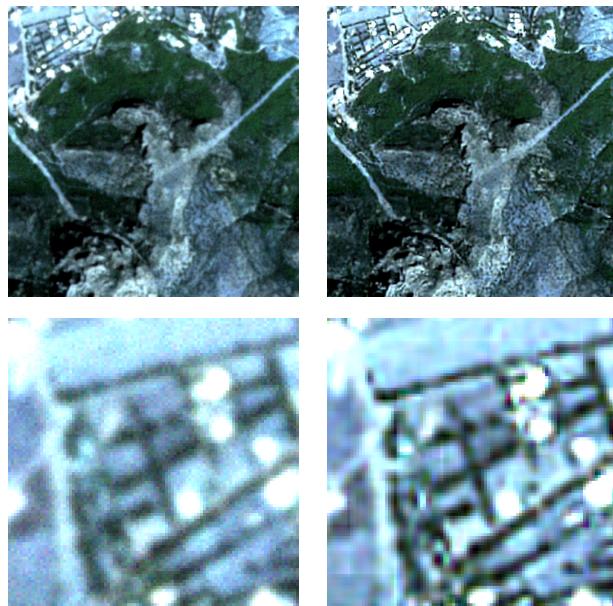


Figure 3.2: (top left) Degraded image, PNSR = 29.90dB ; (top right) Restored image, PNSR = 31.58dB; (bottom left) Zoom on the degraded image ; (bottom right) Zoom on the restored image.

3.6.2 Image reconstruction in the presence of Poisson noise

Our second example focuses on a medical imaging reconstruction problem, typically encountered in Positron Emission Tomography (PET) imaging.

3.6.2.1 Positron Emission Tomography

PET is a medical imaging technique used to detect the presence of specific molecules enriched in tissues (e.g. tumors) of an organ by means of an appropriate tracer injected to the patient. The tracer is marked

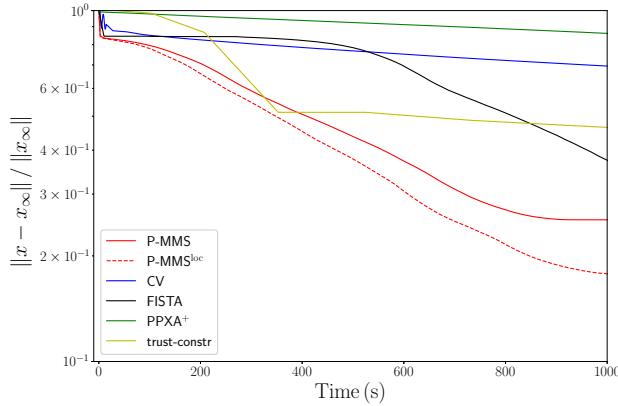


Figure 3.3: Distance to the optimum versus time.

by a radioactive element that emits a positron when it disintegrates. The positron then annihilates with an electron. This results in the emission of two photons travelling in two opposite directions (see Figure 3.4(a)), making it possible to identify a “line-of-response” (LOR) in which the annihilation occurred (see Figure 3.4(b)). The PET acquisition of a 2D scene, represented as a vector $\bar{\mathbf{x}} \in \mathbb{R}^N$, is a so-called sinogram $\mathbf{y} \in \mathbb{R}^M$ gathering the number of disintegrations detected per pair of detectors. This leads to the observation model:

$$\mathbf{y} = \mathcal{P}(\mathbf{H}\bar{\mathbf{x}}), \quad (3.6.18)$$

where $\mathbf{H} \in \mathbb{R}^{M \times N}$ is the projection matrix whose elements $H_{m,n}$ represents the intersection area between the n -th pixel and the m -th LOR, and \mathcal{P} models a Poisson noise.

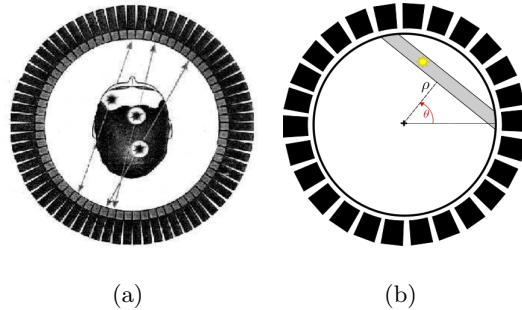


Figure 3.4: (a) Two photons are emitted in opposite directions and are detected almost simultaneously by two different detectors. (b) The annihilation is assumed to occur within an LOR delimited by the two detectors [Pustelnik, 2010].

3.6.2.2 Considered optimization problem

We propose to tackle the inverse problem (3.6.18) of reconstructing $\bar{\mathbf{x}}$ from the sinogram \mathbf{y} and the geometry matrix \mathbf{H} , by minimizing a data fidelity function reflecting the properties of Poisson noise, under a Total Variation (TV) regularization constraint and a pixel range constraint. A typical choice to account for Poisson noise [Dupe et al., 2009, Harizanov et al., 2013] consists of making use of the

Anscombe variance stabilizing transform [Anscombe, 1948],

$$\begin{aligned} T : \quad [0, +\infty)^M &\longrightarrow [0, +\infty)^M \\ \mathbf{u} = (u_i)_{1 \leq i \leq M} &\longmapsto \left(2\sqrt{u_i + \frac{3}{8}}\right)_{1 \leq i \leq M}, \end{aligned} \quad (3.6.19)$$

which approximately transforms the Poisson noise into an i.i.d. Gaussian noise with zero-mean and unit variance. The associated data-fidelity function is then defined as

$$(\forall \mathbf{x} \in \mathcal{D}) \quad F(\mathbf{x}) = \|T(\mathbf{Hx}) - T(\mathbf{y})\|^2, \quad (3.6.20)$$

with the domain

$$\mathcal{D} = \{\mathbf{x} \in \mathbb{R}^N \mid \mathbf{Hx} \in [0, +\infty)^M\}. \quad (3.6.21)$$

One can rewrite (3.6.20), using (3.6.19):

$$(\forall \mathbf{x} \in \mathcal{D}) \quad F(\mathbf{x}) = 4 \sum_{i=1}^M \left((\mathbf{Hx})_i + \frac{3}{8} \right) + 4 \sum_{i=1}^M (T(\mathbf{y}))_i \rho((\mathbf{Hx})_i) + \|T(\mathbf{y})\|^2, \quad (3.6.22)$$

where $(\cdot)_i$ denotes the i -th component and

$$(\forall v \in [0, +\infty)) \quad \rho(v) = -\sqrt{v + 3/8}. \quad (3.6.23)$$

In order to extend the domain of F to \mathbb{R}^N , we introduce the above differentiable extension of ρ on \mathbb{R} :

$$(\forall v \in \mathbb{R}) \quad \bar{\rho}(v) = \begin{cases} \rho(v) & \text{if } v \geq 0, \\ -\sqrt{\frac{2}{3}}v - \sqrt{3/8} & \text{if } v < 0, \end{cases} \quad (3.6.24)$$

which yields the following extension of F on the whole space:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad F(\mathbf{x}) = 4 \sum_{i=1}^M \left((\mathbf{Hx})_i + \frac{3}{8} \right) + 4 \sum_{i=1}^M (T(\mathbf{y}))_i \bar{\rho}((\mathbf{Hx})_i) + \|T(\mathbf{y})\|^2. \quad (3.6.25)$$

The optimization problem to tackle is then

$$\begin{aligned} &\underset{\mathbf{x} \in \mathbb{R}^N}{\text{minimize}} \quad F(\mathbf{x}), \\ &\text{subject to} \quad \text{TV}(\mathbf{x}) \leq \alpha \text{ and } \mathbf{x} \in [0, x_{\max}]^N, \end{aligned} \quad (3.6.26)$$

where $\alpha \in (0, +\infty)$ is the TV bound acting as a regularization parameter, and $x_{\max} \in (0, +\infty)$ is the maximal pixel intensity of the sought image $\bar{\mathbf{x}}$. Moreover, TV stands for the total variation semi-norm [Rudin et al., 1992]:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \text{TV}(\mathbf{x}) = \sum_{n=1}^N \|(\mathbf{Gx})_n\|_2, \quad (3.6.27)$$

where $G \in \mathbb{R}^{2N \times N}$ is the 2D discrete gradient operator. A natural choice for the regularization parameter would be $\alpha = \text{TV}(\bar{\mathbf{x}})$. As discussed in [Combettes and Pesquet, 2004], opting for a slightly smaller value of α (and thus over-regularizing) can improve the reconstruction performance.

The optimization problem (3.6.26) is an instance of Problem 3.5 with F given by (3.6.25) and

$$C = \{\mathbf{x} \in \mathbb{R}^N \mid \text{TV}(\mathbf{x}) \leq \alpha \text{ and } \mathbf{x} \in [0, x_{\max}]^N\}. \quad (3.6.28)$$

As we will see, our penalized MMS approach, although designed for differentiable objective functions, is able to handle nonsmooth regularization constraints such as the TV constraint involved here.

3.6.2.3 Proposed algorithm

We propose to apply the penalized MMS method described in Sections 3.4 and 3.5 to the resolution of (3.6.26). To do so, we introduce the penalty function for the TV constraint:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad R_1(\mathbf{x}) = d_B^2(\mathbf{Gx}), \quad (3.6.29)$$

where \mathcal{B} is the $\ell_{1,2}$ -ball with radius α , i.e.

$$\mathcal{B} = \{\mathbf{g} = (\mathbf{g}_n)_{1 \leq n \leq N} \in (\mathbb{R}^2)^N \mid \sum_{n=1}^N \|\mathbf{g}_n\|_2 \leq \alpha\}. \quad (3.6.30)$$

Moreover, we define the penalty function for the pixel-range constraint as

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad R_2(\mathbf{x}) = 2 d_{[0, x_{\max}]^N}^2(\mathbf{x}). \quad (3.6.31)$$

Finally, the penalty function for problem (3.6.26) is

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad R(\mathbf{x}) = R_1(\mathbf{x}) + R_2(\mathbf{x}). \quad (3.6.32)$$

Our MMS approach requires the design of quadratic tangent majorants for the functions F , R_1 , and R_2 , respectively defined in (3.6.25), (3.6.29) and (3.6.31).

To determine a majorant function for F , we rely on the following proposition extending the result from [Erdogan and Fessler, 2002].

Proposition 3.23. *Let $\rho: [0, +\infty) \rightarrow \mathbb{R}$ be a convex, twice continuously differentiable function with concave derivative ρ' . We denote by $\rho'(0)$ (resp. $\rho''(0)$) the right-hand derivative (resp. second-order derivative) of ρ at 0. Let $\bar{\rho}$ the continuously differentiable extended version of ρ on \mathbb{R} , given by*

$$(\forall v \in \mathbb{R}) \quad \bar{\rho}(v) = \begin{cases} \rho(v) & \text{if } v \geq 0, \\ \rho'(0)v + \rho(0) & \text{if } v < 0. \end{cases} \quad (3.6.33)$$

Then, $\bar{\rho}$ satisfies the following majorization property at every $\tau \in \mathbb{R}$:

$$(\forall v \in \mathbb{R}) \quad \bar{\rho}(v) \leq \bar{\rho}(\tau) + \bar{\rho}'(\tau)(v - \tau) + \frac{1}{2}\omega(\tau)(v - \tau)^2, \quad (3.6.34)$$

with $\bar{\rho}'$ the derivative of $\bar{\rho}$ and

$$(\forall \tau \in \mathbb{R}) \quad \omega(\tau) = \begin{cases} \frac{\bar{\rho}'(|\tau|) - \bar{\rho}'(0)}{|\tau|} & \text{if } \tau \neq 0, \\ \rho''(0) & \text{if } \tau = 0. \end{cases} \quad (3.6.35)$$

A proof of Proposition 3.23 is given in Appendix 3.A. In particular, Proposition 3.23 holds for ρ in (3.6.23) and $\bar{\rho}$ defined in (3.6.24). The majorant obtained in Proposition 3.23 is more accurate than the one based on the descent lemma ([Bauschke and Combettes, 2011, Lemma 2.64]) as illustrated in the example of Figure 3.5. This behaviour should be beneficial for the practical convergence speed of our method.

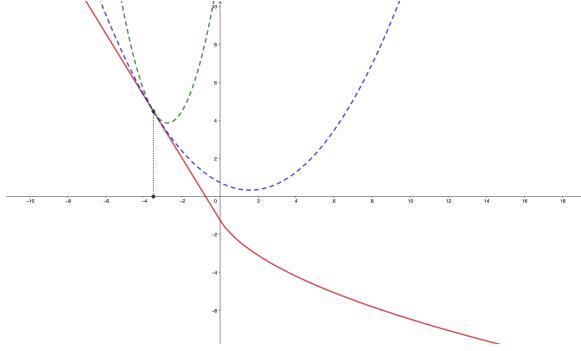


Figure 3.5: Comparison of two strategies for majorizing function $\bar{\rho}$ (red) around $v = -1$. The green curve is the majorant resulting from the descent lemma (i.e., exploiting the Lipschitz constant of $\bar{\rho}'$). The blue curve is the majorant obtained with our Proposition 3.23.

Using now Lemma 2.26 and Proposition 3.23, we build a quadratic majorant of F at $\mathbf{x} \in \mathbb{R}^N$ characterized by the curvature matrix:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{A}_F(\mathbf{x}) = 4\mathbf{H}^\top \text{Diag} \left\{ (T(\mathbf{y})_i \omega((\mathbf{H}\mathbf{x})_i))_{1 \leq i \leq M} \right\} \mathbf{H} + \epsilon \mathbf{I}_N,$$

with ϵ a small positive parameter which ensures that the curvature matrix is positive. Quadratic majorants of R_1 and R_2 can be obtained following our Example 3.8, which yields the respective curvatures:

$$(\forall \mathbf{x} \in \mathbb{R}^N) \quad \mathbf{A}_{R_1}(\mathbf{x}) = 2\mathbf{G}^\top \mathbf{G} \quad \text{and} \quad \mathbf{A}_{R_2}(\mathbf{x}) = 4 \cdot \mathbf{I}_N. \quad (3.6.36)$$

We illustrate the performance of P-MMS and P-MMS^{loc} methods. For both algorithms, since the constraint set C defined in (3.6.28) is nonempty (it contains the zero image), the convergence conditions derived in Section 3.5.3 are met. Indeed, Assumptions 3.1-3.4 and 3.6 are fulfilled by functions F and R defined in (3.6.32). In addition, as the memory gradient subspace (3.3.11) was chosen, Assumption 3.5 is satisfied. Finally, penalty parameters $(\gamma_j)_{j \in \mathbb{N}}$ in P-MMS and P-MMS^{loc} were set to $(\forall j \in \mathbb{N}) \quad \gamma_j = (0.002j)^{0.6}$, and the precision parameters $(\varepsilon_j)_{j \in \mathbb{N}}$ were chosen as $(\forall j \in \mathbb{N}) \quad \varepsilon_j = 0.01/(\gamma_j)^{0.15}$. These sequences satisfy the requirements of Assumption 3.7. Assumption 3.8 also holds since F and R are convex.

3.6.2.4 Results

We simulate a realistic example using the brain phantom image $\bar{\mathbf{x}}$ from the dataset [Belzunce, 2018] of size $N = 128 \times 128$ pixels and intensity range $[0, 200]$ displayed in Figure 3.6 (left). Measurements are generated using Model (3.6.18) where \mathbf{H} models a simplified 2D PET geometry with 144 detectors regularly distributed on a circle with radius equals to $64\sqrt{2} + 16$ pixel units, leading to $M = 10296$ pairs

of detectors and approximately $4 \cdot 10^5$ Poisson counts. The hyper-parameters were set to $x_{\max} = 200$, $\alpha = 0.9 \text{TV}(\bar{x})$, and $\epsilon = 10^{-3}$.

We again compare the convergence speed of P-MMS and its local acceleration P-MMS^{loc} with the primal-dual CV algorithm [Condat, 2012, Vũ, 2013], PPXA+ [Combettes and Pesquet, 2011], and the improved FISTA [Beck and Teboulle, 2009] scheme from [Chambolle and Dossal, 2015] using the method [Abboud et al., 2017] for the inner proximity steps. All the aforementioned algorithms require projection steps onto the $\ell_{1,2}$ -ball, so as to handle the TV term. This can be expressed as a function of the projection on the ℓ_1 -ball, that we implemented using Condat's algorithm in [Condat, 2015]. Note that a comparison to a second-order algorithm was not possible as the objective function F is not twice differentiable and the TV function is not differentiable.

The reconstructed image is displayed in Figure 3.6 (right). One can assess the good quality of reconstruction obtained with the proposed formulation (3.6.26), compared to the use of a simpler least-square approach coupled with a quadratic regularization (Figure 3.6 (bottom)). We display in Figure 3.7 the evolution of the relative distance between the outer iterate and the limit point x_∞ , for each method. Hereagain, one can notice that the proposed method P-MMS^{loc} outperforms all the other competitors.

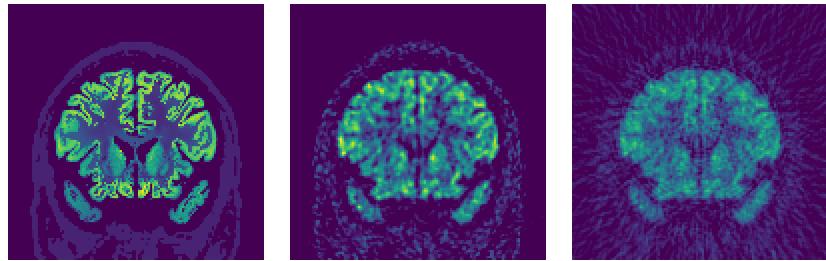


Figure 3.6: (top left) Original image ; (top right) Reconstructed image with model (3.6.26) ; (bottom) Reconstructed image with a Tikhonov least squares approach.

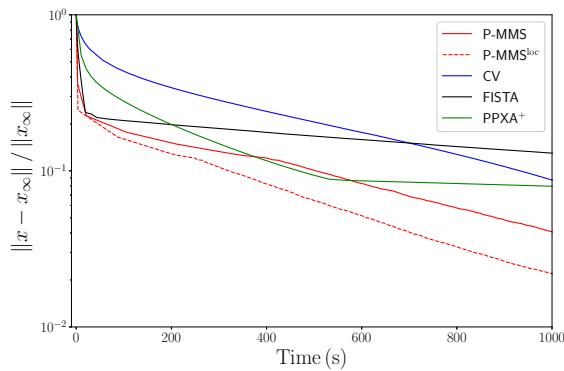


Figure 3.7: Comparison of convergence speed on the PET reconstruction example. Evolution of the distance to the optimum versus time.

3.7 Conclusion

In this chapter, we have proposed a new combined subspace-exterior penalty algorithm for solving constrained, differentiable optimization problems. An interesting feature of our approach is the trust-region technique used to accelerate the algorithm. The proposed method can handle a large variety of constraints and was shown to compare favorably with state-of-the-art algorithms in terms of convergence speed on two large-scale image recovery problems. In future work, it could be interesting to relax the convexity assumption (Assumption 3.8) made in the theoretical analysis of the penalty method, possibly by better relying on the KL property. Another development would consist in establishing a local convergence rate for our algorithm.

3.A Appendix

We give the proof of Proposition 3.23.

Proof. Let $\tau \in \mathbb{R}$. Define, for every $v \in \mathbb{R}$,

$$\phi(v) = \bar{\rho}(v) - \bar{\rho}(\tau) - \bar{\rho}'(\tau)(v - \tau) - \frac{\omega(\tau)}{2}(v - \tau)^2. \quad (3.A.1)$$

We shall prove that ϕ is nonpositive on \mathbb{R} . First, note that our assumptions on ρ imply that ρ' is nondecreasing on $[0, +\infty)$ and ρ'' is positive, nonincreasing on $[0, +\infty)$. We distinguish three cases namely $\tau > 0$, $\tau = 0$, and $\tau < 0$.

(i) Suppose that $\tau > 0$. Differentiating ϕ twice yields the equality

$$(\forall v \in (0, +\infty)) \quad \phi''(v) = \rho''(v) - \frac{\rho'(\tau) - \rho'(0)}{v}. \quad (3.A.2)$$

and $\phi''(0^+) = \rho''(0) - (\rho'(\tau) - \rho'(0))/\tau$. Since ρ' is concave on $[0, +\infty)$ and $\tau > 0$, it follows from the tangent inequality that

$$\phi''(0^+) \geq 0 \quad \text{and} \quad \phi''(\tau) \leq 0. \quad (3.A.3)$$

In addition, ϕ'' is nonincreasing and continuous on $[0, +\infty)$, thus there exists $\zeta \in [0, \tau]$ such that $\phi''(\zeta) = 0$. Hence, ϕ' is nondecreasing on $[0, \zeta]$ and nonincreasing on $[\zeta, +\infty)$. Since $\phi'(0) = \phi'(\tau) = 0$, it follows that

$$(\forall v \in [0, \tau]) \quad \phi'(v) \geq 0, \quad (3.A.4)$$

$$(\forall v \in [\tau, +\infty)) \quad \phi'(v) \leq 0. \quad (3.A.5)$$

Finally, ϕ is nondecreasing on $[0, \tau]$, nonincreasing on $[\tau, +\infty)$, and since $\phi(\tau) = 0$,

$$(\forall v \in [0, +\infty)) \quad \phi(v) \leq 0. \quad (3.A.6)$$

Let us prove the same inequality on $] -\infty, 0[$. We have

$$\forall v \in (-\infty, 0[, \quad \phi''(v) = -\frac{\rho'(\tau) - \rho'(0)}{v}. \quad (3.A.7)$$

Since ρ' is nondecreasing on $[0, +\infty)$ and $\tau > 0$,

$$\forall v \in (-\infty, 0), \quad \phi''(v) \leq 0. \quad (3.A.8)$$

We deduce that ϕ' is nonincreasing on $(-\infty, 0)$. Moreover, since $\phi'(0) = 0$, we obtain by continuity that $\phi'(v) \geq 0$ for all $v \in (-\infty, 0)$. Hence ϕ is nondecreasing on $(-\infty, 0)$ and, according to (3.A.6), $\phi(0) \leq 0$. Thus

$$(\forall v \in (-\infty, 0)) \quad \phi(v) \leq 0, \quad (3.A.9)$$

which concludes the study in the case when $\tau > 0$.

- (ii) Suppose that $\tau = 0$. The derivative of ϕ reads

$$(\forall v \in \mathbb{R}) \quad \phi'(v) = \bar{\rho}'(v) - \bar{\rho}'(0) - \rho''(0)v, \quad (3.A.10)$$

thus by concavity of ρ' on $[0, +\infty)$, for every $v \geq 0$, $\phi'(v) \leq 0$. Moreover, for every $v < 0$,

$$\phi'(v) = -\rho''(0)v \geq 0. \quad (3.A.11)$$

Since $\phi(0) = 0$, we easily conclude that

$$(\forall v \in \mathbb{R}) \quad \phi(v) \leq 0. \quad (3.A.12)$$

- (iii) Suppose that $\tau < 0$. For every $v \in [0, +\infty)$,

$$\phi''(v) = \rho''(v) - \frac{\rho'(-\tau) - \rho'(0)}{-\tau}. \quad (3.A.13)$$

Since ρ' is concave on $[0, +\infty)$,

$$\phi''(0^+) \geq 0 \quad \text{and} \quad \phi''(-\tau) \leq 0. \quad (3.A.14)$$

As for the case $\tau > 0$, there exists $\zeta \in [0, -\tau]$ such that $\phi''(\zeta) = 0$. Hence ϕ' is nondecreasing on $[0, \zeta[$ and nonincreasing on $[\zeta, +\infty)$. Let us show that $\phi'(\zeta) \leq 0$. Using the fact that $\bar{\rho}'(\tau) = \rho'(0)$ (because of (3.6.33) when $\tau < 0$), an immediate calculation yields

$$\begin{aligned} \phi'(\zeta) &= \rho'(\zeta) - \rho'(0) - (\rho'(-\tau) - \rho'(0)) \\ &\quad - \frac{\rho'(-\tau) - \rho'(0)}{-\tau} \zeta. \end{aligned} \quad (3.A.15)$$

Since ρ' is nondecreasing and $\rho''(\zeta) = \frac{\rho'(-\tau) - \rho'(0)}{-\tau}$ by definition of ζ , we have

$$\begin{aligned} \phi'(\zeta) &\leq \rho'(\zeta) - \rho'(0) - (\rho'(\zeta) - \rho'(0)) - \rho''(\zeta)\zeta \\ &= -\rho''(\zeta)\zeta \\ &\leq 0. \end{aligned} \quad (3.A.16)$$

Therefore

$$(\forall v \in [0, +\infty)) \quad \phi'(v) \leq 0, \quad (3.A.17)$$

and ϕ is nonincreasing $[0, +\infty)$.

Now, it suffices to prove that $\phi(0) \leq 0$ to deduce the nonpositivity of ϕ on $[0, +\infty)$. We have

$$\begin{aligned}\phi(0) &= \bar{\rho}(0) - \bar{\rho}(\tau) - \bar{\rho}'(\tau)(-\tau) - \frac{\rho'(-\tau) - \rho'(0)}{-2\tau}\tau^2 \\ &= \rho(0) - (\rho'(0)\tau + \rho(0)) + \rho'(0)\tau \\ &\quad + \frac{1}{2}(\rho'(-\tau) - \rho'(0))\tau \\ &= \frac{1}{2}(\rho'(-\tau) - \rho'(0))\tau \\ &\leq 0,\end{aligned}\tag{3.A.18}$$

where the last inequality follows from the fact that ρ' is nondecreasing on $]-\infty, 0]$. Hence

$$(\forall v \in [0, +\infty)) \quad \phi(v) \leq \phi(0) \leq 0.\tag{3.A.19}$$

To prove a similar inequality on $(-\infty, 0)$, we observe that

$$(\forall v \in (-\infty, 0)) \quad \phi''(v) = -\omega(\tau) \leq 0.\tag{3.A.20}$$

Thus ϕ' is nonincreasing on $(-\infty, 0)$. Since $\phi'(\tau) = 0$, we deduce that ϕ is nondecreasing on $(-\infty, \tau]$ and nonincreasing on $[\tau, 0)$. It follows that

$$(\forall v \in (-\infty, 0)) \quad \phi(v) \leq \phi(\tau) = 0.\tag{3.A.21}$$

Finally, gathering inequalities (3.A.19) and (3.A.21) yields

$$(\forall v \in \mathbb{R}) \quad \phi(v) \leq 0,\tag{3.A.22}$$

which concludes the proof.

□

- Chapter 4 -

A novel variational approach for multiphoton microscopy image restoration: from PSF estimation to 3D deconvolution

This chapter is based on the following article:

J. Ajdenbaum, E. Chouzenoux, S. Martin, C. Lefort, J.-C. Pesquet, A novel variational approach for multiphoton microscopy image restoration: from PSF estimation to 3D deconvolution, in *hal-*, 2023.

In multi-photon microscopy (MPM), a recent *in-vivo* fluorescence microscopy system, the task of image restoration can be decomposed into two interlinked inverse problems: firstly, the characterization of the Point Spread Function (PSF) and subsequently, the deconvolution (i.e., deblurring) to remove the PSF effect, and reduce noise.

The acquired MPM image quality is critically affected by PSF blurring and intense noise. The PSF in MPM is highly spread in 3D and is not well characterized, presenting high variability with respect to the observed objects. This makes the restoration of MPM images challenging. Common PSF estimation methods in fluorescence microscopy, including MPM, involve capturing images of sub-resolution beads, followed by quantifying the resulting ellipsoidal 3D spot. In this work, we revisit this approach, coping with its inherent limitations in terms of accuracy and practicality. We estimate the PSF from the observation of relatively large beads (approximately $1\mu\text{m}$ in diameter). This goes through the formulation and resolution of an original non-convex minimization problem, for which we propose a proximal alternating method along with convergence guarantees.

Following the PSF estimation step, we then introduce an innovative strategy to deal with the high level multiplicative noise degrading the acquisitions. We rely on a heteroscedastic noise model for which we estimate the parameters. We then solve a constrained optimization problem to restore the image, accounting for the estimated PSF and noise, while allowing a minimal hyper-parameter tuning. Theoretical guarantees are given for the restoration algorithm.

These algorithmic contributions lead to an end-to-end pipeline for 3D image restoration in MPM, that we share as a publicly available Python software. We demonstrate its effectiveness through several experiments on both simulated and real data.

4.1 Introduction

Multi-photon microscopy (MPM) is a non-invasive laser imaging technique that selectively induces fluorescence in a thin plane through localized nonlinear excitation while avoiding excitation elsewhere [Diaspro et al., 2006]. This technique enables three-dimensional imaging with infrared light, reaching depths that are at least double compared to traditional single-photon microscopy [Göbel et al., 2007, Centonze and White, 1998, Larson, 2011]. Particularly, MPM contactless and non-invasive nature makes it suitable for imaging diverse objects, spanning from living organisms [Zipfel et al., 2003] to materials [Jonard et al., 2022].

Though MPM offers significant advantages, its widespread adoption remains limited due to degradations affecting the images, especially blur and noise, requiring innovative computational solutions to unlock its full potential. Central to MPM imaging is the Point Spread Function (PSF). This function describes how the imaging system responds to a Dirac impulse, in practice a single-point light object. In practical terms, this response appears as an extended spot in the image around the original object location, which is responsible for distortions and blurry aspects [Young et al., 2011a, Young et al., 2011b, Descloux et al., 2016], particularly in the depth direction. As MPM typically operates as a non-coherent and space-invariant system, it is commonplace to model the captured observation as a convolution of the actual object with a blur kernel. In the following, we will use the terms *PSF* and *blur kernel* interchangeably as they refer to the same mathematical object. A major challenge with the PSF in MPM is its strong dependency on the medium [Dong et al., 2003], implying that it should be individually estimated for each new acquisition.

In MPM, the restoration process aims to recover the sought (non-degraded) object of interest, $\bar{x}: \mathbb{R}^3 \rightarrow \mathbb{R}$, jointly with an estimation of the Point Spread Function (PSF), $\bar{h}: \mathbb{R}^3 \rightarrow \mathbb{R}$, guided by the following linear observational model, at each location \mathbf{u} of the 3D volume:

$$(\forall \mathbf{u} \in \mathbb{R}^3) \quad y(\mathbf{u}) = \mathcal{D}(\bar{h} \star \bar{x} + \alpha)(\mathbf{u}). \quad (4.1.1)$$

In (4.1.1), $y: \mathbb{R}^3 \rightarrow \mathbb{R}$ stands for the observation, $\alpha \in \mathbb{R}$ represents a background factor, and \star the continuous 3D convolution operation, here to be assumed to be defined. Additionally, \mathcal{D} refers to the noise model. In the computational imaging literature, there are two main ways of addressing the above inverse problem. Some methods directly tackle the *blind* inverse problem represented by (4.1.1), i.e. they simultaneously estimate both \bar{x} and \bar{h} [Vorontsov and Jefferies, 2017, Prato et al., 2013, Huang et al., 2022, Debarnot and Weiss, 2021]. However, this approach can be computationally demanding due to its non-convex nature and potential underdetermined nature. Another strategy is to break down the problem into two successive inverse problems. In this two-step approach, one first performs a calibration step [Krist et al., 2011, Högbom, 1974, Dusch et al., 2007, Huang et al., 2021, Samuylov et al., 2019], providing an estimate of the PSF \bar{h} in a controlled situation where \bar{x} is a known (i.e., calibrated) entity. Then, utilizing the estimated PSF, one can infer an estimate of another, more complex, unknown object \bar{x} .

PSF calibration in MPM typically involves imaging sub-resolution fluorescent beads in a homogeneous medium [Cole et al., 2011, Doi et al., 2018] or directly in the sample (*in situ*) [Lefort et al., 2021, Doi et al.,

2018, Von Tiedemann et al., 2006]. The image of such a bead captured with the MPM device appears as a spread spot, which is then fit with a shape function (e.g., Gaussian) to estimate the PSF characteristics [Von Tiedemann et al., 2006, Zhang et al., 2007, Guo, 2011, Chouzenoux et al., 2019b]. However, this method presents significant practical limitations. Firstly, handling beads as small as $0.2\mu\text{m}$ in diameter requires special care, as sub-micrometric objects are difficult to even observe with the microscope. As a result, one often injects hundreds of beads into the sample, often leading to beads aggregation that compromises the accurate completion of the PSF calibration protocol. Secondly, observing the impulse response of the instrument on such tiny beads requires the use of high laser power, which can potentially damage the sample itself. These limitations highlight the need for an alternative and a more practical protocol for PSF estimation in MPM.

Following PSF estimation, the subsequent task is image deblurring and denoising. Over the years, several computational solutions have been proposed to solve this inverse problem [Dao et al., 2015, Ströhl and Kaminski, 2015, Sarder and Nehorai, 2006] in the context of microscopy. A common strategy is to minimize a cost function which balances a data-fitting term with a regularization term [Difato et al., 2004, De Monvel et al., 2003, Dey et al., 2006]. Yet, such an approach often suffers from a time-intensive parameter tuning phase, made even more challenging by subjective decision-making in the absence of a clear ground-truth. Further difficulties emerge from consideration on the nature of the noise in MPM. While most studies assume a standard Gaussian additive noise [Doi et al., 2018, Danielyan et al., 2014], the noise in MPM images is more appropriately considered as multiplicative, as pointed out in [Crivaro et al., 2011, Niu et al., 2022]. However, such a noise model can be tedious to deal with numerically at large-scale, often involving minimizing a non-smooth data-fidelity term [Zanella et al., 2009, Chouzenoux et al., 2015]. While this issue has been tackled in confocal microscopy [Vicidomini et al., 2009, Zanella et al., 2009, Bohra et al., 2019], its exploration in MPM imaging remains scarce.

In this work, we present a novel approach for addressing the MPM image restoration inverse problem in an end-to-end fashion. Our comprehensive restoration pipeline revisits the conventional restoration protocol from acquisition to the final restored outcome. Within this pipeline, we introduce two major advancements:

1. PSF Calibration Step. Our contributions regarding the estimation of the instrumental PSF are two-fold.

- We introduce a novel optimization problem formulation for estimating the a Gaussian-shaped PSF assuming large diameter beads as opposed to sub-resolution beads.
- Next, we address the PSF estimation inverse problem by deploying an alternating proximal approach whose convergence is proved.

2. Image restoration Step. Our contributions to the image restoration phase encompass two key elements.

- First, we adopt an additive heteroscedastic noise model that approximates the Poisson-Gaussian noise. We then introduce a quantization-based image analysis method to estimate the noise model parameters.

- Additionally, we address the parameter tuning issue raised in penalized restoration approaches, by introducing an original constrained formulation of the image restoration inverse problem. Our formulation introduces an interpretable upper bound on the data fidelity term, making the restoration stage stable, reliable, and parameter-free. The problem resolution is then addressed with the local subspace Majorization-Minimization algorithm proposed in [Chouzenoux et al., 2022], for which we show the convergence in our context.

We demonstrate the effectiveness of our two-step restoration pipeline on both simulated and real-world data acquired with an Olympus XLPLN25XWMP two-photon microscope. The associated Python code is publicly shared¹, ensuring reproducible research.

The outline is as follows. Section 4.2 introduces our PSF estimation method, and validation experiments on synthetic data. In Section 4.3, we introduce our noise modeling and image restoration approaches, and we assess them on synthetic datasets. Lastly, in Section 4.4, we present the experimental evaluation of our complete pipeline to the imaging of real mouse muscle data.

4.2 A novel approach to PSF calibration using large beads

In this section, we introduce our novel approach for estimating the PSF from acquisition of calibrated fluorescent beads with a diameter of $1\mu\text{m}$, larger than the microscope resolution.

4.2.1 Preparing the data

For performing the calibration stage, multiple fluorescent beads with known size, are observed under the microscope in one comprehensive 3D image. The PSF estimation aims at fitting a model on each individual bead. To do so, the initial large image is cropped, to form several volumes of interest each containing an isolated bead. The cropping process we propose consists in applying a standard Wiener filtering to reduce noise, followed by a binarization using a predefined threshold (in practice, a given pourcentage of the maximum intensity), and an automatic connected component labeling algorithm [Wu et al., 2005]. Each selected component then defines a cropped region, to which we apply the PSF estimation method. Under the assumption of stationarity of the PSF model within the observed volume, the final PSF estimate can be obtained by averaging the models fitted in each volume interest, to get a more accurate overall estimation. We now explicit our PSF estimation process, given a volume containing a single bead. An example of image of a polystyrene bead with $1\mu\text{m}$ diameter (FluoSphere™ Carboxylate-Modified Microsphere, manufactured by ThermoFisher, product number: F8823), acquired by our Olympus XLPLN25XWMP two-photon microscope is displayed in Figure 4.1.

4.2.2 Modeling of the problem

Since we are in a calibration context, the true bead characteristics are known. In particular, the experimenter selects the bead diameter, and a spherical shape can be assumed. Moreover, the position of the bead can be easily deduced by a basic identification of the centroid of the bead image. This allows us

¹<https://github.com/SegoleneMartin/biphoton>

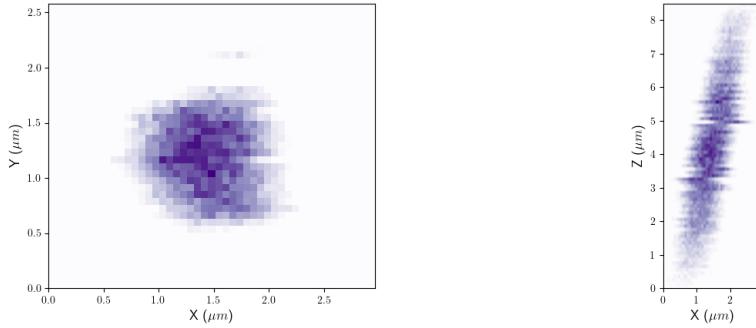


Figure 4.1: Two-photon acquisition of a polystyrene bead of $1\mu\text{m}$ of diameter. XY plane section (left) and XZ plane section (right). The acquisition was carried out with an excitation wavelength of 810nm . The voxel size was fixed to $0.049 \times 0.049 \times 0.1\mu\text{m}^3$.

to build a complete model for the observed object, up to a potential multiplicative factor in the pixel intensities.

For the remainder of the chapter, let us introduce the following notations: $\|\cdot\|$ represents the Euclidean norm on \mathbb{R}^N , $\mathbf{1}_N$ the unit vector of \mathbb{R}^N , $\|\cdot\|_F$ the Frobenius norm on $\mathbb{R}^{N \times N}$, $2^{\mathcal{C}}$ the set of all subsets of a given set \mathcal{C} , \mathcal{S}_3 the set of symmetric matrices in $\mathbb{R}^{3 \times 3}$, \mathcal{S}_3^+ the set of symmetric positive semi-definite matrices in $\mathbb{R}^{3 \times 3}$, \mathcal{S}_3^{++} the set of symmetric positive definite matrices in $\mathbb{R}^{3 \times 3}$, and \mathbf{I}_3 the identity matrix of \mathbb{R}^3 .

Continuous Modeling

Let $x: \mathbb{R}^3 \rightarrow \mathbb{R}$ represent a sphere with diameter τ , modeling the bead fluorescence. The bead is assumed to emit with a uniform intensity, so we set, for all $\mathbf{u} \in \mathbb{R}^3$, $x(\mathbf{u}) = 1$ if $\|\mathbf{u}\| \leq \tau$, and $x(\mathbf{u}) = 0$ otherwise. The ground truth fluorescence intensity is then scaled by a factor $\bar{\beta}_c \in (0, +\infty)$, which is usually unknown.

Let $y: \mathbb{R}^3 \rightarrow \mathbb{R}$ be the image of the bead obtained with the MPM device. The observation model takes the form of (4.1.1). For the sake of simplicity, in this calibration step, we adopt an assumption of an additive i.i.d. Gaussian noise. Thus, we have:

$$(\forall \mathbf{u} \in \mathbb{R}^3) \quad y(\mathbf{u}) = \bar{\alpha} + \bar{\beta}_c (\bar{h} * x)(\mathbf{u}) + \nu(\mathbf{u}), \quad (4.2.1)$$

where ν is a function accounting for the Gaussian noise, $\bar{\alpha} \in \mathbb{R}$ is a background term, and $\bar{h} \in L^1(\mathbb{R}^3)$ is a convolution kernel (i.e., the PSF) such that $\bar{h}(\mathbf{u}) \geq 0$ almost everywhere on \mathbb{R}^3 and satisfying

$$\int_{\mathbb{R}^3} \bar{h}(\mathbf{u}) d\mathbf{u} = 1. \quad (4.2.2)$$

Given the observation model (4.2.1), the goal is to estimate the unknowns $\bar{\alpha}$, $\bar{\beta}_c$, and \bar{h} .

Discrete Modeling

MPM acquisitions are done point-by-point, on a voxel grid with a given resolution. Hence, one only has access to a sampling of y on a bounded discrete set Ω of \mathbb{R}^3 , paved into $N \in \mathbb{N}$ voxels with mass centers $(\omega_n)_{1 \leq n \leq N} \in (\mathbb{R}^3)^N$. The discretized version of the continuous problem (4.2.1) then becomes:

$$\mathbf{y} = \bar{\alpha}\mathbf{1}_N + \bar{\beta}(\bar{\mathbf{h}} * \mathbf{x}) + \boldsymbol{\nu}, \quad (4.2.3)$$

where $\mathbf{y} = (y(\omega_n))_{1 \leq n \leq N} \in \mathbb{R}^N$ (resp. $\mathbf{x} = (x(\omega_n))_{1 \leq n \leq N} \in \mathbb{R}^N$ and $\boldsymbol{\nu} = (\nu(\omega_n))_{1 \leq n \leq N} \in \mathbb{R}^N$) is the discretization of y (resp. x and ν) on Ω , $\bar{\mathbf{h}} = (\bar{h}_n)_{1 \leq n \leq N} \in \mathbb{R}^N$ is a discrete convolution kernel satisfying $\bar{\mathbf{h}} \in \Delta_N$ where Δ_N is the simplex of \mathbb{R}^N defined as

$$\Delta_N = \left\{ \mathbf{h} = (h_n)_{1 \leq n \leq N} \in \mathbb{R}^N \mid (\forall n \in \{1, \dots, N\}) h_n \geq 0 \text{ and } \sum_{n=1}^N h_n = 1 \right\}, \quad (4.2.4)$$

and $*$ states for the discrete 3D convolution operator with appropriate padding (typically, circular padding). In addition, $\bar{\beta} > 0$ is a scaling factor for the discrete model, which differs from the one in the continuous model, $\bar{\beta}_c$. Indeed, observe that $\bar{\mathbf{h}}$ is not, per se, a discretization of \mathbf{h} . In particular, an implicit rescaling is assumed to ensure the sum of $\bar{\mathbf{h}}$ equals 1, for simplicity. The goal is thus to provide estimates of $\bar{\mathbf{h}}$, and the shift/scaling factors $(\bar{\alpha}, \bar{\beta})$, given the knowledge of the measurements \mathbf{y} and the bead model \mathbf{x} .

4.2.3 Proposed minimization problem

Following [Chouzenoux et al., 2019b], we introduce a prior so as to promote a PSF with a Gaussian shape. This leads to solving the following regularized variational problem:

$$\underset{\substack{\alpha \in \mathcal{A}, \beta \in \mathcal{B}, \\ \mathbf{h} \in \Delta_N, \mathbf{D} \in \mathcal{S}_3^+}}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{y} - \alpha\mathbf{1}_N - \beta(\bar{\mathbf{h}} * \mathbf{x})\|^2 + \lambda \mathcal{KL}(\mathbf{h} \| \zeta \mathbf{g}(\mathbf{D} + \epsilon_1 \mathbf{I}_3)) + \epsilon_2 \|\mathbf{D}\|_F^2. \quad (4.2.5)$$

Hereabove, $\epsilon_1 > 0$, so for every $\mathbf{D} \in \mathcal{S}_3^+$, matrix $\mathbf{D} + \epsilon_1 \mathbf{I}_3 \in \mathcal{S}_3^{++}$, and $\mathbf{g}(\mathbf{D} + \epsilon_1 \mathbf{I}_3) \in \mathbb{R}^N$ states for the discretization on Ω of the centered Gaussian density function with inverse covariance $\mathbf{D} + \epsilon_1 \mathbf{I}_3$, corresponding to

$$(\forall \mathbf{S} \in \mathcal{S}_3^{++}) (\forall n \in \{1, \dots, N\}) \quad [\mathbf{g}(\mathbf{S})]_n = \sqrt{\frac{|\mathbf{S}|}{(2\pi)^3}} \exp\left(-\frac{1}{2} \omega_n^\top \mathbf{S} \omega_n\right). \quad (4.2.6)$$

Moreover, $\lambda > 0$ is a regularization parameter, and $\zeta \in (0, +\infty)$ is a known scaling parameter accounting for the measure of the discrete Gaussian function $\mathbf{g}(\mathbf{D} + \epsilon_1 \mathbf{I}_3)$ on the grid Ω . When the grid is sufficiently fine, which we will assume throughout this discussion, ζ can be approximated by

$$\zeta \simeq r_X r_Y r_Z, \quad (4.2.7)$$

where $r = (r_X, r_Y, r_Z) \in (0, +\infty)^3$ represents the dimensions, in micrometers, of the voxels in the captured image (i.e., the image resolution). Sets $\mathcal{A} = [\alpha_-, \alpha_+]$ and $\mathcal{B} = [\beta_-, \beta_+]$ are real intervals corresponding to known bounds on α and β , respectively. \mathcal{KL} denotes the Kullback-Leibler divergence

between two discrete probability distributions, defined as

$$(\forall (\mathbf{p}, \mathbf{q}) \in (\Delta_N \cap (0, +\infty))^2) \quad \mathcal{KL}(\mathbf{p} \| \mathbf{q}) = \sum_{n=1}^N p_n \log \left(\frac{p_n}{q_n} \right). \quad (4.2.8)$$

Finally $\epsilon_2 \in (0, +\infty)$ is an arbitrarily small penalty parameter.

Let us denote, for conciseness:

$$(\forall \mathbf{h} \in \Delta_N) (\forall \mathbf{D} \in \mathcal{S}_3^+) \quad \Psi(\mathbf{h}, \mathbf{D}) = \mathcal{KL}(\mathbf{h} \| \zeta \mathbf{g}(\mathbf{D} + \epsilon_1 \mathbf{I}_3)). \quad (4.2.9)$$

To avoid potential numerical issues with the logarithmic term in \mathcal{KL} divergence, we extend Ψ into a twice continuously differentiable function $\tilde{\Psi}$ on $\Delta_N \times \mathcal{S}_3$, defined as

$$(\forall \mathbf{h} \in \Delta_N) (\forall \mathbf{D} \in \mathcal{S}_3) \quad \tilde{\Psi}(\mathbf{h}, \mathbf{D}) = \sum_{n=1}^N \left(E(h_n) - h_n \ln \zeta \right. \\ \left. + \frac{h_n}{2} (3 \ln(2\pi) + \Phi(\mathbf{D}) + \boldsymbol{\omega}_n^\top (\mathbf{D} + \epsilon_1 \mathbf{I}_3) \boldsymbol{\omega}_n) \right), \quad (4.2.10)$$

where, for any $\mathbf{D} \in \mathcal{S}_3$ diagonalized as $\mathbf{D} = \mathbf{U} \text{Diag}(\mathbf{s}) \mathbf{U}^\top$ with \mathbf{U} an orthogonal matrix of $\mathbb{R}^{3 \times 3}$ and $\mathbf{s} \in \mathbb{R}^3$,

$$\Phi(\mathbf{D}) = \sum_{i=1}^3 \phi(s_i), \quad (4.2.11)$$

with ϕ defined as

$$(\forall t \in \mathbb{R}) \quad \phi(t) = \begin{cases} -\ln(t + \epsilon_1) & \text{if } t \geq 0, \\ -\ln(\epsilon_1) - \epsilon_1^{-1}t + \epsilon_1^{-2}t^2 & \text{otherwise,} \end{cases} \quad (4.2.12)$$

and

$$(\forall u \in \mathbb{R}) \quad E(u) = \begin{cases} u \ln(u) & \text{if } u > 0, \\ 0 & \text{if } u = 0, \\ +\infty & \text{otherwise.} \end{cases} \quad (4.2.13)$$

In order to model the constraints in Problem (4.2.5), we introduce, for any non-empty, closed, convex set $C \subseteq \mathbb{R}^n$, its indicator function ι_C , namely $\iota_C(x) = 0$ if $x \in C$, $\iota_C(x) = +\infty$ otherwise. In a nutshell, Problem (4.2.5) is equivalent to minimize, for $(\alpha, \beta, \mathbf{h}, \mathbf{D}) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3$,

$$F(\alpha, \beta, \mathbf{h}, \mathbf{D}) = \frac{1}{2} \|\mathbf{y} - \alpha \mathbf{1}_N - \beta(\mathbf{h} * \mathbf{x})\|^2 + \lambda \tilde{\Psi}(\mathbf{h}, \mathbf{D}) \\ + \iota_A(\alpha) + \iota_B(\beta) + \iota_{\Delta_N}(\mathbf{h}) + \iota_{\mathcal{S}_3^+}(\mathbf{D} + \epsilon_1 \mathbf{I}_3) + \epsilon_2 \|\mathbf{D}\|_F^2. \quad (4.2.14)$$

It will be established in Proposition 4.5 that, since $\epsilon_2 \in (0, +\infty)$, F admits a minimizer.

4.2.4 Minimization algorithm

The objective function in (4.2.14) is nonconvex. However, it is convex with respect to each variable. A common strategy for such scenarios consists of using alternating minimization techniques. At each iteration, we minimize F with respect to a variable while the other variables are held constant. This

simple method, also known as Block Coordinate Descent or the nonlinear Gauss-Seidel method, has previously been utilized for PSF model fitting in microscopy, for instance in [Fortun et al., 2018, Li et al., 2018]. Yet, closed-form updates are not always available. Moreover, convergence to an optimal solution using this method is only guaranteed when the partial functions on each variable and at each iteration are uniformly strongly convex [Luo and Tseng, 1993, Beck and Tetruashvili, 2013]. This stringent assumption is not met here. To ensure both efficiency and guarantees of convergence, a strategy enriched with proximal tools is preferable, as highlighted for instance in [Chouzenoux et al., 2019b, Chouzenoux et al., 2016, Bonettini et al., 2018]. The particular structure of (4.2.14) suggests a hybrid proximal alternating scheme ([Xu and Yin, 2013, Phan and Gillis, 2023, Chouzenoux et al., 2023]) that we present hereafter.

4.2.4.1 Proposed algorithm

Guided by the methodology in [Xu and Yin, 2013] and [Phan and Gillis, 2023], we opt for an alternating minimization algorithm acting on the variables α , β , \mathbf{h} , and \mathbf{D} . At each iteration of the algorithm, we update a specific variable employing one of the three following distinct schemes applied to the partial function with respect to this variable: an exact update step, a proximal step, or a proximal linearized step (also called forward-backward, or proximal gradient).

Within our framework, the variables α and β undergo exact step updates, \mathbf{D} is updated through a proximal point step, and \mathbf{h} is updated through a proximal linearized step. We provide the expressions for these updates in the section that follows. Our algorithm, called GENTLE (Gaussian kErNel fiTting using Large BEads), is summarized in Algorithm 10, where we use the short notation

$$(\forall(\alpha, \beta, \mathbf{h}, \mathbf{D}) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3) \quad \tilde{F}(\alpha, \beta, \mathbf{h}, \mathbf{D}) = \lambda \tilde{\Psi}(\mathbf{h}, \mathbf{D}) + \iota_{\Delta_N}(\mathbf{h}), \quad (4.2.15)$$

and the stepsizes $(\gamma_h, \gamma_D) \in (0, +\infty)^2$. In addition, we denote $\mathbf{X} \in \mathbb{R}^{N \times N}$ the block Toeplitz matrix such that, for all $(\mathbf{u}, \mathbf{x}) \in (\mathbb{R}^N)^2$, $\mathbf{X}\mathbf{u} = \mathbf{x} * \mathbf{u}$.

Algorithm 10: GENTLE: Gaussian kErNel fiTting using Large BEads

Inputs: Let $(\alpha^{(0)}, \beta^{(0)}, \mathbf{h}^{(0)}, \mathbf{D}^{(0)}) \in \mathcal{A} \times \mathcal{B} \times \Delta_N \times \mathcal{S}_3^+, (\gamma_h, \gamma_D) \in (0, +\infty)^2$.

for $\ell = 0, 1, \dots$ **do**

$$\left| \begin{array}{l} \alpha^{(\ell+1)} = \underset{\alpha \in \mathbb{R}}{\operatorname{argmin}} F(\alpha, \beta^{(\ell)}, \mathbf{h}^{(\ell)}, \mathbf{D}^{(\ell)}) \\ \beta^{(\ell+1)} = \underset{\beta \in \mathbb{R}}{\operatorname{argmin}} F(\alpha^{(\ell+1)}, \beta, \mathbf{h}^{(\ell)}, \mathbf{D}^{(\ell)}) \\ \mathbf{h}^{(\ell+1)} = \operatorname{prox}_{\gamma_h \tilde{F}} \left(\mathbf{h}^{(\ell)} - \gamma_h \beta^{(\ell+1)} \mathbf{X}^\top (\mathbf{y} - \alpha^{(\ell+1)} \mathbf{1}_N - \beta^{(\ell+1)} \mathbf{X} \mathbf{h}^{(\ell)}) \right) \\ \mathbf{D}^{(\ell+1)} = \operatorname{prox}_{\gamma_D F(\alpha^{(\ell+1)}, \beta^{(\ell+1)}, \mathbf{h}^{(\ell+1)}, \cdot)}(\mathbf{D}^{(\ell)}) \end{array} \right.$$

Return: $\alpha, \beta, \mathbf{h}, \mathbf{D}$.

4.2.4.2 Expressions of the updates

We now explicit the expressions for the four updates involved at each iteration of GENTLE.

Proposition 4.1. (Update on α) Let $(\beta, \mathbf{h}, \mathbf{D}) \in \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3$. Then the minimizer of $F(\cdot, \beta, \mathbf{h}, \mathbf{D})$ at α

is given by

$$\alpha = \text{proj}_{\mathcal{A}} \left(\frac{1}{N} (\mathbf{y} - \beta(\mathbf{h} * \mathbf{x}))^\top \mathbf{1}_N \right), \quad (4.2.16)$$

where $\text{proj}_{\mathcal{A}}$ is the orthogonal projection on the closed convex set \mathcal{A} , reading $\text{proj}_{\mathcal{A}}(\alpha) = \max(\alpha_-, \min(\alpha, \alpha_+))$.

Proof. Let $f = F(\cdot, \beta, \mathbf{h}, \mathbf{D})$. Then, for any $\alpha \in \mathbb{R}$,

$$\begin{aligned} 0 \in \partial f(\alpha) &\iff (\mathbf{y} - \beta(\mathbf{h} * \mathbf{x}))^\top \mathbf{1}_N - \alpha N \in \partial \iota_{\mathcal{A}}(\alpha) \\ &\iff \frac{1}{N} (\mathbf{y} - \beta(\mathbf{h} * \mathbf{x}))^\top \mathbf{1}_N - \alpha \in \partial \iota_{\mathcal{A}}(\alpha) \\ &\iff \alpha = \text{proj} \left(\frac{1}{N} (\mathbf{y} - \beta(\mathbf{h} * \mathbf{x}))^\top \mathbf{1}_N \right), \end{aligned} \quad (4.2.17)$$

where we used Proposition 2.16 and Proposition 2.17. \square

Proposition 4.2. (Update on β) Let $(\alpha, \mathbf{h}, \mathbf{D}) \in \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3$, and $\gamma_\beta \in (0, +\infty)$. Then the minimizer of $F(\alpha, \cdot, \mathbf{h}, \mathbf{D})$ is given by

$$\beta = \text{proj}_{\mathcal{B}} \left(\frac{(\mathbf{y} - \alpha \mathbf{1}_N)^\top (\mathbf{h} * \mathbf{x})}{\|\mathbf{h} * \mathbf{x}\|^2} \right). \quad (4.2.18)$$

where $\text{proj}_{\mathcal{B}}$ is the orthogonal projection on the closed convex set \mathcal{B} , reading $\text{proj}_{\mathcal{B}}(\beta) = \max(\beta_-, \min(\beta, \beta_+))$.

We skip the proof which is similar to the previous one.

Proposition 4.3. (Update on \mathbf{h}) Let $(\alpha, \beta, \mathbf{h}', \mathbf{D}) \in \mathbb{R}^2 \times \mathbb{R}^N \times \mathcal{S}_3$, and $\gamma_h \in (0, +\infty)$. Then the proximity operator of $\gamma_h \tilde{F}(\alpha, \beta, \cdot, \mathbf{D})$ at \mathbf{h}' is given by

$$\text{prox}_{\gamma_h \tilde{F}(\alpha, \beta, \cdot, \mathbf{D})}(\mathbf{h}') = (\rho^{-1} W(\rho \exp(w_n(\hat{\mu})))_{1 \leq n \leq N}, \quad (4.2.19)$$

where W denotes the Lambert-W function [Corless et al., 1996],

$$\rho = \frac{1}{\lambda \gamma_h}, \quad (4.2.20)$$

and, for every $n \in \{1, \dots, N\}$, w_n is the function defined as

$$(\forall \mu \in \mathbb{R}) \quad w_n(\mu) = -1 - c_n + \rho(h'_n - \mu) + \ln \zeta, \quad (4.2.21)$$

with

$$c_n = \frac{1}{2} (3 \log(2\pi) + \Phi(D) + \boldsymbol{\omega}_n^\top (\mathbf{D} + \epsilon_1 \mathbf{I}_3) \boldsymbol{\omega}_n). \quad (4.2.22)$$

Moreover, $\hat{\mu} \in \mathbb{R}$ is the unique zero of the function

$$(\forall \mu \in \mathbb{R}) \quad \kappa(\mu) = \rho^{-1} \sum_{n=1}^N W(\rho \exp(w_n(\mu))) - 1. \quad (4.2.23)$$

Proof. The function to minimize is $\tilde{F}(\alpha, \beta, \zeta, \cdot, \mathbf{D}) + \frac{1}{2\gamma_h} \|\cdot - \mathbf{h}'\|^2$. It is equivalent to minimizing the function f defined by

$$(\forall \mathbf{h} \in (0, +\infty)^N) \quad f(\mathbf{h}) = \lambda \gamma_h \sum_{n=1}^N (h_n \ln h_n - h_n \ln \zeta + c_n h_n) + \iota_{\Delta_N}(\mathbf{h}) + \frac{1}{2} \|\mathbf{h} - \mathbf{h}'\|^2. \quad (4.2.24)$$

The Lagrangian function associated with the minimization of f reads

$$(\forall \mathbf{h} \in (0, +\infty)^N) (\forall \mu \in \mathbb{R}) \quad \mathcal{L}(\mathbf{h}, \mu) = \lambda \gamma_h \sum_{n=1}^N \left(h_n \ln h_n + h_n (-\ln \zeta + c_n) + \frac{1}{2\lambda \gamma_h} (h_n - h'_n)^2 \right) + \mu \left(\sum_{n=1}^N h_n - 1 \right). \quad (4.2.25)$$

Since Slater's condition obviously holds, there exists $\hat{\mu} \in \mathbb{R}$ such that $(\mathbf{h}, \hat{\mu})$ is a saddle point of \mathcal{L} [Bertsekas, 1995]. By Fermat's rule [Bauschke and Combettes, 2019], \mathbf{h} is thus obtained by finding a zero of the derivative of $\mathcal{L}(\cdot, \hat{\mu})$. This yields, for every $n \in \{1, \dots, N\}$,

$$\begin{aligned} & \lambda \gamma_h \left(1 + \ln \hat{h}_n + c_n - \ln \zeta \right) + \hat{h}_n - h'_n + \hat{\mu} = 0, \\ \iff & \ln(\hat{h}_n) + \rho \hat{h}_n = w_n(\hat{\mu}), \\ \iff & \rho \hat{h}_n \exp(\rho \hat{h}_n) = \rho \exp(w_n(\hat{\mu})), \end{aligned} \quad (4.2.26)$$

where ρ is given by (4.2.20) and $w_n(\hat{\mu}) = -1 + \rho(h'_n - \hat{\mu}) - c_n + \ln \zeta$.

Finally, recalling that the W -Lambert function is such that $(\forall z \in \mathbb{R}) W(z) \exp(W(z)) = z$, we deduce that

$$\hat{h}_n = \rho^{-1} W(\rho \exp(w_n(\hat{\mu}))). \quad (4.2.27)$$

In addition, one can obtain $\hat{\mu}$ from the linear equality constraint. This amounts to finding a zero of the function κ defined as

$$(\forall \mu \in \mathbb{R}) \quad \kappa(\mu) = \rho^{-1} \sum_{n=1}^N W(\rho \exp(w_n(\mu))) - 1. \quad (4.2.28)$$

It was shown in [Chouzenoux et al., 2019b], relying on the properties of the W -Lambert function, that κ admits a unique zero. \square

The Lambert-W function appearing in Proposition 4.3 is commonly in the expression of the proximity operators of entropic functions [Lapin et al., 2017, Cherni et al., 2016]. While its evaluation, requiring the solution of a transcendental equation, can be efficiently achieved using a Newton-based method, the composition of W with the exponential can lead to arithmetic overflow for large inputs. To address this, we employ the asymptotic expansion $W(\exp(u)) \approx u - \log(u)$ for $u > 10^2$ [Corless et al., 1996].

Proposition 4.4. (Update on \mathbf{D}) *Let $(\alpha, \beta, \mathbf{h}, \mathbf{D}) \in \mathbb{R}^2 \times (0, +\infty)^N \times \mathcal{S}_3$, and $\gamma_D \in (0, +\infty)$. Then the proximity operator of $\gamma_D F(\alpha, \beta, \mathbf{h}, \cdot)$ at \mathbf{D} is given by*

$$\text{prox}_{\gamma_D F(\alpha, \beta, \mathbf{h}, \cdot)}(\mathbf{D}) = \frac{1}{2} \mathbf{V} \text{Diag} \left(\left(\max(\mu_i - \epsilon_1 + \sqrt{(\mu_i + \epsilon_1)^2 + 4m}, 0) \right)_{1 \leq i \leq d} \right) \mathbf{V}^\top, \quad (4.2.29)$$

where $\boldsymbol{\mu} = (\mu_i)_{1 \leq i \leq 3}$ is a vector of eigenvalues of $(2\epsilon_2 \gamma_D + 1)^{-1} \mathbf{D} - \mathbf{S}$ and \mathbf{V} is a 3×3 orthogonal matrix such that $(2\epsilon_2 \gamma_D + 1)^{-1} \mathbf{D} - \mathbf{S} = \mathbf{V} \text{Diag}(\boldsymbol{\mu}) \mathbf{V}^\top$ with

$$\mathbf{S} = \frac{\gamma_D}{2(2\epsilon_2 \gamma_D + 1)} \lambda \sum_{n=1}^N h_n \boldsymbol{\omega}_n \boldsymbol{\omega}_n^\top, \quad (4.2.30)$$

and $m = \frac{1}{2} \gamma_D \lambda$.

Proof. Direct extension of [Chouzenoux et al., 2019b, Prop.4]. \square

4.2.4.3 Convergence analysis

We now establish the convergence of GENTLE. We first show that the considered minimization problem has at least one solution.

Proposition 4.5. *The cost function F defined in (4.2.14) is lower-bounded and admits a minimizer.*

Proof. It is clear that F is lower semi-continuous on $\mathbb{R} \times \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3$. Let us now show that F is coercive. Since \mathcal{A} , \mathcal{B} , and Δ_N are bounded subsets, it suffices to show that, for any $(\alpha, \beta, \mathbf{h}) \in \mathcal{A} \times \mathcal{B} \times \Delta_N$,

$$F(\alpha, \beta, \mathbf{h}, \mathbf{D}) \xrightarrow[\substack{\|\mathbf{D}\|_F \rightarrow +\infty \\ \mathbf{D} \in \mathcal{S}_3^+}]{} +\infty. \quad (4.2.31)$$

The following lower bound holds for F :

$$\begin{aligned} (\forall \mathbf{D} \in \mathcal{S}_3) \quad F(\alpha, \beta, \mathbf{h}, \mathbf{D}) &\geq \lambda c_1 + \frac{\lambda}{2} \sum_{n=1}^N h_n (3 \ln(2\pi) + \Phi(\mathbf{D}) + \boldsymbol{\omega}_n^\top (\mathbf{D} + \epsilon_1 \mathbf{I}_3) \boldsymbol{\omega}_n) \\ &\quad + \epsilon_2 \|\mathbf{D}\|_F^2 \\ &\geq \lambda c_1 + \frac{3\lambda}{2} \ln(2\pi) + \frac{\lambda}{2} \Phi(\mathbf{D}) + \epsilon_2 \|\mathbf{D}\|_F^2, \end{aligned} \quad (4.2.32)$$

where $c_1 = \inf \left\{ \sum_{n=1}^N E(h_n) - h_n \ln \zeta \mid \mathbf{h} \in \Delta_N \right\}$ and we have used the fact that $\mathbf{D} + \epsilon_1 \mathbf{I}_3 \succ 0$. Finally, given the definition of Φ in (4.2.11), it is clear the lower bound in (4.2.32) goes to $+\infty$ when $\|\mathbf{D}\|_F \rightarrow +\infty$. Therefore, F admits a minimizer. \square

Let us now establish the convergence of the iterates generated by Algorithm 10. We base our analysis on the convergence result stated in [Xu and Yin, 2013, Thm. 2].

Theorem 4.6

Let $(\alpha^{(0)}, \beta^{(0)}, \mathbf{h}^{(0)}, \mathbf{D}^{(0)}) \in \mathcal{A} \times \mathcal{B} \times \Delta_N \times \mathcal{S}_3^+$ be an initial point. For every $\ell \in \mathbb{N}$, let $\mathbf{t}^{(\ell)} = (\alpha^{(\ell)}, \beta^{(\ell)}, \mathbf{h}^{(\ell)}, \mathbf{D}^{(\ell)})$ be the sequence generated by Algorithm 10. Assume that $\gamma_h < \frac{2}{L}$ with $L = \beta_+^2 \max \{(|\text{DFT}(\mathbf{x})_n|^2)_{1 \leq n \leq N}\}$ and DFT denotes the discrete Fourier transform. Assume that the grid is discretized finely enough such that the true bead \mathbf{x} has at least one pixel with unit intensity. Then $(\mathbf{t}^{(\ell)})_{\ell \in \mathbb{N}}$ converges to a critical point of the objective function (4.2.14), $\hat{\mathbf{t}} = (\hat{\alpha}, \hat{\beta}, \hat{\mathbf{h}}, \hat{\mathbf{D}})$.

Proof. We show that the assumptions required by [Xu and Yin, 2013, Thm. 2] are satisfied.

- **Splitting of the objective function.** We first split the objective function F into 1) a coupling term $f: \mathbb{R} \times \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3$ which is block convex, differentiable with a Lipschitz continuous gradient on bounded subsets, 2) a separable sum of functions $g_\alpha \in \Gamma_0(\mathbb{R})$, $g_\beta \in \Gamma_0(\mathbb{R})$, $g_h \in \Gamma_0(\mathbb{R}^N)$ and $g_D \in \Gamma_0(\mathcal{S}_3)$ such that:

$$F(\alpha, \beta, \mathbf{h}, \mathbf{D}) = f(\alpha, \beta, \mathbf{h}, \mathbf{D}) + g_\alpha(\alpha) + g_\beta(\beta) + g_h(\mathbf{h}) + g_D(\mathbf{D}). \quad (4.2.33)$$

To do so, we define

$$(\forall (\alpha, \beta, \mathbf{h}, \mathbf{D}) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^N \times \mathcal{S}_3)$$

$$f(\alpha, \beta, \mathbf{h}, \mathbf{D}) = \frac{1}{2} \|\mathbf{y} - \alpha \mathbf{1}_N - \beta(\mathbf{h} * \mathbf{x})\|^2 + \frac{\lambda}{2} \sum_{n=1}^N h_n (\Phi(\mathbf{D}) + \boldsymbol{\omega}_n^\top (\mathbf{D} + \epsilon_1 \mathbf{I}_3) \boldsymbol{\omega}_n), \quad (4.2.34)$$

$$g_\alpha = \iota_A, \quad g_\beta = \iota_B, \quad g_D = \iota_{\mathcal{S}_3^+} + \epsilon_2 \|\mathbf{D}\|_F^2, \quad (4.2.35)$$

and

$$(\forall \mathbf{h} \in \mathbb{R}^N) \quad g_h(\mathbf{h}) = \lambda \sum_{n=1}^N \left(E(h_n) - h_n \ln(\zeta) + \frac{3h_n}{2} \ln(2\pi) \right). \quad (4.2.36)$$

- **Properties of the partial functions.** We first need to check that the partial functions with respect to α and β are strongly convex along the iterates $(\mathbf{t}^{(\ell)})_{\ell \in \mathbb{N}}$, with modulus independent of ℓ . It is clear that it is the case for the partial function with respect to variable α . On the other hand, for any $\ell \in \mathbb{N}$, let us denote $F_B^{(\ell)}$ the function $\beta \mapsto F(\alpha^{(\ell+1)}, \beta, \mathbf{h}^{(\ell)}, \mathbf{D}^{(\ell)})$. Then, since the bead \mathbf{x} has at least one pixel value equal to 1, say at ω_i , $i \in \{1, \dots, N\}$,

$$\begin{aligned} \nabla^2 F_B^{(\ell)}(\beta) &= \|\mathbf{h}^{(\ell)} * \mathbf{x}\|^2 \\ &\geq \|\mathbf{h}^{(\ell)} * \boldsymbol{\delta}_i\|^2 \\ &= \|\mathbf{h}^{(\ell)}\|^2, \end{aligned} \quad (4.2.37)$$

with $\boldsymbol{\delta}_i \in \mathbb{R}^N$ the image corresponding to one illuminated pixel with intensity equal to 1 at ω_i , and 0 elsewhere. The inequality holds because $\mathbf{h}^{(\ell)} \in (0, +\infty)^N$. Moreover, since $\|\cdot\| \geq \frac{1}{N} \|\cdot\|_1$ and $\mathbf{h}^{(\ell)} \in \Delta_N$, we deduce that $\nabla^2 F_B^{(\ell)}(\beta) \geq \frac{1}{N^2}$. Therefore, $F_B^{(\ell)}$ is $\frac{1}{N^2}$ -strongly convex. Secondly, for the proximal linearized step on variable \mathbf{h} , the step-size γ_h has to satisfy the condition:

$$\gamma_h < \frac{2}{L}, \quad (4.2.38)$$

where L is the Lipschitz constant of the gradient of the differentiable component $\mathbf{h} \mapsto \frac{1}{2} \|\mathbf{y} - \alpha \mathbf{1}_N - \beta(\mathbf{h} * \mathbf{x})\|^2$. It can easily been shown that $L \leq \bar{L}$.

- **Boundedness of the sequence.** Let us demonstrate that the sequence $(\mathbf{t}^{(\ell)})_{\ell \in \mathbb{N}}$ is bounded. Since GENTLE alternates exact, proximal and proximal linearized updates, the sequence $(F(\mathbf{t}^{(\ell)}))_{\ell \in \mathbb{N}}$ is non-increasing. In addition, the objective function is coercive as seen in Proposition 4.5. This implies that $(\mathbf{t}^{(\ell)})_{\ell \in \mathbb{N}}$ is bounded.
- **Kurdyka-Łojasiewicz's inequality.** Lastly, the objective function F must satisfy the so-called Kurdyka-Łojasiewicz's inequality [Kurdyka, 1998, Bolte et al., 2007]. This property is satisfied by a wide range of functions in the context of image processing. In the present case, we omit the proof that F satisfies the Kurdyka-Łojasiewicz's inequality as a very similar proof to the one in [Chouzenoux et al., 2019b] could be driven.

Finally, applying [Xu and Yin, 2013, Thm. 2], we deduce that $(\mathbf{t}^{(\ell)})_{\ell \in \mathbb{N}}$ converges to a critical point of the objective function F . In practice, we assess the convergence using the stopping criterion $\|\mathbf{t}^{(\ell+1)} - \mathbf{t}^{(\ell)}\| \leq \varepsilon$ with $\varepsilon = 10^{-7}$.

□

4.2.5 Validation of the PSF calibration method

In this section, we present experiments conducted to validate the proposed PSF calibration method GENTLE. These experiments have been designed to test the performance of our method both in simulated scenarios and in real-life settings. In all our experiments, the regularization parameter λ is determined through an empirical grid search. Specifically, we select the λ that minimizes the criterion $\|\mathbf{y} - \hat{\alpha} - \hat{\beta}\mathbf{g}(\hat{\mathbf{D}} + \epsilon_1 \mathbf{I}_3) * \mathbf{x}\|^2$, given estimates $(\hat{\alpha}, \hat{\beta}, \hat{\mathbf{D}})$ produced by GENTLE for a certain λ . The other hyper-parameters are set to $(\alpha_-, \alpha_+) = (0, 1)$, $(\beta_-, \beta_+) = (0, 3)$, and $\epsilon_1 = \epsilon_2 = 10^{-6}$.

4.2.5.1 Experiments in simulated scenarios

We simulate a 3D synthetic bead image $\mathbf{x} \in \mathbb{R}^N$ of $1\mu\text{m}$ of diameter on a regularly spaced grid with size $N = 40 \times 40 \times 80$ and voxel dimension $0.05 \times 0.05 \times 0.1\mu\text{m}^3$ (which is a typical resolution grid in MPM). The observation $\mathbf{y} \in \mathbb{R}^N$ is then simulated as $\mathbf{y} = \bar{\alpha} + \bar{\beta}\bar{\mathbf{h}} * \mathbf{x} + \boldsymbol{\nu}$, for fixed values of $\bar{\alpha} \in [0, +\infty)$, $\bar{\beta} \in [0, +\infty)$, $\bar{\mathbf{h}} \in \Delta_N$ and $\boldsymbol{\nu} \in \mathbb{R}^N$ the realization of a zero-mean Gaussian noise, with standard deviation chosen so as to obtain a given input signal-to-noise ratio (SNR).

We choose generalized exponential distributions shapes for the ground truth PSF $\bar{\mathbf{h}}$. This family of distributions not only includes Gaussian ones, but also effectively captures those with shorter or longer tails. In microscopy, the PSF often deviates from a ideal Gaussian form, optical aberrations being a notable factor influencing this shape variation. [Stallinga and Rieger, 2010]. These distributions are uniquely defined by a matrix $\bar{\mathbf{S}} \in \mathcal{S}_3^+$ and a parameter $\eta > 0$, as

$$\bar{\mathbf{h}} \propto \left(\exp \left\{ -\frac{1}{2} (\omega_n^\top (\bar{\mathbf{S}}) \omega_n)^{\eta/2} \right\} \right)_{1 \leq n \leq N}, \quad (4.2.39)$$

where $\bar{\mathbf{h}}$ is normalized so that $\bar{\mathbf{h}} \in \Delta_N$. When $\eta = 2$, we retrieve a Gaussian distribution. The matrix $\bar{\mathbf{S}}$ can be designed to represent a kernel with a specific inclination and width. Formally, $\bar{\mathbf{S}}$ can be decomposed in an orthonormal basis so that it only depends on the second and third Euler angles $(\bar{\theta}, \bar{\varphi}) \in [0, \pi] \times [-\pi, \pi]$ and on the eigenvalues $\bar{s} \in (0, +\infty)^3$ for each principal direction. Indeed, for symmetry reasons, the first Euler angle, corresponding to a rotation around the vertical axis, has no effect so can be set to 0. We thus adopt the notation $\bar{\mathbf{S}} = \mathbf{S}(\bar{\theta}, \bar{\varphi}, \bar{s})$. GENTLE provides an estimate $\hat{\mathbf{D}}$ such that $\hat{\mathbf{D}} + \epsilon_1 \mathbf{I}_N$ approaches $\mathbf{S}(\bar{\theta}, \bar{\varphi}, \bar{s})$.

We draw a comparison of our method GENTLE with another method, which will refer to as Nonlinear Least Squares (NLS) and which has been extensively used in MPM [De Moraes Marim et al., 2008, Kirshner et al., 2013, Zhu and Zhang, 2013]. It corresponds to considering directly the following non-regularized and non-multiconvex problem:

$$\underset{\substack{\alpha \in \mathcal{A}, \beta \in \mathcal{B}, \\ \theta \in [0, \pi], \varphi \in [-\pi, \pi], s \in (0, +\infty)^3}}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{y} - \alpha \mathbf{1}_N - \beta \mathbf{g}(\mathbf{S}(\theta, \varphi, s)) * \mathbf{x}\|^2, \quad (4.2.40)$$

where $\mathbf{g}(\mathbf{S}(\theta, \varphi, s))$ is the normalized discrete Gaussian kernel on Ω defined with the Euler angles (θ, φ) and the eigenvalues s . This problem can be tackled using a Levenberg-Marquardt solver [Marquardt,

[1963]. Note that, despite its popularity, NLS has its shortcomings, notably it lacks convergence guarantees.

Our results are reported in Figure 4.2 using the Percent root-mean-square difference (PRD) metric, defined as $100 \times \|\hat{\alpha} + \hat{b}(\hat{\mathbf{h}} * \mathbf{x}) - \bar{\alpha} - \bar{b}(\bar{\mathbf{h}} * \mathbf{x})\| / \|\bar{\alpha} + \bar{b}(\bar{\mathbf{h}} * \mathbf{x})\|$. In the case of GENTLE, variable $\hat{\mathbf{h}}$ is obtained as an output, while for NLS, we set $\hat{\mathbf{h}} = \mathbf{g}(\mathbf{S}(\hat{\theta}, \hat{\varphi}, \hat{\mathbf{s}}))$. For this experiment, the true Euler angles were set to $(\bar{\theta}, \bar{\varphi}) = (5\pi/6, \pi/6)$, the true eigenvalues to $\bar{\mathbf{s}} = (138.6, 138.6, 3.2)$, and the voxels dimensions to $0.049 \times 0.049 \times 0.1 \mu\text{m}^3$. The results were averaged over 10 random noise realisations. As one can observe, NLS performance degrades significantly for values of the model exponent η lower or larger than 2, i.e., when there is a mismatch between the assumed Gaussian model and the true one. By contrast, although promoting Gaussian shapes using a regularization strategy, GENTLE manages to provide accurate and robust estimations even in the presence of non-Gaussian generalized exponential PSF shapes.

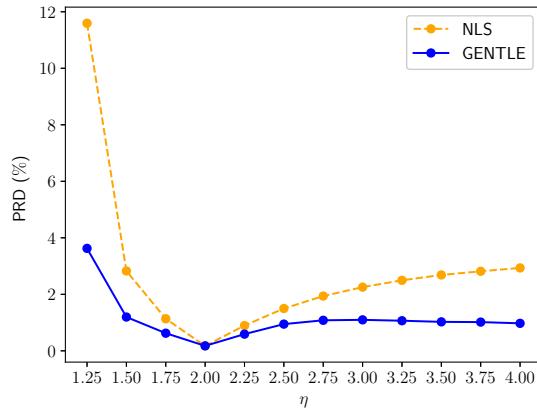


Figure 4.2: Percent root-mean-square difference (PRD) as a function of the parameter η , for both GENTLE and NLS, for a noise level corresponding to SNR=10dB.

4.2.5.2 Experiments on real beads in homogeneous medium

The next step of our validation process consists in testing our method on real beads acquired under optimal conditions, namely, beads within in a homogenous medium. Such setting is typical in MPM device calibration, although it usually involves sub-resolution beads instead of $1\mu\text{m}$ diameter beads. The PSF depends on the system optical configuration, including the lens properties and acquisition settings [Diaspro, 2001, Cole et al., 2011]. We evaluate the PSF estimation using the Full Width at Half Maximum (FWHM) as a metric, as it is commonly done in the microscopy community. Measured in micrometers for our study, the FWHM represents the width of a bell-shaped curve when it is at half peak value. In the ideal case of multiphoton acquisitions where the optics are assumed “perfect”, the values of the FWHM along each principal axis X' , Y' and Z' are [Diaspro, 2001]:

$$\text{FWHM}_{X'} \simeq \text{FWHM}_{Y'} \simeq \frac{0.7\lambda_{\text{em}}}{\text{NA}}, \quad (4.2.41)$$

$$\text{FWHM}_{Z'} \simeq \frac{2.3\lambda_{\text{em}}n_r}{\text{NA}^2}, \quad (4.2.42)$$

where λ_{em} is the emission wavelength, n_r is the refractive index of the immersion medium and NA the numerical aperture. The above formula will serve as a reference to check the consistency of GENTLE results.

For our experiment, we imaged a distilled water solution ($n_r = 1.33$) containing fluorescent polystyrene microspheres with a diameter of $\tau = 1\mu\text{m}$, marked with a yellow-green fluorophore emitting at 515 nm. The acquisition was performed with an excitation wavelength $\lambda_{\text{exc}} = 810$ nm, a numerical aperture NA = 1.05, and a voxel size of $0.037 \times 0.037 \times 0.05\mu\text{m}^3$.

In the obtained image, we were able to select four individual volumes of interest containing isolated beads, using our cropping procedure, on which we applied GENTLE with $\epsilon_1 = \epsilon_2 = 10^{-6}$, so yielding estimates $(\hat{\alpha}, \hat{\beta}, \mathbf{H}, \hat{\mathbf{D}})$. Then, performing a singular value decomposition of $\hat{\mathbf{D}} + \epsilon_1 \mathbf{I}_3$ and using the trigonometry formulas from [Depriester, 2018], we deduce the Euler angles $(\hat{\theta}, \hat{\varphi}) \in [0, \pi] \times [-\pi, \pi]$ and the eigenvalues $\hat{\mathbf{s}} = (\hat{s}_{X'}, \hat{s}_{Y'}, \hat{s}_{Z'}) \in (0, +\infty)^3$ along each principal axis (where re-ordering is performed to align the axis with the Cartesian grid (X, Y, Z)). The FWHM along each axis $A \in \{X', Y', Z'\}$ is obtained as $\text{FWHM}_A = 2\sqrt{2\ln(2)/\hat{s}_A}$.

As Figure 4.3 demonstrates, the estimated FWHM and angles are consistent across the four beads, validating the assumption of a stationary PSF in the observed field. Moreover, the average FWHM values are close to the theoretical expectations computed with (4.2.41) and (4.2.42), equal to $0.34\mu\text{m}$ along the axial axes, and $1.43\mu\text{m}$ in the depth direction, which validates our approach.

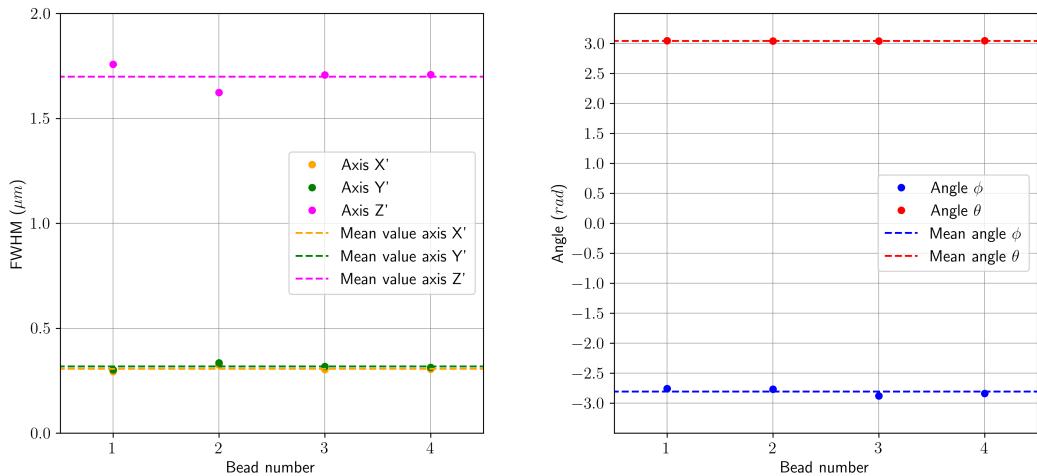


Figure 4.3: Estimated FWHM (left) in the 3 principal component directions $\{X', Y', Z'\}$, and associated Euler angles (right), for the four isolated beads.

4.3 Proposed solution for the inverse problem

We now shift our attention to the image restoration phase, where a non-calibrated and potentially complex, object, denoted by $\bar{\mathbf{x}} \in \mathbb{R}^M$, is observed, through the multi-photon microscope. The (degraded) observation is denoted $\mathbf{y} \in \mathbb{R}^M$. Note that the problem dimension M usually differs from the dimension associated with the prior PSF estimation problem, typically with $M > N$, as the object of interest is

typically much more spread than a micrometric bead. The goal is to accurately reconstruct an estimate of $\bar{\mathbf{x}} \in \mathbb{R}^M$ given $\mathbf{y} \in \mathbb{R}^M$, and our knowledge of the PSF, deduced from the approach from Section 4.2. Note that, in a practical scenario, the calibration phase could have been performed at another grid resolution, which would require an interpolation step, before using the estimated PSF from GENTLE algorithm. For the sake of simplicity, we opt here for a simple (though common) setting where the PSF $\mathbf{h} \in \Delta_N$ has been estimated under the same voxel resolution than the acquisition of the new object. GENTLE method also provides the background parameter, $\alpha \in \mathbb{R}$. The inverse problem thus reads,

$$\mathbf{y} = \mathcal{D}(\mathbf{H}\bar{\mathbf{x}} + \alpha), \quad (4.3.1)$$

where $\mathcal{D}: \mathbb{R}^M \rightarrow \mathbb{R}^M$ represents the noise model and $\mathbf{H} \in \mathbb{R}^{M \times M}$ is the linear operator such that, for every $\mathbf{u} \in \mathbb{R}^M$, $\mathbf{H}\mathbf{u} = \mathbf{h} * \mathbf{u}$. Here, the convolution product is performed with zero-padding. While the calibration step is usually performed in ideal acquisition conditions, where the Gaussian additive noise assumption is sufficient, the situation differs in the image restoration stage (for example, *in vivo* acquisitions might require low laser power, and as such, leads to low-photon counting). It then becomes necessary to introduce a more realistic noise model, to guarantee high quality restored images. In the following we introduce the considered noise model, we propose a direct method for estimating noise parameters, as well as a variational approach paired with an algorithmic solution to address the inverse problem (4.3.1).

4.3.1 A heteroscedastic noise model for MPM

Many restoration techniques for MPM imaging assume a Gaussian homoscedastic noise (i.e., with constant variance across the whole image) [Lefort et al., 2021, Danielyan et al., 2014, Doi et al., 2018]. While such assumption simplifies computational needs, it might not always align with the reality of fluorescence microscopy imaging. The Gaussian assumption generally holds true for high-exposure scenes with abundant photons per the central limit theorem. In contrast, multiphoton or confocal imaging often exhibits a Poisson or Poisson-Gaussian noise profile [Crivaro et al., 2011]. In [Van Kempen et al., 1997], the authors illustrate the benefits of adopting a Poisson noise model over a Gaussian one in terms of restoration quality. However, handling such noise types can be computationally intensive, as it requires minimization of the non-smooth Poisson data-fidelity function [Zanella et al., 2009, Carlavan and Blanc-Féraud, 2011, Chouzenoux et al., 2015]. The mixed Poisson-Gaussian noise model describes the corrupted image $\mathbf{y} = (y_m)_{1 \leq m \leq M}$ as

$$(\forall m \in \{1, \dots, M\}) \quad y_m \sim \alpha + a\mathcal{P}(a^{-1}[\mathbf{H}\bar{\mathbf{x}}]_m) + \mathcal{N}(0, b), \quad (4.3.2)$$

with some noise parameters $a \in (0, +\infty)$ and $b \in (0, +\infty)$. Such noise consists of the mixture of a multiplicative noise with mean equal to the intensity of the pixel and a variance proportional to the intensity of the pixel up to the fixed scale a , and an additive Gaussian noise identically distributed in each pixel, corresponding to a background noise with variance b . However, as shown in [Chouzenoux et al., 2015], the resulting likelihood term is complicated, involving intractable series that need to be approximated, at the price of the computational time.

In the present work, we consider instead an additive heteroscedastic (i.e., with variance depending on the pixel) noise model approximating the Poisson-Gaussian noise, in the line of the method proposed in [Foi et al., 2008, Foi, 2009] for low-exposure natural images. To transform the Poisson-Gaussian noise in (4.3.2) into a fully additive one, we use the following well-known result: if X is a random variable following a Poisson distribution $\mathcal{P}(\theta)$, with $\theta > 0$, then

$$\frac{1}{\sqrt{\theta}}(X - \theta) \xrightarrow[\theta \rightarrow +\infty]{\mathcal{L}} \bar{X}, \quad (4.3.3)$$

where $\bar{X} \sim \mathcal{N}(0, 1)$ and \mathcal{L} denotes the convergence in distribution. In other words, for large values of θ , the approximation $\mathcal{P}(\theta) \simeq \mathcal{N}(\theta, \theta)$ is valid. Therefore, assuming the multiplicative and background noises are independent, the final model of noise we consider is Gaussian, with a variance varying linearly with the intensity of the pixel, i.e.,

$$(\forall m \in \{1, \dots, M\}) \quad y_m = \alpha + [\mathbf{H}\bar{\mathbf{x}}]_m + w_m, \quad (4.3.4)$$

where $(w_m)_{1 \leq m \leq M}$ are independent variables sampled according to

$$(\forall m \in \{1, \dots, M\}) \quad w_m \sim \mathcal{N}(0, \sigma^2([\mathbf{H}\bar{\mathbf{x}}]_m)), \quad (4.3.5)$$

and σ is the function corresponding to the standard deviation, defined as

$$(\forall t \in \mathbb{R}) \quad \sigma(t) = \begin{cases} \sqrt{at + b} & \text{if } t \geq 0, \\ 0 & \text{otherwise.} \end{cases} \quad (4.3.6)$$

The noise model (4.3.5) depends on two parameters a and b , for which we present hereafter an estimation strategy.

4.3.2 Estimation of the noise parameters

We propose in Algorithm 11 our procedure for estimating heteroscedastic Gaussian noise parameters in 3D images. Our approach deviates from that proposed in [Foi et al., 2008]. In [Foi et al., 2008], the segmentation into non-overlapping level sets it performed through a wavelet decomposition of the noisy image, which we found too computationally intensive and unnecessary in our MPM context. Moreover, the protected toolbox associated with [Foi et al., 2008] is not tailored for 3D images.

Algorithm 11: Estimation of parameters a and b

Step 1: 3D image smoothing. Let $\mathbf{y} \in \mathbb{R}^M$ the observed 3D image. Convolve \mathbf{y} with a normalized uniform kernel of size $s \times s \times s$, with $s \in \mathbb{N}^*$, yielding a smoothed image \mathbf{y}_s .

Step 2: Volume segmentation.

1. Let $J \in \mathbb{N}^*$. Segment \mathbf{y}_s into J distinct parts using the Lloyd-Max estimator [Lloyd, 1982] for optimal quantization, so as to minimize the mean squared error between the segmented image and \mathbf{y}_s .
2. Denote $\ell_1 \leq \ell_2 \leq \dots \leq \ell_J$ the resulting J intensity levels.
3. Denote S_1, S_2, \dots, S_J the segmented parts of the image, corresponding to the intensity intervals $([\ell_{j-1}, \ell_j])_{1 \leq j \leq J}$.

Step 3: Intensity and variance estimation. For each segmented zone $(S_j)_{1 \leq j \leq J}$:

1. Estimate the intensity \hat{I}_j within S_j as:

$$\hat{I}_j = \frac{1}{|S_j|} \sum_{m \in S_j} y_m, \quad (4.3.7)$$

with $|S_j|$ denotes the number of pixels (i.e., the cardinal) in S_j .

2. Estimate the variance $\hat{\sigma}_j^2$ within S_j as:

$$\hat{\sigma}_j^2 = \frac{1}{|S_j|} \sum_{m \in S_j} (y_m - \hat{I}_j)^2. \quad (4.3.8)$$

Step 4: Linear regression. Obtain the coefficients (a, b) by conducting a least squares linear regression on the paired values $(\hat{I}_j, \hat{\sigma}_j^2)$ to fit Relation (4.3.6).

4.3.3 A constrained parameter-free deconvolution framework

After the noise modelling step, we move to the resolution of the inverse problem (4.3.1). A standard approach is to minimize a cost function composed of a data fidelity term, $f: \mathbb{R}^M \rightarrow \mathbb{R}$, in conjunction with a regularization function, $g: \mathbb{R}^M \rightarrow \mathbb{R}$, which can be expressed as

$$\underset{\mathbf{x} \in [0, +\infty)^M}{\text{minimize}} \quad f(\mathbf{x}) + \chi g(\mathbf{x}), \quad (4.3.9)$$

where $\chi > 0$ denotes a regularization parameter. The data fidelity term is designed to ensure that the reconstructed data aligns with the observation model. The regularization function introduces specific *a priori* characteristics to the reconstructed image, such as smoothness. The role of the regularization parameter is to balance the two terms. Its tuning can be based on image quality metrics such as the Mean Square Error or the SNR. However, when the ground truth or reference images are unavailable, this strategy becomes infeasible. Thus, image restoration often involves a time-consuming adjustment of the regularization parameter, with the final decision largely influenced by the user's expertise and subjective judgment. While various statistical methods exist for estimating this parameter [Antoni et al., 2023, Vatankhah et al., 2014], they are usually too computationally intensive for 3D data.

Instead, we propose to tackle the restoration challenge using a constrained approach (sometimes

referred to as the discrepancy principle). This strategy is reminiscent of those described in [Carrillo et al., 2012, Afonso et al., 2010a, Harizanov et al., 2013]. The data fidelity term is constrained so that it does not exceed a known (or easily estimated) value. Our formulation reads

$$\begin{aligned} & \underset{\mathbf{x} \in \mathbb{R}^M}{\text{minimize}} \quad g(\mathbf{x}), \\ & \text{subject to} \quad f(\mathbf{x}) \leq B \text{ and } \mathbf{x} \in [0, +\infty)^M, \end{aligned} \quad (4.3.10)$$

where $B > 0$ is a bound, either predetermined or deduced from the data. A main advantage of this formulation is that, by assuming an appropriate choice for the data fidelity function f , it is often feasible to derive a statistical-based upper bound for f , giving a good first trial value for B . It is worth noting that, if f and g are both convex, then for each $B > 0$, there usually exists a corresponding $\chi > 0$ such that (4.3.10) and (4.3.9) are equivalent. Let us now explicit our choices for f , g , and B .

4.3.4 Choices for the data-fidelity and regularization functions

Following our noise model (4.3.5), we set

$$(\forall \mathbf{x} \in \mathbb{R}^M) \quad f(\mathbf{x}) = \|\mathbf{W}(\mathbf{H}\mathbf{x} - \mathbf{y} + \boldsymbol{\alpha})\|^2. \quad (4.3.11)$$

with $\mathbf{W} \in \mathcal{S}_M^+$ a weighting matrix accounting for the heterodasticity of the noise. Here, we take

$$\mathbf{W} = \text{Diag} \left\{ \left(\frac{1}{\sigma([\mathbf{H}\bar{\mathbf{x}}]_m)} \right)_{1 \leq m \leq M} \right\}. \quad (4.3.12)$$

In practice, since $\mathbf{H}\bar{\mathbf{x}}$ is unknown, we approximate it by \mathbf{y}_s , a denoised version of \mathbf{y} as defined in Algorithm 11. Because of the large number of pixels, the law of large numbers makes the following approximation valid: $f(\mathbf{x}) \approx M$, which suggests setting $B = M$ in (4.3.10).

For the regularization function g , we propose a re-weighted smooth total variation term, adjusted according to the voxel size along the axes X , Y , and Z . Specifically, given a smoothing parameter $\delta > 0$, function g is defined as

$$(\forall \mathbf{x} \in \mathbb{R}^M) \quad g(\mathbf{x}) = \sum_{m=1}^M \sqrt{\delta + \frac{1}{r_X} (\mathbf{G}_X \mathbf{x})_m^2 + \frac{1}{r_Y} (\mathbf{G}_Y \mathbf{x})_m^2 + \frac{1}{r_Z} (\mathbf{G}_Z \mathbf{x})_m^2}, \quad (4.3.13)$$

with $\mathbf{G} = [\mathbf{G}_X^\top | \mathbf{G}_Y^\top | \mathbf{G}_Z^\top]^\top \in \mathbb{R}^{3M \times M}$, where matrices \mathbf{G}_X , \mathbf{G}_Y , and \mathbf{G}_Z represent discrete gradient operators along the axes X , Y , and Z , respectively.

4.3.5 Restoration algorithm

The resulting optimization problem (4.3.10) reads as the minimization of a smoothed convex function under convex constraints. An efficient Majorization-Minimization strategy [Sun et al., 2017], entitled P-MMS, was recently proposed in [Chouzenoux et al., 2022] to address such a class of optimization problem. Specifically tailored for large-scale constrained image processing problems such as ours, this algorithm features rapid execution times while providing theoretical convergence guarantees.

The central idea behind P-MMS is to use the external penalty principle to cope with the constraints.

Namely, for the constraints $f(\mathbf{x}) \leq B$ and $\mathbf{x} \in [0, +\infty)^M$, we introduce corresponding penalty functions, denoted by R_1 and R_2 :

$$(\forall \mathbf{x} \in \mathbb{R}^M) \quad R_1(\mathbf{x}) = d_{B(\mathbf{0}_M, B)}^2(\mathbf{W}(\mathbf{H}\mathbf{x} - \mathbf{y} + \alpha)), \quad (4.3.14)$$

and

$$(\forall \mathbf{x} \in \mathbb{R}^M) \quad R_2(\mathbf{x}) = d_{[0, +\infty)^M}^2(\mathbf{x}), \quad (4.3.15)$$

where d_C is the distance to the set $C \subseteq \mathbb{R}^M$. Note that the gradient of R_1 and R_2 are easily computable since the projections onto the ball $B(\mathbf{0}_M, B)$ and the nonnegative orthant $[0, +\infty)^M$ are closed form. A key assumption for P-MMS to be applicable is that every function in the problem admits a tangent quadratic upper bound at every point. In mathematical terms, for a differentiable function $\psi: \mathbb{R}^M \rightarrow \mathbb{R}$, this requirement corresponds to the existence, for every $\mathbf{x}' \in \mathbb{R}^M$, of a curvature matrix $\mathbf{A}_\psi(\mathbf{x}') \in \mathcal{S}_M^+$, such that

$$(\forall \mathbf{x} \in \mathbb{R}^M) \quad \psi(\mathbf{x}) \leq \psi(\mathbf{x}') + \nabla \psi(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + (\mathbf{x} - \mathbf{x}')^\top \mathbf{A}_\psi(\mathbf{x}') (\mathbf{x} - \mathbf{x}'). \quad (4.3.16)$$

The existence of such quadratic upper bounds can be readily derived for each function g , R_1 , and R_2 in our problem, using the descent Lemma [Bauschke and Combettes, 2019, Lemma 2.64] or the half-quadratic majorization formula [Chouzenoux et al., 2013a, Lemma 1]. This leads the following valid curvature matrices:

$$\mathbf{A}_g(\mathbf{x}) = \mathbf{G}^\top \text{BDiag} \left\{ \left(\left(\delta + \frac{(\mathbf{G}_X \mathbf{x})_m^2}{r_X} + \frac{(\mathbf{G}_Y \mathbf{x})_m^2}{r_Y} + \frac{(\mathbf{G}_Z \mathbf{x})_m^2}{r_Z} \right)^{-1/2} \mathbf{I}_3 \right)_{1 \leq m \leq M} \right\} \mathbf{G}, \quad (4.3.17)$$

$$\mathbf{A}_{R_1}(\mathbf{x}) = \mathbf{H}^\top \mathbf{W}^\top \mathbf{W} \mathbf{H}, \quad \text{and} \quad \mathbf{A}_{R_2}(\mathbf{x}) = 2\mathbf{I}_M, \quad (4.3.18)$$

where BDdiag stand for block diagonal matrix. Then, for any penalty parameter $\gamma > 0$, the penalized function

$$F_\gamma(\mathbf{x}) = g(\mathbf{x}) + \gamma(R_1(\mathbf{x}) + R_2(\mathbf{x})) \quad (4.3.19)$$

admits a quadratic tangent majorant at \mathbf{x} uniquely defined by its curvature matrix:

$$\mathbf{A}_{F_\gamma}(\mathbf{x}) = \mathbf{A}_g(\mathbf{x}) + \gamma(\mathbf{A}_{R_1}(\mathbf{x}) + \mathbf{A}_{R_2}(\mathbf{x})). \quad (4.3.20)$$

We present in Algorithm 12 the P-MMS approach, to solve Problem (4.3.10). At each iteration of the inner algorithm, \mathbf{x}_{k+1} is updated within the affine space defined by the directions $\{-\nabla F_{\gamma_j}(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}_{k-1}\}$, employing a Majorization-Minimization approach to determine the multidimensional step-size. A local version P-MMS^{loc} of this algorithm was developed which implements a trust-region strategy to accelerate further convergence. We redirect readers to [Chouzenoux et al., 2022] or to its publicly available code, for more details. The P-MMS algorithm benefits from the convergence guarantees given in Theorem 4.3.5.

Algorithm 12: P-MMS

Inputs: $(\gamma_j)_{j \in \mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$, $(\varepsilon_j)_{j \in \mathbb{N}} \in (\mathbb{R}^+)^{\mathbb{N}}$, $\mathbf{x}_0 \in \mathbb{R}^M$.

```

for  $j = 0, 1, \dots$  do
    // find an (approximated) minimizer of  $F_{\gamma_j}$ 
    Set initial point  $\mathbf{x}_0$ ,
    for  $k = 1, \dots$  do
        Construct subspace directions  $\mathbf{D}_k = [-\nabla F_{\gamma_j}(\mathbf{x}_k), \mathbf{x}_k - \mathbf{x}_{k-1}]$ ,
        Compute  $\mathbf{A}_{F_{\gamma_j}}(\mathbf{x}_k)$  according to (4.3.20),
         $\mathbf{u}_k = -[\mathbf{A}_{F_{\gamma_j}}(\mathbf{x}_k)]^\dagger \mathbf{D}_k^\top \nabla F_{\gamma_j}(\mathbf{x}_k)$ ,
         $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_k$ ,
        if  $\|\nabla F_{\gamma_j}(\mathbf{x}_{k+1})\| < \varepsilon_j$  then
            exit loop // stop inner algorithm if given precision  $\varepsilon_j$  on the norm
            of the gradient is reached
            return  $\mathbf{x}_{k+1}$ .
     $\mathbf{x}_j = \mathbf{x}_{k+1}$ .
return  $\mathbf{x}_j$ .

```

Theorem 4.7. Convergence of the P-MMS algorithm

Assume the sequence of parameters $(\varepsilon_j)_{j \in \mathbb{N}}$ satisfies, for every $j \in \mathbb{N}$, $\varepsilon_j > 0$ and $\lim_{j \rightarrow +\infty} \varepsilon_j = 0$. Also assume that $(\gamma_j)_{j \in \mathbb{N}}$ is a nondecreasing sequence of positive reals and $\lim_{j \rightarrow +\infty} \gamma_j = +\infty$. Then, the sequence $(\mathbf{x}_j)_{j \in \mathbb{N}}$ generated by Algorithm 12 is bounded and any of its cluster point is a solution to Problem (4.3.10).

Proof. It suffices to check that the assumptions for [Chouzenoux et al., 2022, Theorem 2] are satisfied.

- The functions g , R_1 and R_2 are differentiable,
- For every $\gamma > 0$, the function F_γ is coercive, convex, and satisfies the KL property.
- For every $\gamma > 0$, the curvature function defined in (4.3.20) is lower bounded independently of \mathbf{x} and is continuous.

This concludes the proof. □

4.3.6 Validation of the restoration method on simulated data

To illustrate the performance of our heteroscedastic constrained formulation for the MPM image restoration task, we first conducted an experiment using simulated data.

For this experiment, we chose an image of a fly brain from [Chalvidal et al., 2023] as the object of interest $\bar{\mathbf{x}}$. This image, with dimensions $M = 128 \times 128 \times 40$, was artificially degraded through a convolution operator \mathbf{H} mimicking the effect of a normalized 3D Gaussian kernel with inverse covariance matrix parameterized by the angles $\theta = 5\pi/6$ rad, $\varphi = 0$ rad, and eigenvalues $\mathbf{s} = (50, 50, 50)$. Moreover, heteroscedastic noise was introduced based on the noise model presented in (4.3.5), with $a = 0.01$ and $b = 10^{-5}$. The voxels dimensions were configured to $0.05 \times 0.05 \times 0.05 \mu\text{m}^3$.

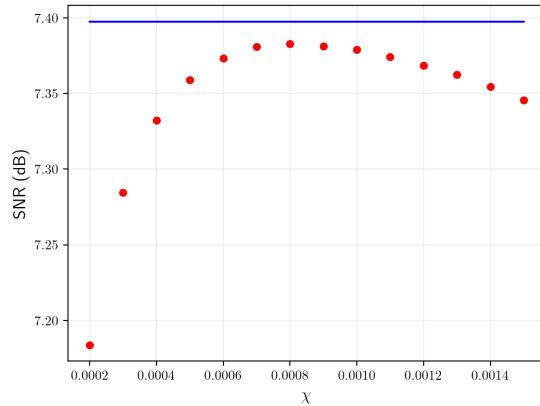


Figure 4.4: Comparison of the penalized least-squares approach (red dots) with the proposed constrained approach assuming a Gaussian heteroscedastic noise model (blue line), in terms of restored image SNR, in dB.

To provide a comprehensive evaluation, we compare our approach against the more traditional regularized approach in (4.3.9), where we choose the data-fidelity function f to fit homoscedastic Gaussian noise, setting $f(\mathbf{x}) = \|\mathbf{Hx} - \mathbf{y} + \alpha\|^2$, and the smoothed TV regularization function g defined in (4.3.13).

For both models, the restoration was performed running the P-MMS algorithm. The parameter δ was set to 0.1 and, for the constrained problem, the bound B to the number of pixels, i.e., $B = M$. The penalty parameters $(\gamma_j)_{j \in \mathbb{N}}$ in Algorithm 12 were set to $\gamma_j = (2j)^2$ and the precision parameters $(\varepsilon_j)_{j \in \mathbb{N}}$ were chosen as $\varepsilon_j = 10^5 / (\gamma_j)^{0.75}$. The algorithm was initialized with $\mathbf{x}_0 = \mathbf{y}$.

Figure 4.5 displays the SNR as a function of the iteration number of the P-MMS algorithm for the constrained formulation, confirming the theoretically convergence established in Proposition 4.3.5, and illustrating the fast stability after few dozens of iterations only. Figure 4.4 captures the performance of the two methodologies. It showcases the SNR of the restored images using the standard regularized approach as a function of the regularization parameter χ , as well as the SNR achieved using our parameter-free method. A notable observation is the important fluctuation in SNR values for the regularized method as χ gets closer to its optimal value. It is crucial to highlight that this optimal value of χ remains unknown in the real-world context of Multiphoton Microscopy (MPM), as the SNR metric is not available. In contrast, our constrained methodology consistently delivers superior SNR, underscoring the interest of a reliable noise model. The absence of any tuning parameter is furthermore a crucial advantage in real-life applications.

4.4 Application of the proposed pipeline to muscle tissue imaging

In this section, we apply the entire pipeline, encompassing both PSF estimation and deconvolution, to the restoration of a mouse muscle volume, characterized by its prominent myosin filaments, from MPM acquisitions. All mice were bred and housed in Limoges University animal facility under controlled conditions (20°C, 12 hours light/12 hours dark cycle) with free access to standard mouse chow and

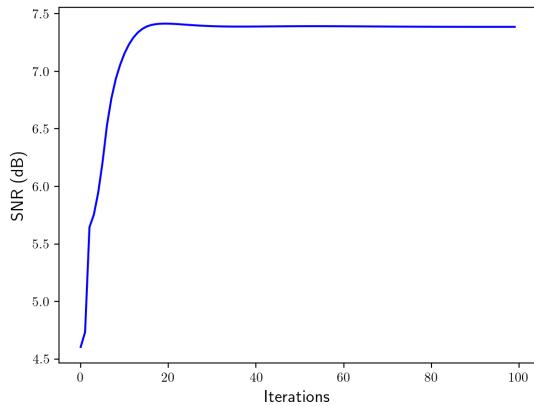


Figure 4.5: SNR evolution along the iterations j in Algorithm 12 for the resolution of the proposed constrained formulation.

tap water. Experimental procedures were carried out in accordance with the European legislation on animal experimentation (Directive 2010/63/UE), and approved by Ethical Committee no.033 (APAFIS #1903-2015091612088147 v2). Phenotypic and molecular analyses were performed on 12-week-old male animals.

4.4.1 Presentation of the experimental setup

A mouse muscle sample was immersed in a water solution. As a reference for the PSF estimation phase, polystyrene microspheres of $1\mu\text{m}$ diameter were integrated along the muscle's perimeter. We employed a bi-channel raw acquisition technique, combining both the Second Harmonic Generation signal from the myosin and the two-photon fluorescence from the microspheres. Thus, the PSF is calibrated in the same field of view, and under the same resolution, than the one where we observe the myosin organization. Such experimental setting is somewhat ideal, as the conditions for recording the PSF match precisely those for capturing the image of the muscle sample. Given that the muscle assembly acts as an absorbing and scattering medium that distorts the optical wavefront, recording the PSF in the location where the image is restored enhances restoration accuracy. The microscopy acquisition was carried out with an excitation wavelength of 810nm. We used the same device than in our previous experiments, that is a water immersion objective tailored for multiphoton acquisitions, specifically the Olympus XLPLN25XWMP with a numerical aperture NA = 1.05. The selected voxel dimensions for the imaging process were $0.049 \times 0.049 \times 0.1\mu\text{m}^3$. The resulting volume was of dimension $M = 840 \times 840 \times 180$.

4.4.2 Results for PSF calibration step

We first conducted the PSF estimation on the channel of the acquired image that contains the beads, that is we perform an *in situ* PSF calibration within an heterogeneous medium. Given these conditions, we anticipate a notably higher dispersion compared to the optimal setting discussed in Section 4.2.5.2. We fitted a convolution kernel on 8 cropped volumes, containing isolated bead images, using GENTLE (Algorithm 10). The hyper-parameters of the PSF estimation model (4.2.5) were set identically to Section

4.2.5. Figure 4.6 presents a comparison of the estimated FWHMs along the principal axis and Euler angles for the 8 beads, using the GENTLE method. Estimations across different beads again exhibit consistency, confirming the PSF stationarity assumption in this experiment. In addition, we display in Figure 4.7 the intensity profiles of the observed beads and of their reconstitution $\hat{\alpha} + \hat{\beta}\hat{h} * \mathbf{x}$, where \mathbf{x} is the theoretical bead. The GENTLE method accurately fits the observation across all three axes and, in particular, fits asymmetrical shapes more effectively than NLS, thanks to its robust formulation.

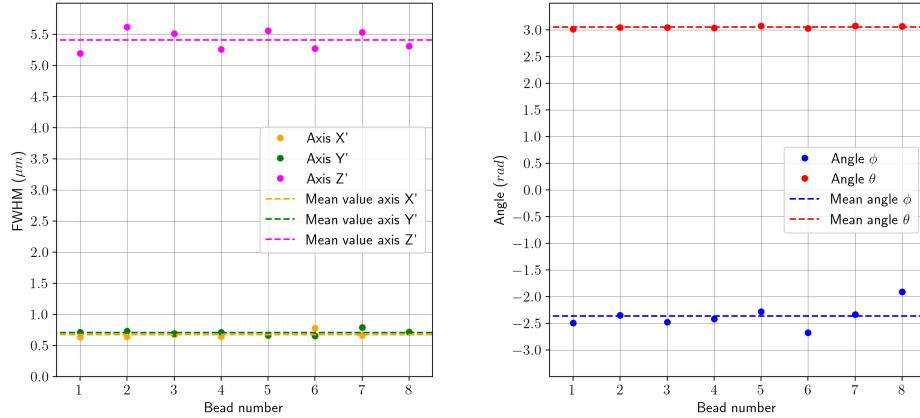


Figure 4.6: Estimated FWHM (left) in the 3 principal component directions denoted by X' , Y' and Z' , and angles (right) for the eight isolated beads.

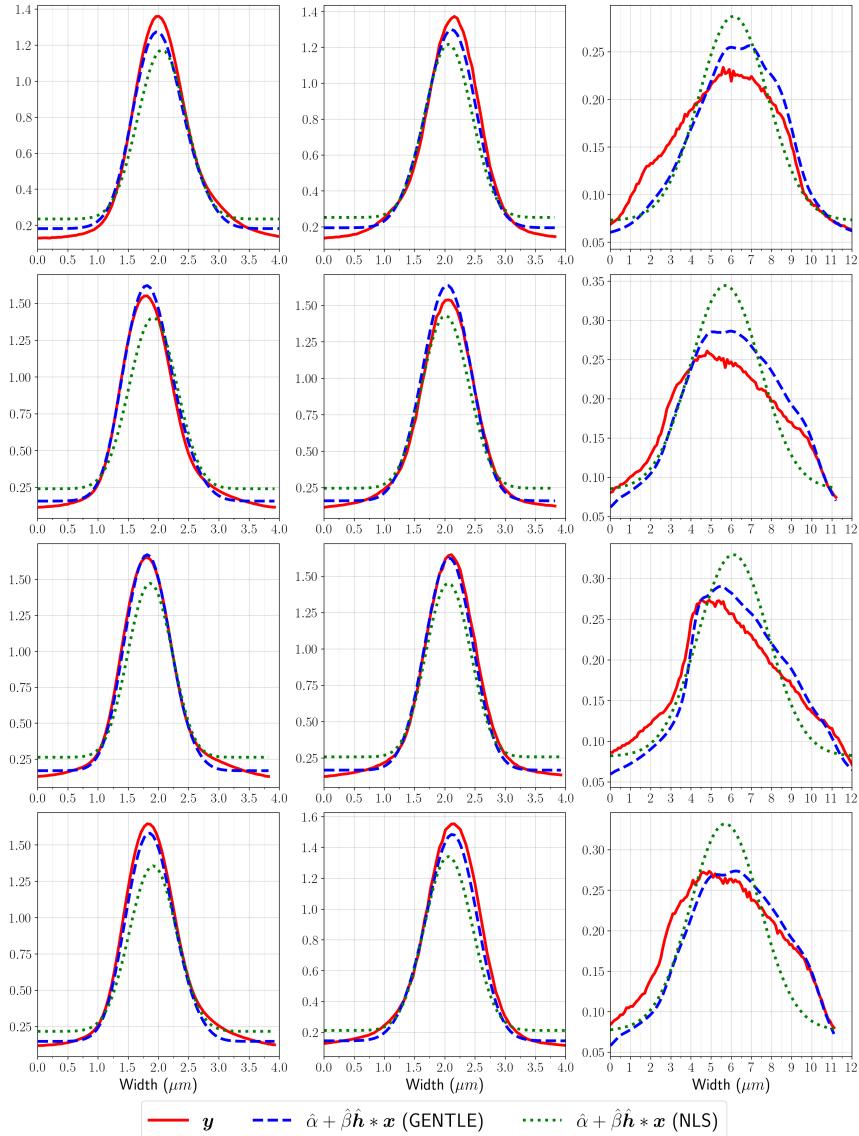


Figure 4.7: Profiles along axis X (left), Y (center) and Z (right) for 4 of the 8 isolated beads (rows). We compare the intensity profiles of the observation \mathbf{y} (red), the reconstitution $\hat{\alpha} + \hat{\beta}\hat{\mathbf{h}} * \mathbf{x}$ obtained with GENTLE (blue) and the reconstitution obtained with NLS (green). The profiles for the remaining 4 beads, omitted for space reason, are consistent with the displayed ones.

4.4.3 Results for the restoration step

In the restoration phase, we first determined the noise parameters a and b across the entire volume using the methodology described in Algorithm 11 with $s = 5$ and $J = 25$. Figure 4.8 displays the linear regression corresponding to the fourth step of the algorithm. The close alignment of the points confirms the appropriateness of the heteroscedastic noise model proposed.

Subsequently, the standard deviations $(\sigma([\mathbf{H}\bar{\mathbf{x}}]_m))_{1 \leq m \leq M}$ were approximated using (4.3.6) with the estimated a , b , and a smoothed version of the image $\mathbf{H}\bar{\mathbf{x}}$ corresponding to the convolution of \mathbf{y} with a $3 \times 3 \times 3$ uniform kernel. We solved Problem (4.3.10) by setting $B = M$ and $\delta = 0.1$, using Algorithm

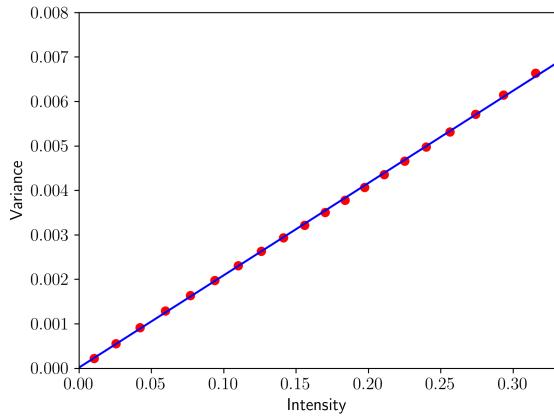


Figure 4.8: Linear regression (blue line) performed on the couples of values $(\hat{I}_j, \hat{\sigma}_j^2)$ (red dots) for the estimation of noise parameters $(a, b) \in \mathbb{R}^2$ on the mouse muscle 3D image.

12. The penalty parameters $(\gamma_j)_{j \in \mathbb{N}}$ were set to $\gamma_j = (1.5j)^{1.2}$ and the precision parameters $(\varepsilon_j)_{j \in \mathbb{N}}$ were chosen as $\varepsilon_j = 10^5 / (\gamma_j)^{0.5}$.

In Figure 4.9, we present slices from a portion of the raw image alongside corresponding slices from the restored image. The visual improvements are evident both in term of denoising and deblurring. In Figure 4.9(b) and (d), the presence of myofibrils appears clearly in the optical plan (XY) and in the optical axis (YZ). The myofiber structure is following the (OY) axis.

4.5 Conclusion

In this chapter, we introduced an end-to-end approach for addressing the MPM image restoration problem. We decomposed the primary problem into two sub-problems: PSF estimation and image deconvolution, and for both, we proposed original formulation and optimization methods. Our results on simulated data, real beads imaged under optimal conditions, and mouse muscle samples, demonstrate the efficiency of our approach.

As highlighted in the chapter, our method uses different noise models in its two distinct phases: initially a Gaussian model, and then a Poisson-Gaussian model. While the current setup has proven effective, considering the same heteroscedastic noise model throughout the methodology could be a promising next step to improve further the results.

Additionally, a future direction for this research could involve extending the PSF estimation strategy to other objects than beads. Given the generality of the method, it has potential applicability to any reference object with known geometrical structure.

On the restoration side, there is potential to investigate advanced regularization techniques to enhance visual outcomes as our framework is versatile, and not restricted to TV-based regularization. Lastly, the heteroscedastic noise model we proposed for MPM deserves deeper exploration. A study analyzing how parameters a and b fluctuate based on different optical settings and depth in the sample could provide insights on the physical properties of the noise.

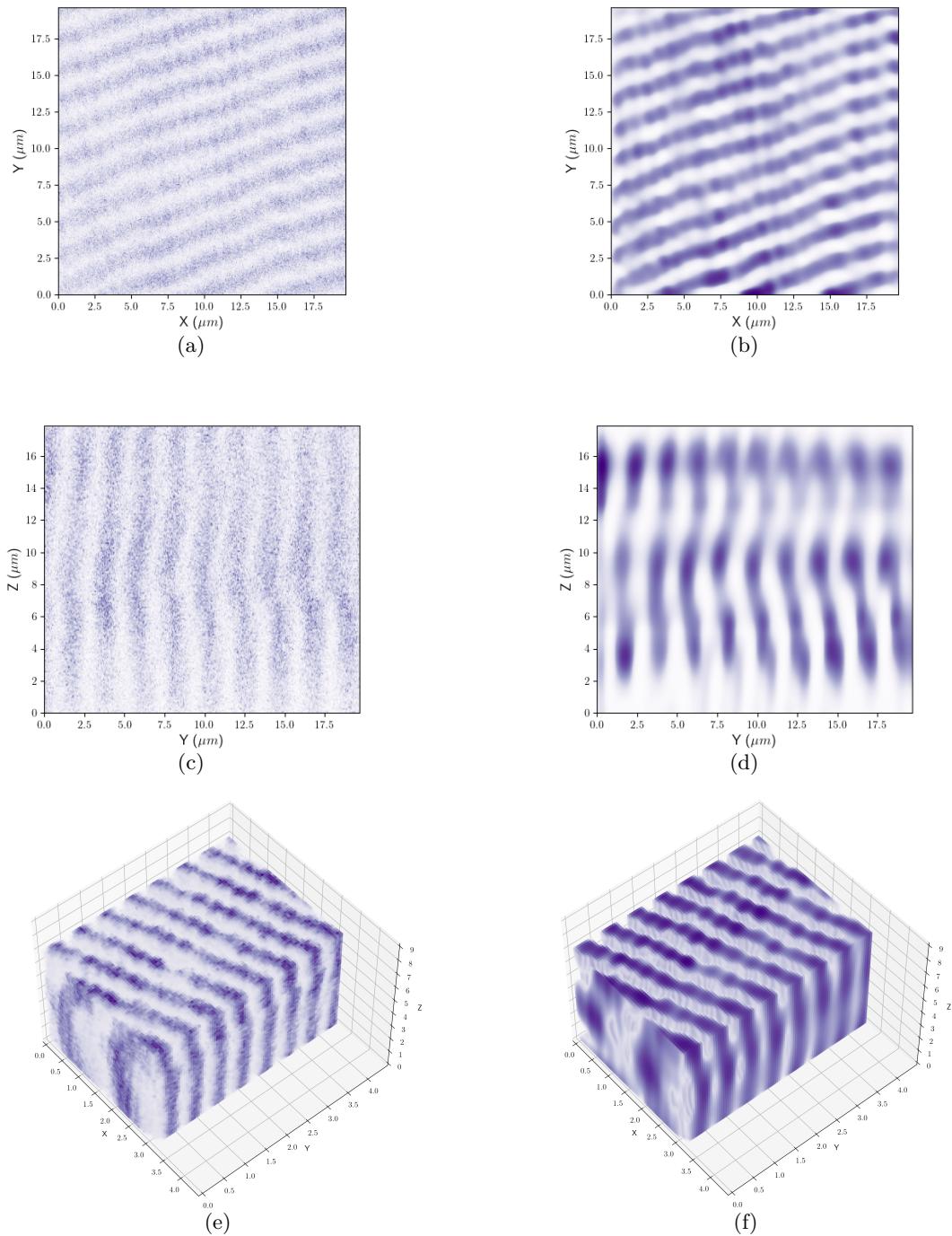


Figure 4.9: Raw image (left column) and restored image (right column) of a mouse muscle sample. For both 3D images, we display the same slice in the plane XY (first row) and YZ (second row), as well as a 3D visualization (third row).

Acknowledgements

We would like to thank Véronique Blanquet, Laetitia Magnol et Alexis Parenté for preparing the mouse muscle samples with microbeads.

- Chapter 5 -

Transductive few-shot learning for unbalanced image classification

This chapter is based on the following article:

S. Martin, M. Boudiaf, E. Chouzenoux, J.-C. Pesquet, I. Ben Ayed, Towards practical few-shot query sets: Transductive minimum description length inference, *Neural Information Processing Systems (NeurIPS)*, 2022.

Standard few-shot benchmarks are often built upon simplifying assumptions on the query sets, which may not always hold in practice. In particular, for each task at testing time, the classes effectively present in the unlabeled query set are known a priori, and correspond exactly to the set of classes represented in the labeled support set. We relax these assumptions and extend current benchmarks, so that the query-set classes of a given task are unknown, but just belong to a much larger set of possible classes. Our setting could be viewed as an instance of the challenging yet practical problem of extremely imbalanced K -way classification, K being much larger than the values typically used in standard benchmarks, and with potentially irrelevant supervision from the support set. Expectedly, our setting incurs drops in the performances of state-of-the-art methods. Motivated by these observations, we introduce a **P**rim**A**l **D**ual **M**inimum **D**escription **L**ength (**PADDLE**) formulation, which balances data-fitting accuracy and model complexity for a given few-shot task, under supervision constraints from the support set. Our constrained MDL-like objective promotes competition among a large set of possible classes, preserving only effective classes that befit better the data of a few-shot task. It is hyper-parameter free, and could be applied on top of any base-class training. Furthermore, we derive a fast block coordinate descent algorithm for optimizing our objective, with convergence guarantee, and a linear computational complexity at each iteration. Comprehensive experiments over the standard few-shot datasets and the more realistic and challenging *i-Nat* dataset show highly competitive performances of our method, more so when the numbers of possible classes in the tasks increase. Our code is publicly available at <https://github.com/SegoleneMartin/PADDLE>.

5.1 Introduction

The performance of deep learning models is often seriously affected when tackling new tasks with limited supervision, i.e., classes that were unobserved during training and for which we have only a handful of

labeled examples. Few-shot learning [Snell et al., 2017, Finn et al., 2017, Vinyals et al., 2016] focuses on this generalization challenge, which occurs in a breadth of applications.

In standard few-shot settings, a deep network is first trained on a large-scale dataset including labeled instances sampled from an initial set of classes, commonly referred to as the base classes. Subsequently, for novel classes, unobserved during the base training, supervision is restricted to a limited number of labeled instances per class. During the test phase, few-shot methods are evaluated over individual tasks, each including a small batch of unlabeled test samples (*the query set*) and a few labeled instances per novel class (*the support set*).

The transduction approach has becomes increasingly popular in few-shot learning, and a large body of recent methods focused on this setting, including, for instance, those based on graph regularization [Ziko et al., 2020, Liu et al., 2019], optimal transport [Lazarou et al., 2021, Hu et al., 2021], feature transformations [Liu et al., 2020, Cui and Guo, 2021], information maximization [Veilleux et al., 2021, Boudiaf et al., 2020, Boudiaf et al., 2021] and transductive batch normalization[Bronskill et al., 2020, Finn et al., 2017], among other works [Dhillon et al., 2019, Wang et al., 2020, Cui and Guo, 2021, Qi et al., 2021, Shen et al., 2021]. On standard benchmarks, the gap in classification accuracy between transductive and inductive few-shot methods may reach 10%; see [Boudiaf et al., 2020], for example. This connects with a well-known fact in classical literature on transductive inference [Vapnik, 1999, Joachims, 1999], which prescribes transduction as an effective way to mitigate the difficulty inherent to learning from limited labels. It is worth mentioning that transductive methods inherently depend on the statistical properties of the query sets. For instance, the recent studies in [Lichtenstein et al., 2020, Veilleux et al., 2021] showed that variations in the class balance within the query set may affect the performances of transductive methods.

Transductive few-shot classification occurs in a variety of practical scenarios, in which we naturally have access to a batch of unlabeled samples at test time. In commonly used few-shot benchmarks, the query set of each task is small (less than 100 examples), and is sampled from a limited number of classes (typically 5). Those choices are relevant in practice: During test time, one typically has access to small unlabeled batches of potentially correlated (non-i.i.d.) samples, e.g., smart-device photos taken at a given time, video-stream sequences, or in pixel prediction tasks such as semantic segmentation [Boudiaf et al., 2021], where only a handful of classes appear in the testing batch. However, the standard few-shot benchmarks are built upon further assumptions that may not always hold in practice: *the few classes effectively present in the unlabeled query set are assumed both to be known beforehand and to match exactly the set of classes of the labeled support set.* We relax these assumptions, and extend current benchmarks so that the query-set classes are unknown and do not match exactly the support-set classes, but just belong to a much larger set of possible classes. Specifically, we allow the total number of classes represented in the support set to be higher than typical values, while keeping the number of classes that are effectively present in the query set to be much smaller.

Our challenging yet practical setting raises several difficulties for state-of-the-art transductive few-shot classifiers: (i) it is an instance of the problem of highly imbalanced classification; (ii) the labeled support set includes “distraction” classes that may not actually be present in the test query samples; (iii) we consider K -way classification tasks with K much larger than the typical values used in the current benchmarks. We evaluated 7 of the best-performing state-of-the-art transductive few-shot methods and,

expectedly, observed drops in their performance in this challenging setting.

Motivated by these experimental observations, we introduce a Minimum Description Length (MDL) inference, which balances data-fitting accuracy and model complexity for a given few-shot task, subject to supervision constraints from the support set. The model-complexity term can be viewed as a continuous relaxation of a discrete label cost, which penalizes the number of non-empty clusters in the solution, fitting the data of a given task with as few unique labels as necessary. It encourages competition among a large set of possible classes, preserving only those that fit better the task. Our formulation is hyper-parameter free, and could be applied on top of any base-class training. Furthermore, we derive a fast primal-dual block coordinate descent algorithm for optimizing our objective, with convergence guarantee, and a linear computational complexity at each iteration thanks to closed-form updates of the variables. We report comprehensive experiments and ablation studies on *mini*-Imagenet, *tiered*-Imagenet, and the more realistic and challenging iNat dataset [Wertheimer and Hariharan, 2019] for fine-grained classification, with 908 classes and 227 ways at test-time. Our method yields competitive performances in comparison to the state-of-the-art, with gaps increasing substantially with the numbers of possible classes in the tasks.

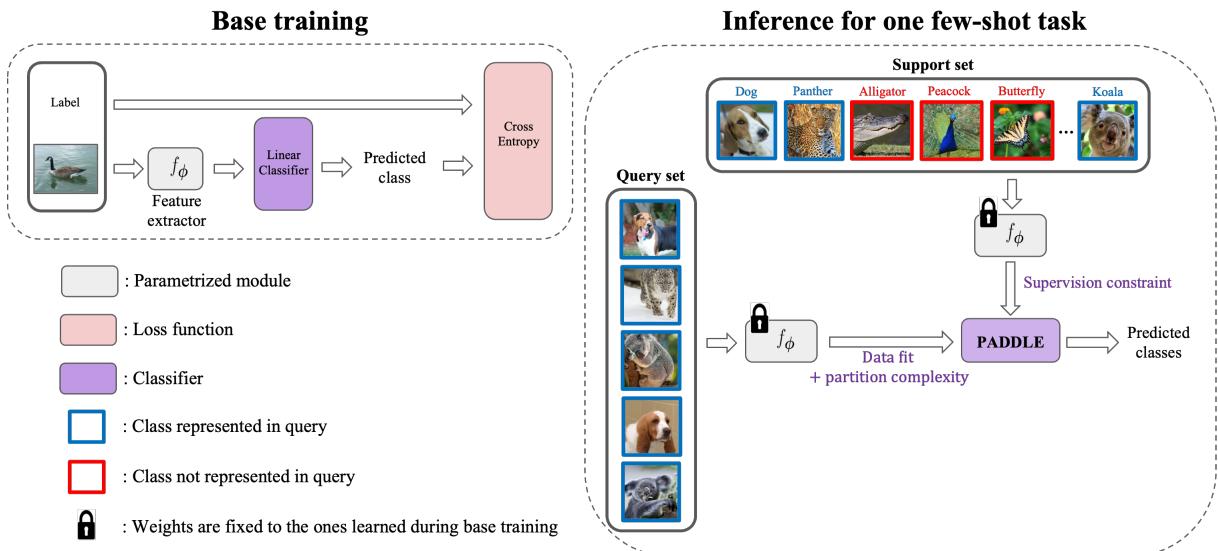


Figure 5.1: Overview of the proposed framework: Base training, PADDLE inference and task generation. The example depicted in the right-hand side illustrates how the support and query classes do not match exactly, unlike in standard few-shot settings. The support set includes “distraction” classes that may not actually be present in the query set, e.g. classes “Alligator”, “Peacock” and “Butterfly”. All possible classes (K classes) are represented in the support set, but only an unknown subset among these K possible classes (K_{eff} effective classes) appear in the query set, with $K_{\text{eff}} \ll K$.

5.2 Proposed few-shot inference formulation

For a given few-shot task, let N represent the total number of images. Define $\mathbb{Q} \subseteq 1, \dots, N$ and $\mathbb{S} \subseteq 1, \dots, N \setminus \mathbb{Q}$ as the indices of the query set and the support set, respectively. It is assumed that the images $(\mathbf{x}_n)_{1 \leq n \leq N}$ are available, along with the labels for the support set $(\mathbf{y}_n)_{n \in \mathbb{S}}$. We denote $\mathbf{z}_n = f_\phi(\mathbf{x}_n)$ the feature vector corresponding to the data sample \mathbf{x}_n , and $\mathbf{u}_n = (u_{n,k})_{1 \leq k \leq K} \in \{0, 1\}^K$ the variable assigning the data to one of the possible classes in $\{1, \dots, K\}$, i.e. $u_{n,k} = 1$ if \mathbf{x}_n belongs to class k and 0 otherwise. We define the variables giving the class proportions $\hat{\mathbf{u}} = (\hat{u}_k)_{1 \leq k \leq K} \in \Delta_K$ as

$$\hat{u}_k = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k} \quad \forall k \in \{1, \dots, K\}, \quad (5.2.1)$$

where Δ_K is the unit simplex of \mathbb{R}^K .

We propose to cast transductive few-shot inference as the minimization of an objective balancing data-fitting accuracy and partition complexity, subject to supervision constraints (known labels) from the support set \mathbb{S} . Our approach estimates jointly the class assignments $\mathbf{U} = (\mathbf{u}_n)_{1 \leq n \leq N}$ and the class prototypes $\mathbf{W} = (\mathbf{w}_k)_{1 \leq k \leq K} \in (\mathbb{R}^d)^K$ through the following problem:

$$\begin{aligned} & \underset{\mathbf{U}, \mathbf{W}}{\text{minimize}} \quad \underbrace{\frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2}_{\text{data-fitting accuracy}} - \lambda \underbrace{\sum_{k=1}^K \hat{u}_k \ln(\hat{u}_k)}_{\text{partition complexity}}, \\ & \text{s.t.} \quad \mathbf{u}_n \in \Delta_K \quad \forall n \in \mathbb{Q}, \\ & \quad \mathbf{u}_n = \mathbf{y}_n \quad \forall n \in \mathbb{S}. \end{aligned} \quad (C) \quad (5.2.2)$$

Note that, in the second line of (5.2.2), we have relaxed the integer constraints on query assignments to facilitate optimization.

Effect of each term

The purpose of objective (5.2.2) is to classify the data of a few-shot task with as few unique labels as necessary. The data-fitting term has the same form as the standard K -means objective for clustering, but is constrained with supervision from the support-set labels. This term evaluates, within each class k , the deviation of features from class prototype \mathbf{w}_k , thereby encouraging consistency of samples of the same class. The partition-complexity term implicitly penalizes the number of effective (non-empty) classes that appear in the solution ($K_{\text{eff}} \leq K$), encouraging low cardinality partitions of query set. This term is the Shannon entropy of class proportions within \mathbb{Q} : it reaches its minimum when all the samples of \mathbb{Q} belong to a single class, i.e., $\exists j \in \{1, \dots, K\}$ such that $\hat{u}_j = 1$ and all other proportions vanish. It achieves its maximum for perfectly balanced partitions of \mathbb{Q} , satisfying $\hat{u}_k = 1/K$ for all k . In practice, this terms promotes solutions that contain only a handful of effective classes among a larger set of K possible classes.

Connection to Minimum Description Length (MDL)

The objective in (5.2.2) could be viewed as a partially-supervised instantiation of the general MDL principle. Originated in information theory, MDL is widely used in statistical model selection [MacKay, 2003]. Assume that we want to describe some input data $\mathbf{Z} = (\mathbf{z}_n)_{n \in \mathbb{Q}}$ with a statistical model \mathbf{M} , among a family of possible models. MDL prescribes that the best model corresponds to the *shortest* description of the data, according to some coding scheme underlying the model. It balances data-fitting accuracy and model complexity by minimizing $\mathcal{L}(\mathbf{Z}|\mathbf{M}) + \lambda\mathcal{L}(\mathbf{M})$ w.r.t \mathbf{M} . $\mathcal{L}(\mathbf{Z}|\mathbf{M})$ measures the *code length* of the prediction of \mathbf{Z} made by \mathbf{M} , while $\mathcal{L}(\mathbf{M})$ encourages simpler models (Occam's razor [MacKay, 2003]), e.g., models with less parameters. For instance, in unsupervised clustering, it is common to minimize a discrete label cost as a measure of model complexity [Delong et al., 2012, Yuan and Boykov, 2010, Zhu and Yuille, 1996]. Such a label cost is the number of effective (non-empty) clusters in the solution, and is used in conjunction with a log-likelihood term for data-fitting:

$$\mathcal{L}(\mathbf{M}) = \sum_{k=1}^K \mathbf{1}_{\hat{u}_k \neq 0} \quad (5.2.3)$$

$$\mathcal{L}(\mathbf{Z}|\mathbf{M}) = -\sum_{k=1}^K \sum_{n \in \mathbb{Q}} u_{n,k} \ln p(\mathbf{z}_n|k; \mathbf{w}_k) \quad (5.2.4)$$

In this expression, $\mathbf{M} = \{(\mathbf{w}_k)_{1 \leq k \leq K}, \mathbf{U}\}$ with \mathbf{w}_k being the parameters of some probability distribution $p(\mathbf{z}_n|k; \mathbf{w}_k)$ describing the samples in cluster k , while \mathbf{U} contains assignments as denoted above.

The discrete model complexity in (5.2.3) is commonly used in MDL-based clustering [Delong et al., 2012, Yuan and Boykov, 2010, Zhu and Yuille, 1996], despite the ensuing optimization difficulty. As this measure is a discrete count of non-empty clusters, it does not accommodate fast optimization techniques. Typically, it is either handled via cluster merging heuristics (starting from an initial set of clusters) [Zhu and Yuille, 1996] or via combinatorial move-making algorithms [Delong et al., 2012], both of which are computationally intensive, more so when dealing with large sets of classes.

Our partition-complexity term in (5.2.2) can be viewed as a *continuous relaxation* of the discrete label count in (5.2.3). We provide in Figure 5.2 simple graphical illustration in the case when $K = 2$. While penalizing similarly the number of effective classes, it has several advantages over the label count in (5.2.3), despite the surprising fact that it is not common in the MDL-based clustering literature, to our best knowledge. First, it is a continuous function of assignment variables \mathbf{U} . This enables us to derive a fast primal-dual block coordinate descent algorithm (Section 5.3), with convergence guarantee, and a linear computational complexity at each iteration thanks to closed-form updates of the variables. Second, it has a clear MDL interpretation: it measures the number of bits required to encode the set of classes, given class probabilities $p(k) = \hat{u}_k, \forall k \in \{1, \dots, K\}$.

Following the Kraft-McMillan theorem [MacKay, 2003], any probability distribution $p(\mathbf{z}_n|k; \mathbf{w}_k)$ corresponds to some coding scheme for storing the features of cluster k , and $-\ln p(\mathbf{z}_n|k; \mathbf{w}_k)$ is the number of bits required to represent any feature using coding scheme $p(\mathbf{z}_n|k; \mathbf{w}_k)$. Clearly, our data-fitting term in (5.2.2) also fits into this MDL interpretation by assuming that each probability $p(\mathbf{z}_n|k; \mathbf{w}_k)$ is a Gaussian

distribution with mean \mathbf{w}_k and covariance fixed to the identity matrix:

$$p(\mathbf{z}_n|k; \mathbf{w}_k) \propto \exp\left(-\frac{1}{2}\|\mathbf{w}_k - \mathbf{z}_n\|^2\right) \quad \forall k \in \{1, \dots, K\}. \quad (5.2.5)$$

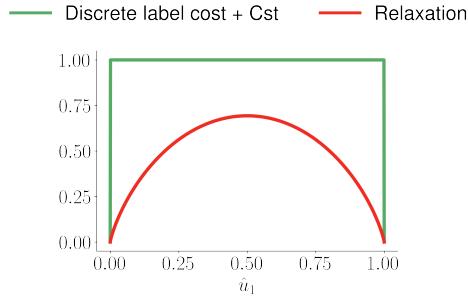


Figure 5.2: Label cost as a function of \hat{u}_1 and our proposed relaxation $\hat{u}_1 \mapsto -\hat{u}_1 \ln(\hat{u}_1) - (1 - \hat{u}_1) \ln(1 - \hat{u}_1)$.

Differences with the existing objectives for transductive few-shot inference

Unlike most of the existing transductive few-shot methods, our MDL formulation does not encode strong assumptions on the label statistics of the query set (e.g. class balance or/and a perfect match between the query and support classes). Information maximization methods, such as [Veilleux et al., 2021, Boudiaf et al., 2020], maximise the confidence of the sample-wise predictions while encouraging class balance. Optimal transport methods [Lazarou et al., 2021, Hu et al., 2021] estimate an optimal mapping matrix, which could be viewed as a joint probability distribution over the features and the labels, while imposing a hard class-balance constraint via the Sinkhorn-Knopp algorithm. Both information-maximization and optimal-transport methods have inherent class-balance bias and, as shown in the experiments, undergo a drastic drop in accuracy when the number of possible classes increases (as this corresponds to highly imbalanced problems). Prototype rectification [Liu et al., 2020] transforms the query features so as to minimize the difference between the overall statistics of the query and support sets. This relies on the assumption that the support and query classes match perfectly. Therefore, it is not adapted to our setting. Inspired from graphical models, the Laplacian regularization in [Ziko et al., 2020] is a pairwise label correction (rather than a learning) method, which encourages assigning the same label to query samples that are close in the input space; it does not learn a class representation from the query set (the prototypes being fixed as those of the support set).

Identifying λ through an unbiased probabilistic K-means interpretation of (5.2.2)

Another interesting view of our few-shot inference objective in (5.2.2) can be drawn from the probabilistic K -means objective¹, well-known in the clustering literature [Kearns et al., 1998, Tang et al., 2019, Boykov et al., 2015]. In fact, probabilistic K -means corresponds to minimizing $\mathcal{L}(\mathbf{Z}|\mathbf{M})$ in (5.2.4) as a generalization of K -means, which corresponds to the particular Gaussian choice in (5.2.5). It is

¹In the general context of clustering, the name probabilistic K -means was first coined by [Kearns et al., 1998].

a well-known that probabilistic K -means has a strong bias towards balanced partitions [Kearns et al., 1998, Boykov et al., 2015]. The objective can be decomposed to reveal a hidden term promoting class balance. Using Bayes rule $p(k|\mathbf{z}_n; \mathbf{w}_k) \propto p(\mathbf{z}_n|k; \mathbf{w}_k)p(k)$, empirical estimates of marginal class probabilities $p(k) = \hat{u}_k$, and the choice of $p(\mathbf{z}_n|k; \mathbf{w}_k)$ in (5.2.5), we have:

$$\begin{aligned} \frac{1}{2} \sum_{k=1}^K \sum_{n \in \mathbb{Q}} u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 &= - \sum_{k=1}^K \sum_{n \in \mathbb{Q}} u_{n,k} \ln p(\mathbf{z}_n|k; \mathbf{w}_k), \\ &\stackrel{c}{=} - \sum_{k=1}^K \sum_{n \in \mathbb{Q}} u_{n,k} \ln p(k|\mathbf{z}_n; \mathbf{w}_k) + |\mathbb{Q}| \sum_{k=1}^K \hat{u}_k \ln(\hat{u}_k), \end{aligned} \quad (5.2.6)$$

where $\stackrel{c}{=}$ stands for equality, up to an additive constant independent of optimization variables. Minimizing the last term in the second line of (5.2.6) has an effect opposite to the model-complexity term in our objective in (5.2.2): it reaches its minimum for perfectly balanced partitions. Therefore, our objective in (5.2.2) is a way to mitigate such a bias in K -means, allowing imbalanced partitions. This suggests setting $\lambda = |\mathbb{Q}|$ to compensate for the hidden class-balance term in K -means. This would make our inference hyper-parameter free. Thus, for our study, we fixed $\lambda = |\mathbb{Q}|$ in all benchmarks, without optimizing the parameter λ via validation.

5.3 Primal-Dual Block-Coordinate Descent Optimization

We derive a block coordinate descent algorithm for our formulation in (5.2.2). At each iteration of our algorithm, the minimization steps are closed-form, with a linear complexity in K and N . We emphasize that, in problem (5.2.2), the minimization over \mathbf{U} could not be carried in closed-form, even for fixed prototypes \mathbf{W} , and the simplex constraints are difficult to handle. One straightforward iterative solution to tackle (5.2.2) would be to deploy a projected gradient descent algorithm. However, as shown in the comparisons in Section 5.4, this strategy is computationally demanding and does not yield better classification results than the algorithm we propose below.

Primal-Dual formulation

We transform problem (5.2.2) into an equivalent optimization problem by introducing a dual variable. Let us first define the linear operator $\mathbf{A}: (\mathbb{R}^K)^N \rightarrow \mathbb{R}^K$, which maps the probabilities to the proportions of the classes, i.e. $\mathbf{A}: \mathbf{U} \mapsto (\hat{u}_k)_{1 \leq k \leq K}$. Then, problem (5.2.2) can be re-written as follows:

$$\underset{\mathbf{U}, \mathbf{W}}{\text{minimize}} \quad \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 - \lambda H(\mathbf{AU}), \quad \text{s.t. } (\mathbf{C})$$

where H is the negative entropy function on the positive orthant of \mathbb{R}^K : $H(\mathbf{x}) = \sum_{k=1}^K \varphi(x_k)$ for $\mathbf{x} = (x_k)_{1 \leq k \leq K} \in \mathbb{R}^K$, with

$$\varphi(t) = \begin{cases} t \ln(t) & \text{if } t > 0, \\ 0 & \text{if } t = 0, \\ +\infty & \text{otherwise.} \end{cases} \quad (5.3.1)$$

We can now appeal to the concept of Fenchel-Legendre conjugate function. For a given convex function f , its conjugate function, denoted by f^* , is defined as $f^*(\mathbf{V}) = \sup_{\mathbf{U}} (\langle \mathbf{V}, \mathbf{U} \rangle - f(\mathbf{U}))$, where \mathbf{V} is a dual variable [Bauschke and Combettes, 2011, Def. 13.1] and $\langle \cdot, \cdot \rangle$ denotes the Euclidean scalar product on $\mathbb{R}^{K \times N}$. Since function H is a proper, lower semicontinuous, convex function, according to the Fenchel-Moreau theorem [Bauschke and Combettes, 2011, Thm. 13.37], the equality $H = (H^*)^*$ holds. Hence, it follows that

$$-\lambda H(\mathbf{A}\mathbf{U}) = \lambda \inf_{\mathbf{V}} \{H^*(\mathbf{V}) - \langle \mathbf{V}, \mathbf{A}\mathbf{U} \rangle\}. \quad (5.3.2)$$

Plugging (5.3.2) into (5.3.1), and using the expression of H^* [Boyd et al., 2011, Ex. 3.21], one obtains the following minimization problem with respect to the primal variable \mathbf{U} , the weights \mathbf{W} and the dual variable \mathbf{V} :

$$\underset{\mathbf{U}, \mathbf{W}, \mathbf{V}}{\text{minimize}} \quad \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 + \lambda \sum_{k=1}^K e^{v_k-1} - \lambda \langle \mathbf{V}, \mathbf{A}\mathbf{U} \rangle, \quad \text{s.t. } (C) \quad (5.3.3)$$

Handling the simplex constraint

To deal efficiently with the simplex constraint in (5.3.3), we add an entropic barrier on soft assignment variables \mathbf{u}_n , leading to the following modified problem:

$$\underset{\mathbf{U}, \mathbf{W}, \mathbf{V}}{\text{minimize}} \quad \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 + \lambda \sum_{k=1}^K e^{v_k-1} - \lambda \langle \mathbf{V}, \mathbf{A}\mathbf{U} \rangle + \underbrace{\sum_{n=1}^N \sum_{k=1}^K \varphi(u_{n,k})}_{\text{entropic barrier}}, \quad (5.3.4)$$

s.t. (C).

The last term in (5.3.4) acts as a barrier for imposing constraints $\mathbf{u}_n \geq 0$ and, at each iteration, yields closed-form updates of both the dual variables for constraints $\sum_{k=1}^K u_{n,k} = 1$ and assignments \mathbf{U} . This simplifies the iterative constrained-minimization steps over \mathbf{U} in our algorithm below through simple closed-form softmax operations.

Block coordinate descent algorithm

To minimize the cost function in (5.3.4), we pursue a block coordinate descent approach [Bertsekas, 1995, Xu and Yin, 2013], which is guaranteed to converge sequentially; see Proposition 5.1 below. At each iteration, we successively minimize the objective with respect to the variables \mathbf{U} , \mathbf{V} , and \mathbf{W} respectively. To do so, we use the adjoint operator of \mathbf{A} , which we denote \mathbf{A}^* . Our algorithm is detailed in Algorithm 13.

Algorithm 13: PrimAl Dual Minimum Description LEngh (PADDLE)

Initialize $\mathbf{W}^{(0)}$ as the prototypes computed on the support, and $\mathbf{V}^{(0)} = \mathbf{0}$.

for $\ell = 1, 2, \dots$, **do**

$$\left\{ \begin{array}{l} \mathbf{U}^{(\ell)} = \text{softmax} \left(-\frac{1}{2} \left(\|\mathbf{w}_k - \mathbf{z}_n\|^2 \right)_{\substack{1 \leq n \leq N \\ 1 \leq k \leq K}} + \lambda \mathbf{A}^* \mathbf{V}^{(\ell-1)} \right), \\ v_k^{(\ell)} = 1 + \ln((\mathbf{A} \mathbf{U}^{(\ell)})_k), \forall k \in \{1, \dots, K\}, \\ \mathbf{w}_k^{(\ell)} = \sum_{n=1}^N \mathbf{u}_{n,k}^{(\ell-1)} \mathbf{z}_n / \sum_{n=1}^N \mathbf{u}_{n,k}^{(\ell-1)}, \forall k \in \{1, \dots, K\}. \end{array} \right.$$

Details on the steps in the algorithm

We provide more details on the derivation of the closed-form update of variable \mathbf{U} at each iteration. Let F be the defined as the cost function in (5.3.4) and let $\partial F_{\mathbf{u}_n}(\mathbf{U}, \mathbf{W}, \mathbf{V})$ denote the Moreau subdifferential of F at $(\mathbf{U}, \mathbf{W}, \mathbf{V})$ with respect to variable \mathbf{u}_n . We define ψ as

$$(\forall \mathbf{x} = (x_k)_{1 \leq k \leq K} \in \mathbb{R}^K) \quad \psi(\mathbf{x}) = \begin{cases} \sum_{k=1}^K x_k \ln(x_k) - \frac{x_k^2}{2} & \text{if } \mathbf{x} \in \Delta_K, \\ +\infty & \text{otherwise.} \end{cases} \quad (5.3.5)$$

It is well known that the proximity operator of ψ (see [Bauschke and Combettes, 2011, Chap. 24] for a definition) is the softmax operator [Combettes and Pesquet, 2020, Ex. 2.23].

At each step of the algorithm, \mathbf{u}_n is updated according to:

$$\begin{aligned} 0 &\in \partial F_{\mathbf{u}_n}(\mathbf{U}, \mathbf{W}, \mathbf{V}) \\ \iff 0 &\in \frac{1}{2} \left(\|\mathbf{w}_k - \mathbf{z}_n\|^2 \right)_{1 \leq k \leq K} - \lambda [\mathbf{A}^* \mathbf{V}]_n + \mathbf{u}_n + \partial_\psi(\mathbf{u}_n), \\ \iff -\frac{1}{2} \left(\|\mathbf{w}_k - \mathbf{z}_n\|^2 \right)_{1 \leq k \leq K} + \lambda [\mathbf{A}^* \mathbf{V}]_n - \mathbf{u}_n &\in \partial_\psi(\mathbf{u}_n), \\ \iff \mathbf{u}_n &= \text{softmax} \left(-\frac{1}{2} \left(\|\mathbf{w}_k - \mathbf{z}_n\|^2 \right)_{1 \leq k \leq K} + \lambda [\mathbf{A}^* \mathbf{V}]_n \right), \end{aligned} \quad (5.3.6)$$

where we used the definition of the proximity operator [Bauschke and Combettes, 2011, Eq. 24.2] to obtain (5.3.6). We thus retrieve the update in Algorithm 13.

Convergence guarantees

In addition to its practical advantages in terms of implementation, our algorithm benefits from the convergence guarantee described in Proposition 5.1.

Proposition 5.1. *The sequence $(\mathbf{U}^{(\ell)}, \mathbf{W}^{(\ell)}, \mathbf{V}^{(\ell)})_{\ell \in \mathbb{N}^*}$ generated by Algorithm 13 is bounded. Moreover, any of its cluster points is a critical point to the minimization problem in (5.3.4).*

A detailed proof of Proposition 5.1 is provided in Appendix 5.A.1.

5.4 Experiments

5.4.1 Experimental details

Datasets

We deployed three datasets for few-shot classification: *mini*-Imagenet [Russakovsky et al., 2015], *tiered*-Imagenet [Ren et al., 2018], and *i-Nat* [Wertheimer and Hariharan, 2019]. The *mini*-Imagenet is a standard few-shot benchmark, with 60,000 color images of size 84×84 pixels [Vinyals et al., 2016]. It contains 100 classes, each represented with 600 images. We followed the standard split of 64 classes for base training, 16 for validation, and 20 for testing [Ravi and Larochelle, 2016, Wang et al., 2019]. The *tiered*-Imagenet is another standard few-shot benchmark, which is a larger subset of ILSVRC-12, with 608 classes and a total of 779,165 color images of size 84×84 pixels. We used a standard split of 351 classes for base training, 97 for validation, and 160 for testing. Finally, the more realistic and challenging dataset *i-Nat* has 908 classes. We follow the split from [Wang et al., 2019, Ziko et al., 2020], with 227 ways at test-time.

Task generation

We build the few-shot tasks as follows. Let s denotes the number of shots. We constitute the support set by sampling s images according to the uniform distribution for each of the K possible classes of the test set (i.e. $K = |\mathcal{Y}_{\text{test}}| = 20, 160$, and 227 for *mini*-ImageNet, *tiered*-ImageNet and *i-Nat*, respectively). For the query set, we first randomly pick $K_{\text{eff}} < K$ classes among the K possible classes. We then randomly choose $|\mathbb{Q}|$ samples among the images that belong to the K_{eff} classes but do not appear in the support set. The right-hand side of Figure 5.1 depicts an example of a task, with the class color coding indicating whether a support class appears in the query set or not. The results presented below are obtained by fixing $|\mathbb{Q}| = 75$, as done in the literature. In the tables, we use a fixed $K_{\text{eff}} = 5$, but we also present the results over a larger range in Figure 5.3 (ResNet18) and 5.4 (WRN28-10). Given the difficulty of K -way tasks, with K large, we consider 5-, 10- and 20-shot supervision to form the support set. Following standard evaluation, we report the accuracy, averaged across 10,000 tasks.

Hyper-parameters

We emphasize that PADDLE inference does not require any hyperparameter tuning since λ in objective (5.3.4) is set to the number of samples in the query set \mathbb{Q} . To provide further insight into the selection of λ , we have conducted additional experiments, the details of which can be found in Appendix 5.A.2. Other few-shot methods typically have an hyper-parameter. For the tuning phase of those methods, we have followed the protocol of [Veilleux et al., 2021], which for each dataset, uses a 5-way, 5-shot scenario on the corresponding validation set. For *i-Nat*, because no validation split is provided, we reuse the hyper-parameters obtained from *tiered*-ImageNet.

Feature extraction

To ensure the fairest comparison of methods, we use the pretrained checkpoints provided by the authors of [Boudiaf et al., 2020]. The models are trained on $\mathcal{D}_{\text{base}}$ via a standard cross-entropy minimization

with label smoothing. The label smoothing parameter is set to 0.1, for 90 epochs, using a learning rate initialized to 0.1 and divided by 10 at epochs 45 and 66. We use batch sizes of 256 for ResNet-18 and of 128 for WRN28-10. The images are resized to 84×84 pixels, both at training and evaluation time. Color jittering, random cropping, and random horizontal flipping augmentations are applied during training.

Method	Backbone	<i>mini</i> -ImageNet ($K = 20$)			<i>tiered</i> -ImageNet ($K = 160$)		
		5-shot	10-shot	20-shot	5-shot	10-shot	20-shot
Baseline [Chen et al., 2019]	ResNet-18	54.1	60.7	65.8	29.1	35.7	39.5
LR+ICI [Wang et al., 2020]		54.7	62.0	67.2	-	-	-
BD-CSPN [Liu et al., 2020]		49.6	54.1	55.6	24.0	26.7	26.0
PT-MAP [Hu et al., 2021]		25.8	27.4	29.0	4.3	5.1	5.9
LaplacianShot [Ziko et al., 2020]		60.4	65.2	68.4	34.7	36.8	39.1
TIM [Boudiaf et al., 2020]		66.4	69.4	70.9	26.6	26.7	25.5
α -TIM [Veilleux et al., 2021]		63.5	67.4	71.7	38.7	44.2	48.4
PADDLE (ours)		63.2	73.3	80.0	45.8	62.5	72.0
Baseline [Chen et al., 2019]	WRN28-10	58.0	64.8	69.5	31.8	37.0	42.1
LR+ICI [Wang et al., 2020]		57.2	64.3	70.9	-	-	-
BD-CSPN [Liu et al., 2020]		51.2	55.5	58.4	23.7	24.9	23.8
PT-MAP [Hu et al., 2021]		26.3	27.9	29.4	4.4	5.0	5.7
LaplacianShot [Ziko et al., 2020]		64.9	65.3	70.9	28.1	38.0	45.2
TIM [Boudiaf et al., 2020]		71.9	75.2	76.1	34.1	34.1	34.5
α -TIM [Veilleux et al., 2021]		68.3	72.5	75.8	41.9	46.1	51.8
PADDLE (ours)		62.5	72.8	79.3	46.4	60.3	71.1

Table 5.1: Comparisons of state-of-the-art methods on *mini*-Imagenet and *tiered*-Imagenet, using the tasks generation process described in Sec. 5.4.1 with $K_{\text{eff}} = 5$. The metric is accuracy (in percentage). Results are averaged across 10,000 tasks. Results marked with '-' were intractable to obtain.

5.4.2 Results

Main results

We compare the performances of our method with state-of-the-art few-shot methods. Our first experimental setting consists in fixing the number of effective classes K_{eff} to 5. In Table 5.1, we evaluate the accuracy on *mini* and *tiered*-ImageNet for 5, 10, and 20 shots, while results on *i-Nat* are displayed in Table 5.2. Note that the *i-Nat* dataset comes with a unique support set for all tasks, with a varying number of shots (labeled samples) per class. Therefore, for *i-Nat*, we present the results separately. In the second experiment displayed in Figure 5.3, 5.5, and 5.4, we plot the accuracy as a function of K_{eff} on *mini*, *tiered*-ImageNet, and *i-Nat* for 5, 10, and 20 shots. Both experiments show that PADDLE is very competitive with the state-of-the-art methods on the proposed new practical few-shot setting. More importantly, the gap between PADDLE and the second best algorithm increases significantly with (i) the number of shots and (ii) the number of possible classes K , when going from the dataset *mini*, which has a small number of test classes ($K = 20$), to *tiered* ($K = 160$) and finally to *i-Nat* ($K = 227$). From the first point, one may conclude that the proposed MDL formulation benefits better from additional support

supervision. We hypothesize this is due to its versatility, as it does not encode strong assumptions on the label statistics of the query set. Also, the second point could be explained along the same line. For instance, methods relying on a class-balance assumptions, such as TIM-GD or PT-MAP, observe a drastic drop in accuracy as K increases, even falling below the inductive Baseline. This is expected because large values of K corresponds to highly imbalanced classification problems for the query sets. For example, $K_{\text{eff}} = 5$ effective classes on *tiered* corresponds to a relatively more drastic form of class imbalance than on *mini*, as it would mean only 5/160 classes are represented in \mathbb{Q} , versus 5/20. In contrast, PADDLE can deal with strongly imbalanced situations. Note that BD-CSPN [Liu et al., 2020] is the only baseline whose performance increases with K_{eff} . This method is also among the most affected when the class overlap between the query and support sets decreases, i.e., when K_{eff} gets smaller (left side of the plots in Figure 5.3) and/or K gets bigger (e.g. tiered-ImageNet, with $K = 160$). This behaviour might be due to the fact that BD-CSPN encodes a strong prior, assuming the support and query classes match perfectly.

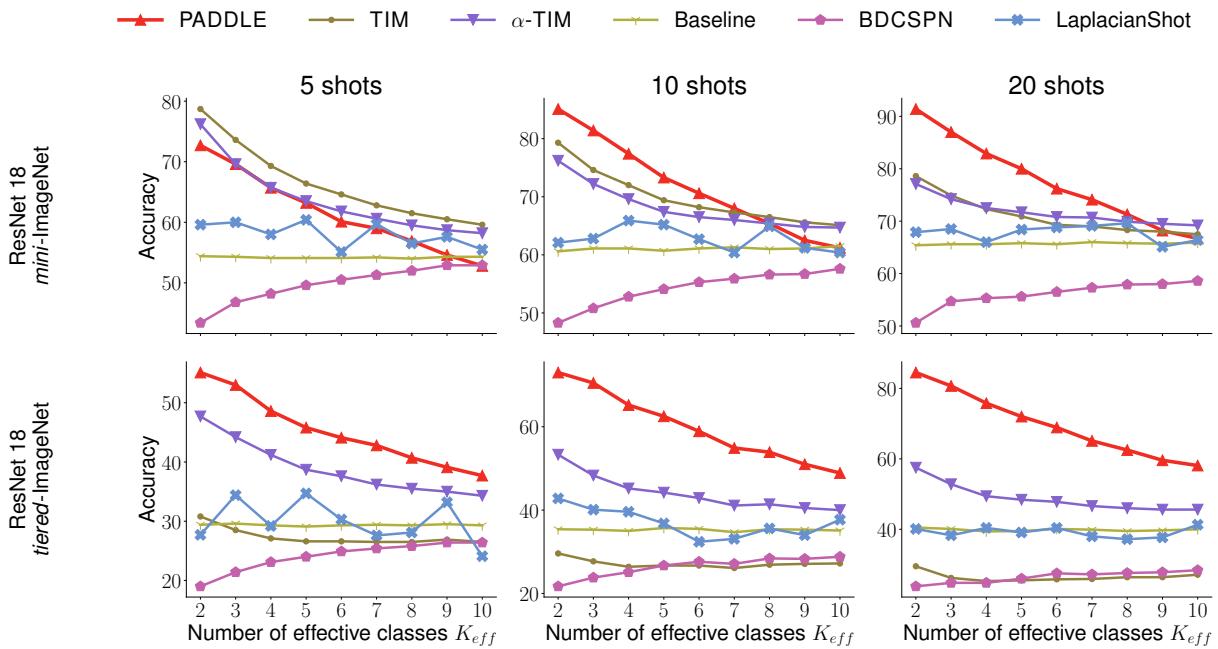


Figure 5.3: Evolution of the accuracy as a function of K_{eff} . Each row represents a dataset, and each column a fixed number of shots. All methods use the same ResNet-18 network. Results are averaged across 10,000 tasks.

Ablation on the objective

We hereby ablate on the importance of the *partition-complexity* term in Eq. (5.2.2). Removing this high-order term yields a partially-supervised version of K -means, which can be optimized effortlessly through iterative closed-form alternating steps. We provide the comparison between this approach (*without* the partition-complexity term) and PADDLE (*with* the partition complexity term) in Table 5.3. One can observe that PADDLE systematically outperforms its K -Means counterpart, with absolute differences in accuracy reaching up to 30% + on challenging scenarios.

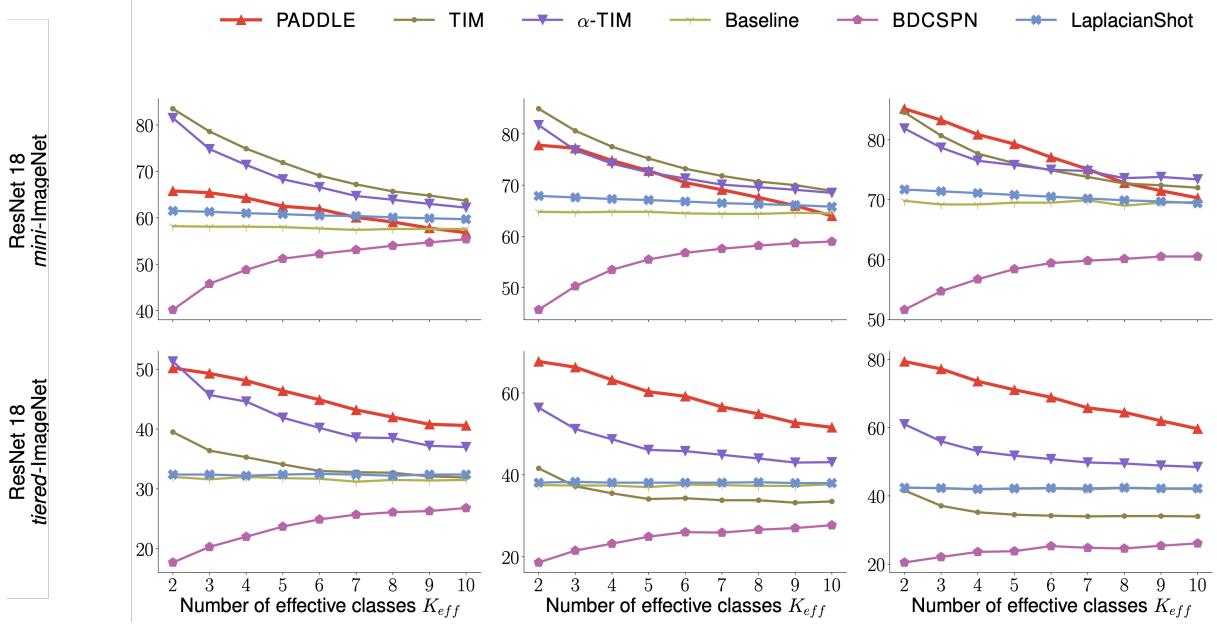


Figure 5.4: Evolution of the accuracy as a function of K_{eff} . Each row represents a dataset, and each column a fixed number of shots. All methods use the same WRN28-10 network. Results are averaged across 10,000 tasks.

Ablation on the optimization procedure

As a second ablation, we propose to compare our proposed Alternating Minimization Algorithm 14 to a straightforward first-order approach. More precisely, as a comparison, we directly optimize Eq. (5.2.2) through Projected Gradient Descent, denoted as PGD, where a simplex projection step [Bauckhage, 2020] takes place after every iteration to ensure that the simplex constraints on \mathbf{U} remains satisfied. For PGD, we use Adam [Kingma and Ba, 2014] with $\alpha = 0.001$. We found this to be the highest learning rate that leads to convergence. We also add the convergence time of the second best competing method, α -TIM. Results are provided in Figure 5.6. PGD is much slower than PADDLE, with a ratio between run times neighboring an order of magnitude. Additionally, as hinted from the last points of the PGD curve, oscillations seem to occur, indicating that a more sophisticated learning rate policy would

Method	<i>i-Nat</i> ($K = 227$)
Baseline [Chen et al., 2019]	58.2
BD-CSPN [Liu et al., 2020]	57.6
PT-MAP [Hu et al., 2021]	6.8
LaplacianShot [Ziko et al., 2020]	43.3
TIM [Boudiaf et al., 2020]	37.5
α -TIM [Veilleux et al., 2021]	66.7
PADDLE (ours)	84.3

Table 5.2: Similarly to Table 5.1, results are provided with a fixed $K_{eff} = 5$ on *i-Nat* with 227-ways tasks.

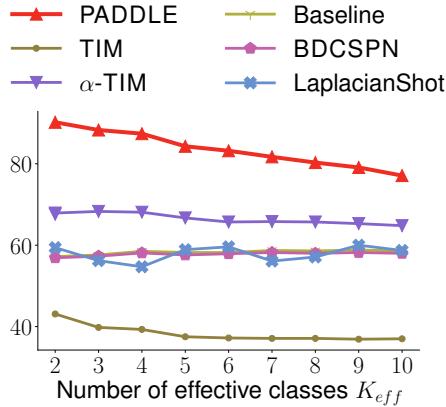


Figure 5.5: Similarly to Figure 5.3, we plot the performances of the methods as functions of K_{eff} on i -Nat with 227-ways tasks, using a ResNet18.

Partition complexity term in Problem 5.2.2	<i>mini</i> ($K = 20$)			<i>tiered</i> ($K = 160$)			<i>i-Nat</i> ($K = 227$)
	5-shot	10-shot	20-shot	5-shot	10-shot	20-shot	
Without	49.6	57.2	63.6	28.3	35.8	39.5	57.5
With	63.2	73.3	80.0	45.8	62.5	72.0	84.3

Table 5.3: Importance of the partition-complexity term. Results are computed using a ResNet-18 and $K_{eff} = 5$, and averaged across 10,000 tasks. Without the *partition-complexity* term in Eq. (5.2.2), the algorithm has little incentive to be parsimonious in its choice of the effective classes. As a matter of fact, removing this term reduces Problem (5.2.2) to a partially-supervised K-means algorithm, notoriously known to encourage balanced solutions across the classes.

be necessary to achieve a clear convergence.

5.5 Conclusion and limitations

We presented a practical few-shot setting where the number of candidate classes can be much larger than the number of classes that appear effectively in the query set. Our setting is an instance of highly imbalanced classification, with large numbers of ways and potentially irrelevant supervision from the support set. We observed much higher gaps across transductive few-shot methods, some of which fall below the simple inductive baseline. As a solution, we introduced PADDLE, which casts this challenge as a partially-supervised MDL partitioning problem, interpreting the number of unique classes found as a measure of model complexity. PADDLE is hyperparameter-free, and remains competitive over various state-of-the-art methods, settings and datasets, without the need for any tuning. However, we do not advocate our method as the one-fits-all, ultimate solution to the extremely challenging few-shot problem. For instance, PADDLE would not be the best performing method in situations where K_{eff} would approach K . In applications where one has knowledge about the label statistics of the query set (e.g. class balance), other methods encoding such a knowledge could be more appropriate. More generally, our setting can be seen as a particular case of class-distribution shift between the support and query sets. Interesting

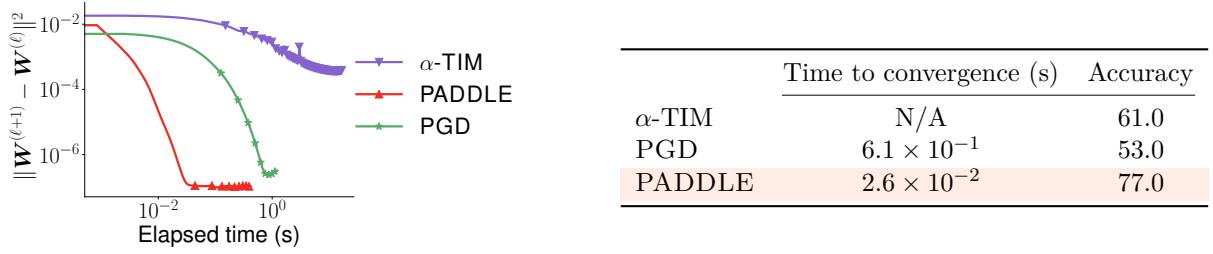


Figure 5.6: Comparison of PADDLE and PGD in terms of convergence speed. (Left) The chosen criterion as a function of the elapsed time on a randomly chosen 20-shot, $K_{\text{eff}} = 5$ task sampled from *tiered*-ImageNet. Both methods are run on the same machine, with markers displayed every 100 iterations. (Right) time to convergence and accuracy, after reaching a value of 10^{-6} on the criterion.

future works could complement our current study by overlaying other forms of shifts, e.g. feature shifts [Guo et al., 2020, Bennequin et al., 2021], to our setting.

5.A Appendices

5.A.1 Proof of Proposition 5.1

Our proof relies on the convergence result established in [Tseng, 2001]. Given a convex set X , we denote ι_X the indicator function of X , i.e. $\iota_X(x) = 0$ if $x \in X$, $\iota_X(x) = +\infty$ otherwise. We rewrite problem 5.3.4 as the minimization of the following cost:

$$F(\mathbf{U}, \mathbf{W}, \mathbf{V}) = \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 + \lambda \sum_{k=1}^K e^{v_k-1} - \lambda \langle \mathbf{V}, (\mathbf{A}\mathbf{U} + \epsilon \mathbf{1}_K) \rangle + \sum_{n=1}^N \sum_{k=1}^K \varphi(u_{n,k}) + \iota_C(\mathbf{U}), \quad (5.A.1)$$

where we have introduced an additional parameter $\epsilon > 0$, the role of which will become clearer in the rest of the proof. The optimum of the cost function $F(\mathbf{U}, \mathbf{W}, \cdot)$ for given $\mathbf{U} \in C$ and $\mathbf{W} \in (\mathbb{R}^d)^K$ is reached when

$$\mathbf{V} = \mathbf{1}_K + \ln(\mathbf{A}\mathbf{U} + \epsilon \mathbf{1}_K) \in \mathbb{V}_\epsilon = [1 + \ln \epsilon, 1 + \ln(1 + \epsilon)]^K. \quad (5.A.2)$$

Thus, minimizing F is actually equivalent to minimizing

$$\tilde{F}(\mathbf{U}, \mathbf{W}, \mathbf{V}) = \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 + \lambda \sum_{k=1}^K e^{v_k-1} - \lambda \langle \mathbf{V}, (\mathbf{A}\mathbf{U} + \epsilon \mathbf{1}_K) \rangle + \sum_{n=1}^N \sum_{k=1}^K \varphi(u_{n,k}) + \iota_C(\mathbf{U}) + \iota_{\mathbb{V}_\epsilon}(\mathbf{V}). \quad (5.A.3)$$

The following algorithm for minimizing \tilde{F} turns out to be a simple modified version of PADDLE (see Algorithm 13):

Algorithm 14: Alternating algorithm for minimizing \tilde{F}

Initialize $\mathbf{W}^{(0)}$ as the prototypes computed on the support, and $\mathbf{V}^{(0)} = \mathbf{0}$.

for $\ell = 1, 2, \dots$, **do**

$$\left| \begin{array}{l} \mathbf{U}^{(\ell)} = \text{softmax} \left(-\frac{1}{2} \left(\|\mathbf{w}_k - \mathbf{z}_n\|^2 \right)_{\substack{1 \leq n \leq N \\ 1 \leq k \leq K}} + \lambda \mathbf{A}^* \mathbf{V}^{(\ell-1)} \right), \\ v_k^{(\ell)} = 1 + \ln((\mathbf{A}\mathbf{U}^{(\ell)})_k + \epsilon), \forall k \in \{1, \dots, K\}, \\ \mathbf{w}_k^{(\ell)} = \sum_{n=1}^N \mathbf{u}_{n,k}^{(\ell-1)} \mathbf{z}_n / \sum_{n=1}^N \mathbf{u}_{n,k}^{(\ell-1)}, \forall k \in \{1, \dots, K\}. \end{array} \right.$$

According to [Tseng, 2001, Thm 4.1], if the following assumptions are satisfied:

1. The set $\{(\mathbf{U}, \mathbf{W}, \mathbf{V}) : \tilde{F}(\mathbf{U}, \mathbf{W}, \mathbf{V}) \leq \tilde{F}(\mathbf{U}^{(0)}, \mathbf{W}^{(0)}, \mathbf{V}^{(0)})\}$ is compact;
2. \tilde{F} is continuous on $C \times (\mathbb{R}^d)^K \times \mathbb{V}_\epsilon$;
3. At each iteration ℓ , the partial functions $\tilde{F}(\cdot, \mathbf{W}^{(\ell)}, \mathbf{V}^{(\ell)})$, $\tilde{F}(\mathbf{U}^{(\ell+1)}, \cdot, \mathbf{V}^{(\ell)})$ and $\tilde{F}(\mathbf{U}^{(\ell+1)}, \mathbf{W}^{(\ell+1)}, \cdot)$ admit a unique minimizer,

then the sequence generated by the algorithm is bounded and every of its cluster points is a coordinatewise minimizer of \tilde{F} . We now show that the above assumptions hold.

1. Let us show that \tilde{F} is coercive. We derive a lower bound on \tilde{F} using the Cauchy-Schwarz inequality:

$$\begin{aligned} \tilde{F}(\mathbf{U}, \mathbf{W}, \mathbf{V}) &\geq \frac{1}{2} \sum_{k=1}^K \sum_{n \in \mathbb{S}} y_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 + \lambda \sum_{k=1}^K e^{v_k-1} - \lambda \|\mathbf{V}\| \|\mathbf{A}\mathbf{U}\| \\ &\quad - \epsilon \langle \mathbf{V}, \mathbf{1}_K \rangle + \sum_{n=1}^N \sum_{k=1}^K \varphi(u_{n,k}) + \iota_C(\mathbf{U}) + \iota_{\mathbb{V}_\epsilon}(\mathbf{V}). \end{aligned} \quad (5.A.4)$$

Since the functions $\mathbf{U} \mapsto \|\mathbf{A}\mathbf{U}\|$ and $\mathbf{U} \mapsto \sum_{n=1}^N \sum_{k=1}^K \varphi(u_{n,k})$ are continuous on the compact set C , there exist constants μ and θ such that

$$\begin{aligned} \tilde{F}(\mathbf{U}, \mathbf{W}, \mathbf{V}) &\geq \frac{1}{2} \sum_{k=1}^K \sum_{n \in \mathbb{S}} y_{n,k} \|\mathbf{w}_k - \mathbf{z}_n\|^2 + \lambda \sum_{k=1}^K e^{v_k-1} - \theta \|\mathbf{V}\| \\ &\quad - \epsilon \langle \mathbf{V}, \mathbf{1}_K \rangle + \mu + \iota_C(\mathbf{U}) + \iota_{\mathbb{V}_\epsilon}(\mathbf{V}). \end{aligned} \quad (5.A.5)$$

The lower bound obtained in (5.A.5) is separable in $(\mathbf{U}, \mathbf{W}, \mathbf{V})$. The term with respect to variable \mathbf{W} is coercive when, for every $k \in \{1, \dots, K\}$, there exists $n \in \mathbb{S}$ such that $y_{n,k} > 0$. In other words, it is coercive if the support set includes at least one example of each class, which is a reasonable assumption. The terms with respect to variables \mathbf{U} and \mathbf{V} are clearly coercive too. Hence, the cost function \tilde{F} is coercive. Finally, since \tilde{F} is lower semi-continuous, condition 1. is satisfied.

2. The continuity of \tilde{F} on $C \times \mathbb{R}^{k \times d} \times \mathbb{V}_\epsilon$ is clear.
3. Let $\ell \in \mathbb{N}^*$. We already proved that the partial function with respect to variable \mathbf{U} has a unique minimizer. It follows from the same arguments as above that the partial function with respect to \mathbf{W} is strictly convex, continuous, and coercive as soon as the support set contains at least one example of each class. Hence, it admits a unique minimizer. Regarding the partial function with

respect to variable \mathbf{V} , we first remark that given the definition of the softmax operator, $\mathbf{AU}^{(\ell+1)}$ is necessarily strictly positive component-wise. Up to some additive term independent of \mathbf{V} , the partial function reads

$$\mathbf{V} \mapsto \lambda \sum_{k=1}^K \left(e^{v_k - 1} - v_k ([\mathbf{AU}^{(\ell+1)}]_k + \epsilon) + \iota_{[\ln \epsilon, \ln(1+\epsilon)]}(v_k - 1) \right). \quad (5.A.6)$$

The latter function is strictly convex, lower-semicontinuous, and coercive, which concludes the proof.

Note that, since

$$v_k \mapsto \lambda \left(e^{v_k - 1} - v_k ([\mathbf{AU}^{(\ell+1)}]_k + \epsilon) \right) \quad (5.A.7)$$

is decreasing on $] -\infty, 1 + \ln([\mathbf{AU}^{(\ell+1)}]_k + \epsilon)]$ and increasing on $[1 + \ln([\mathbf{AU}^{(\ell+1)}]_k + \epsilon), +\infty[$, the resulting cluster points are also coordinatewise minimizers of F .

In summary, PADDLE can be understood as the limit case of Algorithm 14 when ϵ goes to zero. This simplification is justified by the fact that ϵ can be chosen arbitrarily small and that we did not observe any change in practical behaviour of the proposed algorithm by setting $\epsilon = 0$.

5.A.2 About the hyper-parameter in our method

As discussed in Section 5.2, PADDLE does not require parameter tuning. In Figure 5.7, we investigate the optimal value of parameter λ in (5.3.4) as a function of the size of the query set, for 3 different values of K_{eff} . We observe that the optimal value of λ increases linearly with $|Q|$. As it could be expected, the higher the level of class imbalance ($K_{\text{eff}} = 2$), the higher the optimal value of λ (w.r.t. its theoretical value). On the contrary, when the query is better balanced ($K_{\text{eff}} = 10$), the optimal value of λ is slightly under its theoretical value. However, Figure 5.8 shows that the gap of performance when using the theoretical value of λ instead of the optimal one, is only of the order of a few percents.

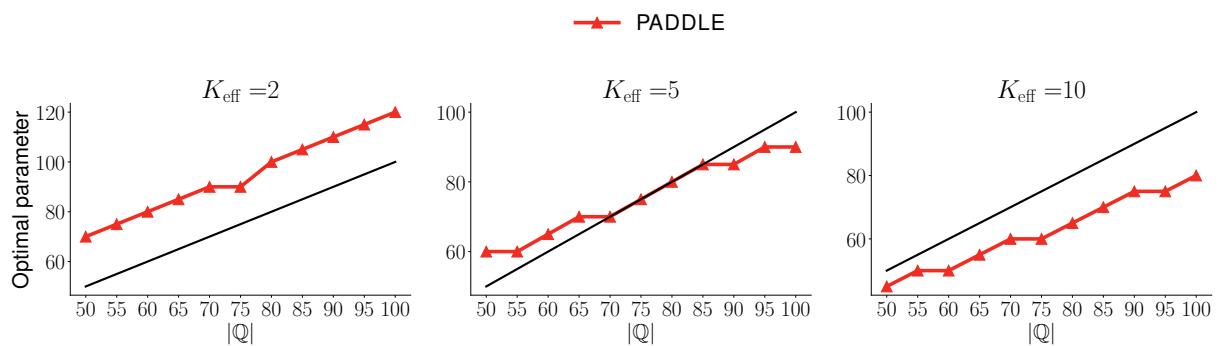


Figure 5.7: Evolution of the optimal parameter λ (i.e. the one with which the best accuracy is reached) as a function of $|Q|$. Each column represents a fixed number of effective classes. The black line represents the identity function. The results were computed on the *tiered* dataset with a Resnet18 as a backbone.

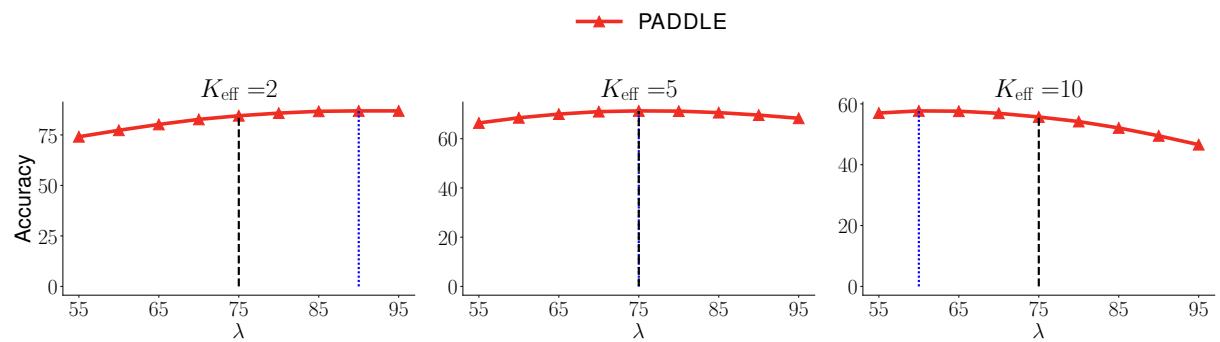


Figure 5.8: Evolution of the accuracy as a function of λ . Each column represents a fixed number of effective classes. The results were computed on the *tiered* dataset with a Resnet18 as a backbone, and the size query set was fixed to $|\mathbb{Q}| = 75$. The blue dotted line represents the optimal value of λ while the black dashed line represents the theoretical value of λ , i.e. $\lambda = |\mathbb{Q}|$.

- Chapter 6 -

Transductive zero-shot and few-shot CLIP

This chapter is based on the following article:

S. Martin, Y. Huang, F. Shakeri, J.-C. Pesquet, I. Ben Ayed, Transductive zero-shot and few-shot CLIP, 2023.

Transductive inference has been widely investigated in few-shot image classification, but completely overlooked in the recent, fast growing literature on adapting vision-language models like CLIP. This chapter addresses the transductive zero-shot and few-shot CLIP classification challenge, in which inference is performed jointly across a mini-batch of unlabeled query samples, rather than treating each instance independently. We initially construct informative vision-text probability features, leading to a classification problem on the unit simplex set. Inspired by Expectation-Maximization (EM), our optimization-based classifying objective models the data probability distribution for each class using a Dirichlet law. The minimization problem is then tackled with a novel block Majorization-Minimization algorithm, which simultaneously estimates the distribution parameters and class assignments. Extensive numerical experiments on 11 datasets underscore the benefits and efficacy of our batch inference approach. On zero-shot tasks with test batches of 75 samples, our approach yields near 20% improvement in ImageNet accuracy over CLIP's zero-shot performance. Additionally, we outperform state-of-the-art methods in the few-shot setting. Code is available at <https://github.com/SegoleneMartin/transductive-CLIP>.

6.1 Introduction

The emergence of large-scale vision-language models like CLIP (Contrastive Language-Image Pre-Training) [Radford et al., 2021] has marked a significant turning point in representation learning [Wang et al., 2022, Jia et al., 2021, Li et al., 2022]. By integrating both visual and textual modalities, these models have shown remarkable potential in crafting generic and richly informative concepts. Unlike traditional vision models, often constrained by task specificity, the representations gleaned from vision-language models are versatile, setting the stage for a breadth of downstream vision tasks and expanding the horizons of what is achievable in the domain.

Among the vision tasks that can be addressed with vision-language models, zero-shot and few-shot classification have particularly attracted attention. Notably, CLIP has demonstrated strong performance in zero-shot classification [Radford et al., 2021]. Several subsequent works have leveraged few-shot data,

a few labeled samples in the target downstream task, to further improve CLIP’s classification accuracy. Following on from the research on prompt learning in the NLP community, CoOp and CoCoOp [Zhou et al., 2022a, Zhou et al., 2022b] fine-tuned the pre-trained CLIP model using learnable textual tokens. Another type of approaches, like CLIP-Adapter [Gao et al., 2023] and TIP-Adapter [Zhang et al., 2022] provided CLIP with a parametric feature transformation, which generates adapted features and combines them with the original CLIP-encoded features. Despite their efficacy on few-shot classification benchmarks, these methods predominantly operate within the so-called *inductive* setting, where inference is conducted independently for each query (i.e., test) sample.

In contrast, in the *transductive* paradigm, one makes joint predictions for a batch of query samples, taking advantage of the query set statistics. The transductive setting for few-shot classification with vision-only models was pioneered in [Liu et al., 2019], and have since become prominent research subject, triggering an abundant, very recent literature on the subject, e.g., [Martin et al., 2022, Liu et al., 2020, Ziko et al., 2020, Veilleux et al., 2021, Tian et al., 2023, Zhu and Koniusz, 2023, Tao et al., 2023, Trosten et al., 2023], to list a few. These transductive few-shot classifiers were shown to significantly outperform their inductive counterparts, with benchmarks indicating up to a 10% increase in classification accuracy [Boudiaf et al., 2020]. In fact, this is in line with well-established theoretical facts in the classical literature on transductive learning [Vapnik, 1999, Joachims, 1999], which points to transductive prediction as a way to alleviate the scarcity of labeled data. Importantly, and beyond theoretical justification, the transductive setting is highly relevant in a breadth of practical computer vision scenarios, in which test data may come in mini-batches. This is the case, for instance, of online video streams and various types of time-series imaging, of portable-device photos, or of pixel-level tasks such as segmentation.

In this study, we take a close look at the transductive zero-shot and few-shot inference problems for the popular vision-language pre-trained CLIP model. We first make the surprising observation that standard clustering models, in the zero-shot case, and recent transductive methods, in the few-shot setting, do not bring improvements comparable to those observed with vision-only models, scoring even below their inductive counterparts; see Tables 6.1 and 6.2. This might explain why the transductive setting, despite its popularity, has not been explored so far for vision-language models. Potential questions that may fill this gap are (i) How to build informative text-image features for transductive inference, leveraging the textual knowledge in vision-language models? and (ii) Are the statistical assumptions underlying standard clustering and transductive inference methods appropriate for text-image features? In light of these challenges, this chapter brings the following contributions:

1. We propose a methodology to compute text-vision probability feature vectors, setting the stage for transductive few-shot classification specifically tailored for CLIP.
2. We reformulate the transductive zero-shot and few-shot classification challenge as an optimization problem on the unit simplex set by modeling the data with Dirichlet probability distributions. Crucially, the non-trivial deployment of the Dirichlet distributions brings substantial improvements in comparison to the common statistical models underlying standard clustering and transductive few-shot methods (e.g. Gaussian).
3. We propose a novel block Majorization-Minimization algorithm that addresses our problem efficiently and effectively, removing the need for cumbersome inner iterations in estimating the Dirich-

let parameters.

4. We report comprehensive evaluations, comparisons and ablations over 11 datasets, which point to the benefits of our mini-batch inference approach. On zero-shot ImageNet tasks with batches of 75 samples, the proposed method scores near 20% higher than inductive zero-shot CLIP in classification accuracy. Additionally, we outperform state-of-the-art methods in the few-shot setting.

6.2 Related works

6.2.1 Vision-language models

Vision-Language models, like CLIP, integrate visual and textual data to improve accuracy over various vision tasks. CLIP uses a dual-encoder structure, with one deep network dedicated for image encoding and another one specialized for text. This structure, along with proper projections at its bottleneck, yield image and text embeddings lying in the same low-dimensional vector space. Trained on a large dataset of 400 million text-image pairs, CLIP maximizes the cosine similarity between text and image embeddings using a contrastive loss. CLIP is pre-trained to match images with text descriptions, making it well-suited for zero-shot prediction. At inference time, to classify an image \mathbf{x} among K classes, the model predicts the class by choosing the one with the highest cosine similarity:

$$\operatorname{argmax}_{k \in \{1, \dots, K\}} \cos(f_{\text{im}}(\mathbf{x}), f_{\text{text}}(\mathbf{t}_k)), \quad (6.2.1)$$

where f_{im} and f_{text} are, respectively, the image and text encoders, and each \mathbf{t}_k is based on a text prompt, typically “a photo of a [name of class k]”. A figure of CLIP’s framework is given Appendix 6.A.1.

6.2.2 Few-shot classification

Inductive v.s. transductive setting

Few-shot image classification with pre-trained vision models has been the subject of extensive research recently [Ziko et al., 2020, Chen et al., 2019, Tian et al., 2020]. Within this area, the problem is tackled either in the *transductive* or *inductive* setting. The latter assumes that each instance in the testing batch is classified independently, omitting the correlations or shared information among instances [Chen et al., 2019, Wang et al., 2019, Hu et al., 2022]. In contrast, transductive inference is more comprehensive, as it makes joint predictions for the entire mini-batch of query samples, leveraging their statistics and shared information. Recent research has increasingly focused on transductive few-shot learning, including, for instance, methods based on constrained clustering [Martin et al., 2022, Boudiaf et al., 2023], label propagation [Zhu and Koniusz, 2023, Liu et al., 2019], optimal transport [Tian et al., 2023, Lazarou et al., 2021], information maximization [Boudiaf et al., 2020, Veilleux et al., 2021], prototype rectification [Liu et al., 2020], among other recent approaches [Tao et al., 2023, Trosten et al., 2023]. It has been consistently observed in this body of literature that the gap in accuracy between transductive and inductive methods could be considerable.

Few-shot CLIP

Beyond its zero-shot capabilities, the CLIP model has also been explored for few-shot image classification. In [Radford et al., 2021], the authors evaluated linear probe, which performs a simple fine-tuning of the visual encoder’s final layer using a few-shot support set (i.e., a few labeled samples in the downstream task). This approach has proven to be relatively ineffective in few-shot scenarios. Since then, a recent body of works have explored CLIP’s few-shot generalization. For instance, there is a noticeable emergence of *prompt learning* methods in computer vision, focusing on this specific problem [Zhou et al., 2022a, Zhou et al., 2022b, Chen et al., 2022]. Inspired by intensive recent prompt learning research in NLP [Shin et al., 2020, hong et al., 2021], these methods fine-tune learnable input text tokens using the few-shot support set. A different type of approaches, coined *adapters* [Gao et al., 2023, Zhang et al., 2022], fine-tune the encoded features rather than input text. For example, CLIP-Adapter [Gao et al., 2023] incorporates additional bottleneck layers to learn new features, while performing residual-style blending with the original pre-trained features. In a similar spirit, TIP-Adapter [Zhang et al., 2022] balances two prediction terms, one summarizing adaptively the information from the support set and the other preserving the textual knowledge from CLIP. All of these recent methods belong to the inductive family. To the best of our knowledge, our work is the first to explore transduction for CLIP’s few-shot image classification.

6.3 Proposed method

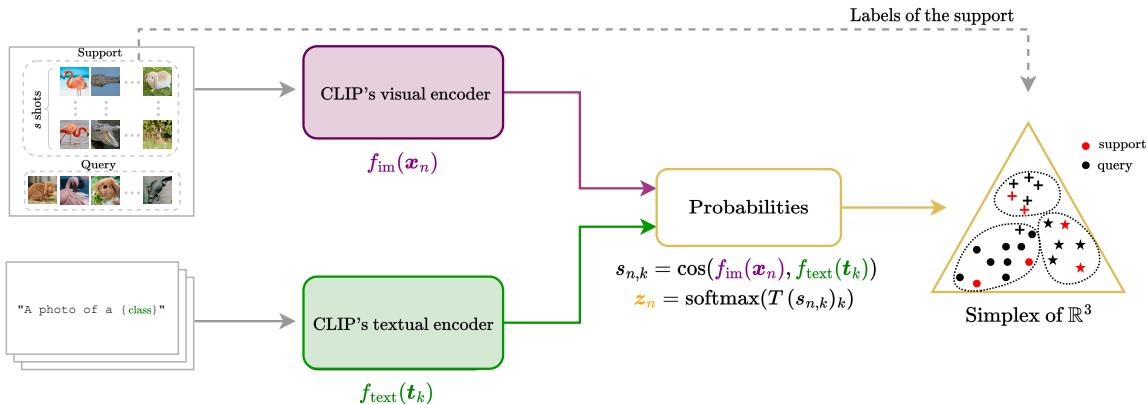


Figure 6.1: Given a transductive few-shot task, both visual and textual information are extracted from the images and class-wise prompts. The embeddings are next combined into vision-text probability vectors. Classification is carried out on the simplex set of \mathbb{R}^K using the labels of the support set. An empty support set corresponds to the zero-shot scenario, which is akin to a clustering problem.

Throughout the chapter, we define and employ specific notations to describe a single, randomly sampled few-shot task, which, during transductive prediction, is treated independently of the other randomly sampled tasks:

- N is the number of images in each randomly sampled task, with $(\mathbf{x}_n)_{1 \leq n \leq N}$ denoting the set of

images.

- K is the total number of distinct classes in the whole data set, among which a much smaller set of randomly sampled classes might appear in each mini-batch task, and might differ from one batch to another. Hence, apart from knowing the set of K classes in the whole data, as in standard inductive inference [Radford et al., 2021, Zhou et al., 2022a, Zhang et al., 2022], our transductive setting do not assume any additional knowledge about the particular set of classes that might appear randomly in each mini-batch.
- $\mathbb{S} \subset \{1, \dots, N\}$ indicates the indices of samples within the support set in the few-shot setting. For all $n \in \mathbb{S}$, one has access to the one-hot-encoded labels $\mathbf{y}_n \in \{0, 1\}^K$, such that for all $k \in \{1, \dots, K\}$, $y_{n,k} = 1$ if \mathbf{x}_n is an instance of class k , $y_{n,k} = 0$ otherwise.
- $\mathbb{Q} = \{1, \dots, N\} \setminus \mathbb{S}$ represents the indices of samples in the query min-batch set. In our experiments, the mini-batch size $|\mathbb{Q}|$ is set to 75.

The goal is to predict the classes of the query samples leveraging the supervision available from the support set. Note that when the support set is empty ($\mathbb{S} = \emptyset$), we encounter a zero-shot scenario, which is akin to a clustering problem.

6.3.1 Computing informative feature vectors

A seemingly intuitive approach to tackle the transductive challenge might be to use the visual embeddings obtained from CLIP’s visual encoder as the input features for the classifier. This is analogous to CLIP’s linear probe when it operates inductively. We pinpoint two main difficulties raised by this approach:

1. **Overlooking textual information:** A significant limitation of this method is that it omits the model’s textual knowledge. This is problematic as textual information is one of CLIP’s most powerful features.
2. **Normalization dilemma:** CLIP’s pre-training maximizes the scalar product between normalized textual and visual embeddings. Using normalized embeddings can introduce complexities in data distribution modeling, which, if misjudged, can impact the method interpretability and accuracy.

While some works in the classification literature have explored spherical distributions like the Von Mises-Fisher [Hasnat et al., 2017, Rossi and Barbaro, 2022, Scott et al., 2021] and the Fisher-Bingham [Hamsici and Martinez, 2007, Ali et al., 2019], our approach differs to address both issues mentioned above.

Our strategy consists in defining, for every $n \in \{1, \dots, N\}$, the feature vector for the data sample \mathbf{x}_n as CLIP’s zero-shot probability. Precisely, we define

$$\mathbf{z}_n = \text{softmax} \left\{ T \cos(f_{\text{im}}(\mathbf{x}_n), f_{\text{text}}(\mathbf{t}_k))_{1 \leq k \leq K} \right\}, \quad (6.3.1)$$

where $T > 0$ is a temperature parameter. Through this, both visual and textual information are incorporated into the feature vectors. Consequently, the task becomes a classification problem on the unit

simplex of \mathbb{R}^K , defined as

$$\Delta_K = \left\{ \mathbf{p} = (p_k)_{1 \leq k \leq K} \in \mathbb{R}_+^K \mid \sum_{k=1}^K p_k = 1 \right\} \quad (6.3.2)$$

Observe that, for datasets with a modest number of classes, defining feature vectors according to (6.3.1) also acts as a dimensionality reduction, with embedding's dimension going from 1024 (from CLIP's ResNet50) down to K , the number of classes. A recap of our framework is given in Figure 6.1.

6.3.2 Data distribution

Given feature vectors lying within the unit simplex set of \mathbb{R}^K , we advocate modeling the data using Dirichlet distributions. The Dirichlet distribution extends the beta distribution into higher dimensions, serving as a natural choice for modeling probability vectors over the simplex. For each class k within the set $\{1, \dots, K\}$, the data is assumed to follow a Dirichlet distribution, characterized by positive parameters $\boldsymbol{\alpha}_k = (\alpha_{k,i})_{1 \leq i \leq K} \in (0, +\infty)^K$, which describes the distribution shape. An illustration in \mathbb{R}^3 is given in Appendix 6.A.2. Mathematically, the density function is given by, for every $\mathbf{z} = (z_i)_{1 \leq i \leq K} \in \mathbb{R}^K$,

$$p(\mathbf{z} \mid \boldsymbol{\alpha}_k) = \frac{1}{\mathcal{B}(\boldsymbol{\alpha}_k)} \prod_{i=1}^K z_i^{\alpha_{k,i}-1} \mathbb{1}_{\mathbf{z} \in \Delta_K}, \quad (6.3.3)$$

where normalization factor $\mathcal{B}(\boldsymbol{\alpha}_k)$ is expressed as

$$\mathcal{B}(\boldsymbol{\alpha}_k) = \frac{\prod_{i=1}^K \Gamma(\alpha_{k,i})}{\Gamma\left(\sum_{i=1}^K \alpha_{k,i}\right)}, \quad (6.3.4)$$

and Γ denoting the Gamma function.

6.3.3 Simplex-based classification criterion

The proposed method simultaneously determines: (i) the soft assignment vectors $\mathbf{u} = (\mathbf{u}_n)_{1 \leq n \leq N}$ within the simplex $(\Delta_K)^N$, where the k -th component $u_{n,k}$ of vector \mathbf{u}_n specifies the probability for the n -th sample belonging to class k ; (ii) the Dirichlet distribution parameters $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_k)_{1 \leq k \leq K}$ where each $\boldsymbol{\alpha}_k$ is a K -dimensional vector with nonnegative components. We achieve this through the following maximum-likelihood estimation

$$\begin{aligned} & \underset{\mathbf{u}, \boldsymbol{\alpha}}{\text{minimize}} \quad -\mathcal{L}(\mathbf{u}, \boldsymbol{\alpha}) + \Phi(\mathbf{u}) + \lambda \Psi(\mathbf{u}), \\ & \text{subject to} \quad \mathbf{u}_n \in \Delta_K \quad \forall n \in \mathbb{Q}, \\ & \quad u_{n,k} = y_{n,k} \quad \forall n \in \mathbb{S}, \forall k \in \{1, \dots, K\}. \end{aligned} \quad (6.3.5)$$

In (6.3.5), \mathcal{L} is the log-likelihood model fitting objective for clustering:

$$\mathcal{L}(\mathbf{u}, \boldsymbol{\alpha}) = \sum_{n=1}^N \sum_{k=1}^K u_{n,k} \ln(p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k)), \quad (6.3.6)$$

where the density functions are defined by the Dirichlet models in (6.3.3). When the support set is not empty, this term also includes the supervision derived from the labeled instances. Term Φ acts as a barrier imposing the nonnegativity constraints on assignment variables, as in the soft K -means objective [MacKay, 2003, p.289], and is defined as

$$\Phi(\mathbf{u}) = \sum_{n=1}^N \sum_{k=1}^K u_{n,k} \ln u_{n,k}. \quad (6.3.7)$$

Finally, the penalty function Ψ , weighted by parameter $\lambda \in [0, +\infty)$, evaluates a partition complexity [Martin et al., 2022, Boykov et al., 2015], linked to the Minimum Description Length (MDL) concept in information theory:

$$\Psi(\mathbf{u}) = - \sum_{k=1}^K \pi_k \ln \pi_k, \quad (6.3.8)$$

where $\pi_k = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}$ is the proportion of query samples within class k . This MDL term penalizes the number of non-empty clusters, encouraging low-complexity partitions, i.e., with lower numbers of clusters.

6.4 Proposed algorithm

To tackle the minimization problem (6.3.5), our algorithm alternates minimization steps on the assignment variables and the Dirichlet parameters, producing sequences $(\mathbf{u}^{(\ell)})_{\ell \in \mathbb{N}}$ and $(\boldsymbol{\alpha}^{(\ell)})_{\ell \in \mathbb{N}}$ making the objective function decrease.

6.4.1 Minimization step w.r.t Dirichlet parameter

Suppose $\mathbf{u} \in \Delta_K^N$ is fixed. The estimation step with respect to the Dirichlet parameters consists in maximizing the log-likelihood in (6.3.6). Given the separability of the cost with respect to the variables $(\boldsymbol{\alpha}_k)_{1 \leq k \leq K}$, we can, without loss of generality, consider the minimization of the function F_k defined as $(\forall \boldsymbol{\alpha}_k \in (0, +\infty)^K)$

$$F_k(\boldsymbol{\alpha}_k) = \sum_{n=1}^N u_{n,k} \left(\sum_{i=1}^K -(\alpha_{k,i} - 1) \ln z_{n,i} + \sum_{i=1}^K \ln \Gamma(\alpha_{k,i}) - \ln \Gamma \left(\sum_{i=1}^K \alpha_{k,i} \right) \right). \quad (6.4.1)$$

The minimization of the Dirichlet negative log-likelihood (6.4.1) has already been explored in the literature [Narayanan, 1991, Huang, 2005, Minka, 2000]. The main strategy consists in resorting to a Majorization-Minimization (MM) algorithm. Specifically, a generic MM procedure corresponds to finding a minimizer $\boldsymbol{\alpha}_k^*$ of F_k by iteratively producing a sequence $(\boldsymbol{\alpha}_k^{(m)})_{m \in \mathbb{N}}$ such that, for every $m \in \mathbb{N}$,

$$\boldsymbol{\alpha}_k^{(m+1)} = \underset{\boldsymbol{\alpha}_k \in (0, +\infty)^K}{\operatorname{argmin}} q(\boldsymbol{\alpha}_k; \boldsymbol{\alpha}_k^{(m)}), \quad (6.4.2)$$

where for all $\boldsymbol{\beta}_k \in (0, +\infty)^K$, $q(\cdot; \boldsymbol{\beta}_k)$ is a so-called *tangent majorant* of F_k , satisfying

$$\begin{cases} F(\boldsymbol{\alpha}_k) \leq q(\boldsymbol{\alpha}_k; \boldsymbol{\beta}_k), \\ F(\boldsymbol{\beta}_k) = q(\boldsymbol{\beta}_k; \boldsymbol{\beta}_k). \end{cases} \quad (6.4.3)$$

The efficiency of the procedure (6.4.2) is highly dependent on the choice of the majorant. In [Minka, 2000], the author proposed a majorant function of (6.4.1) which consists in linearizing the concave term $\alpha_k \mapsto -\ln \Gamma\left(\sum_{i=1}^K \alpha_{k,i}\right)$ at β_k . The resulting MM algorithm was used for simplex clustering in [Blei et al., 2003, Pal and Heumann, 2022]. However, minimizing this majorant requires inverting the digamma function (i.e., the derivative of the log-Gamma function) with a Newton method, which can jeopardize the numerical convergence and slow down the overall algorithm.

In the following lemma, we introduce a novel tight majorant of F_k which yields closed-form updates, therefore avoiding sub-iterations within the MM algorithm.

Lemma 6.1. (Majorant of the negative log-likelihood) *Let $\varphi = \ln \Gamma(\cdot + 1)$. Then, for any $\beta_k = (\beta_{k,i})_{1 \leq i \leq K} \in (0, +\infty)^K$, the function $q(\cdot; \beta_k)$ defined as, for every $\alpha_k \in (0, +\infty)^K$,*

$$\begin{aligned} q(\alpha_k; \beta_k) = & \sum_{n=1}^N u_{n,k} \left[\sum_{i=1}^K \left(-(\alpha_{k,i} - 1) \ln z_{n,i} - \ln \alpha_{k,i} \right. \right. \\ & + \varphi(\beta_{k,i}) + \varphi'(\beta_{k,i})(\alpha_{k,i} - \beta_{k,i}) + \frac{c(\beta_{k,i})}{2}(\alpha_{k,i} - \beta_{k,i})^2 \Big) \\ & \left. \left. - \ln \Gamma\left(\sum_{i=1}^K \beta_{k,i}\right) - \left(\sum_{i=1}^K (\alpha_{k,i} - \beta_{k,i})\right) (\ln \Gamma)' \left(\sum_{i=1}^K \beta_{k,i}\right) \right] \end{aligned} \quad (6.4.4)$$

is a tangent majorant of F_k at β_k , where the function c is defined by

$$c: t \mapsto \begin{cases} \varphi''(0) & \text{if } t = 0, \\ \frac{\varphi(0) - \varphi(t) + \varphi'(t)t}{t^2} & \text{otherwise.} \end{cases} \quad (6.4.5)$$

A proof of Lemma 6.1 is provided in Appendix 6.A.3. At each iteration of our MM procedure, the minimizer of the majorizing function (6.4.4) is the positive root of a quadratic polynomial equation, resulting in Algorithm 15. In Appendix 6.A.3, we show that this majorant speeds up the MM scheme over Minka's [Minka, 2000].

Algorithm 15: MM-quadratic($\mathbf{u}_{\cdot, k}$, α_k)

```

Initialize  $\alpha_k^{(0)} = \alpha_k$ .
for  $m = 0, 1, \dots$ , do
  for  $i \in \{1, \dots, K\}$  do
     $b_{k,i} = \varphi'(\alpha_{k,i}^{(m)}) - (\ln \Gamma)' \left( \sum_{j=1}^K \alpha_{k,j}^{(m)} \right) - c(\alpha_{k,i}^{(m)}) \alpha_{k,i}^{(m)} - \left( \sum_{n=1}^N u_{n,k} \right)^{-1} \sum_{n=1}^N u_{n,k} \ln(z_{n,i})$ 
     $\alpha_{k,i}^{(m+1)} = \frac{-b_{k,i} + \sqrt{b_{k,i}^2 + 4c(\alpha_{k,i}^{(m)})}}{2c(\alpha_{k,i}^{(m)})}$ .
  
```

6.4.2 Minimization step w.r.t assignment variable

Let the iteration number $\ell \in \mathbb{N}$ and $\alpha = (\alpha_k)_{1 \leq k \leq K} \in ((0, +\infty)^K)^K$ be fixed. Because of the partition complexity term in (6.3.8), the direct minimization of the partial function with respect to \mathbf{u}_n , for every

$n \in \mathbb{Q}$, is not closed form. Since the partition complexity penalty is concave, we propose to replace it by its linear upper-bound, leading us to minimize

$$\mathbf{u}_n \mapsto -\sum_{k=1}^K u_{n,k} \ln(p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k)) + \sum_{k=1}^K u_{n,k} \ln u_{n,k} - \frac{\lambda}{|\mathbb{Q}|} (\ln(\boldsymbol{\pi}^{(\ell+1)}) + 1)^\top (\mathbf{u}_n - \mathbf{u}_n^{(\ell)}), \quad (6.4.6)$$

under the simplex and supervision constraints. In (6.4.6), $\boldsymbol{\pi}^{(\ell+1)} = (\pi_k^{(\ell+1)})_k$ is the vector whose k -th component is $\pi_k^{(\ell+1)} = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}^{(\ell)}$, and the log function operates componentwise.

Solving this minimization problem yields the updates, for every $n \in \mathbb{Q}$,

$$\mathbf{u}_n^{(\ell+1)} = \text{softmax} \left(\left(\ln p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k) + \frac{\lambda}{|\mathbb{Q}|} \ln(\pi_k^{(\ell+1)}) \right)_k \right).$$

Details for deriving this expression are given in Appendix 6.A.4.

6.4.3 Global algorithm and class-assignment

Finally, given the estimation steps on the assignment variables $\mathbf{u} = (\mathbf{u}_n)_{n \in \mathbb{Q}}$ and on the Dirichlet parameters $(\boldsymbol{\alpha}_k)_{1 \leq k \leq K}$ derived respectively in Sections 6.4.2 and 6.4.1, our complete procedure to tackle the minimization problem in (6.3.5) is detailed in Algorithm 16. We name it EM-Dirichlet as it shares close links to the EM algorithm, as it will be established in Proposition 6.2 in Section 6.5.

In the zero-shot scenario, the tasks at hand can be seen as a form of simplex clustering. There exists a straightforward method to map each cluster to a corresponding class label in an injective manner. Let $(\mathcal{C}_k)_{k \in \mathcal{K}}$ denote the set of non-empty clusters found with a clustering method, for instance ours, with \mathcal{K} a subset of $k \in \{1, \dots, K\}$ and for all $k \in \mathcal{K}$, \mathcal{C}_k a subset of \mathbb{Q} . We proceed in the following way:

1. For each $k \in \mathcal{K}$, calculate the mean of cluster k , $\mathbf{m}_k = (m_{k,\ell})_{1 \leq \ell \leq K} \in \Delta_K$, as

$$\mathbf{m}_k = \frac{1}{|\mathcal{C}_k|} \sum_{n \in \mathcal{C}_k} \mathbf{z}_n. \quad (6.4.7)$$

The element $m_{k,\ell}$ is interpreted as the probability that cluster k is associated with class ℓ . While it may seem intuitive to assign cluster k the class ℓ for which $m_{k,\ell}$ is maximal, this could lead to multiple clusters being assigned to the same class, which we wish to avoid.

2. Resolve the class-to-cluster assignments through a bipartite graph matching that maximizes

$$\sum_{k \in \mathcal{K}} \sum_{\ell=1}^K a_{k,\ell} m_{k,\ell} \quad (6.4.8)$$

over all possible assignment matrices $\mathbf{A} = (a_{k,\ell}) \in \{0, 1\}^{|\mathcal{K}| \times K}$ that satisfy $\mathbf{A}^\top \mathbf{1}_{|\mathcal{K}|} = \mathbf{1}_K$. This class assignment integer linear programming problem can be solved with algorithms such as [Crouse, 2016]. Figure 6.2 gives an illustration of our graph matching procedure for assigning each cluster to a class.

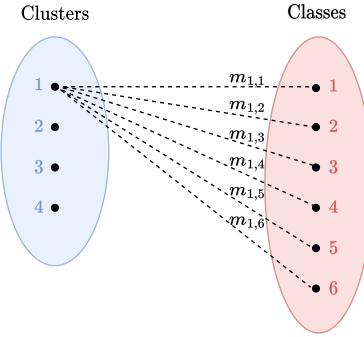


Figure 6.2: Illustration of the bipartite matching for class assignment.

Algorithm 16: EM-Dirichlet

```

Initialize  $\mathbf{u}^{(0)}$  as CLIP's probabilities and for all  $k \in \{1, \dots, N\}$ ,  $\boldsymbol{\alpha}_k^{(0)} = \mathbf{1}_K$ .
for  $\ell = 0, 1, \dots$ , do
    // Update Dirichlet parameter for each class
     $\boldsymbol{\alpha}_k^{(\ell+1)} = \text{MM-quadratic}(\mathbf{u}_{\cdot,k}^{(\ell)}, \boldsymbol{\alpha}_k^{(\ell)}), \quad \forall k \in \{1, \dots, K\}$ ,
    // Update class proportions
     $\pi_k^{(\ell+1)} = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}^{(\ell)}, \quad \forall k \in \{1, \dots, K\}$ ,
    // Update assignment variable for all query samples
     $\mathbf{u}_n^{(\ell+1)} = \text{softmax} \left( \left( \ln p(z_n | \boldsymbol{\alpha}_k^{(\ell+1)}) + \frac{\lambda}{|\mathbb{Q}|} \ln(\pi_k^{(\ell+1)}) \right)_k \right), \quad \forall n \in \mathbb{Q}$ .

```

6.5 Links with other clustering and transductive few-shot objectives

The general log-likelihood model fitting objective in (6.3.6), also referred to as probabilistic K -means [Kearns et al., 1998, Boykov et al., 2015], is well-established in the clustering literature. Indeed, it is a generalization of the ubiquitous K -means, which corresponds to the particular choice of the Gaussian distribution for the densities in (6.3.6), with covariance matrices fixed to the identity matrix. This general objective has a strong, inherent bias towards K -balanced partitions, a theoretically well-established fact in the clustering literature [Kearns et al., 1998, Boykov et al., 2015]. To mitigate this bias and address realistic, potentially imbalanced few-shot query sets, the recent transductive few-shot method in [Martin et al., 2022] coupled the MDL term in (6.3.8) with the standard K -means objective. This corresponds to the general data-fitting function we tackle in (6.3.6), but with the likelihood densities assumed to be Gaussian. As we will see in our experiments (Table 6.2), the non-trivial deployment of the Dirichlet model is crucial, outperforming significantly [Martin et al., 2022] in CLIP's few-shot setting. Furthermore, we show in the following an interesting result, which connects the general unbiased clustering problem we propose in (6.3.5), to the well-known Expectation-Maximization (EM) algorithm for mixture models [Bishop and Nasrabadi, 2006, p.438]. Indeed, optimizing the objective in (6.3.5) could be viewed as a generalization of EM, enabling to control the class-balance parameter λ .

Proposition 6.2. Consider the unsupervised classification problem, i.e. $\mathbb{S} = \emptyset$. Suppose the value of λ

in (6.3.5) is set to the size of the query set, i.e., $\lambda = |\mathbb{Q}|$. Then Algorithm 16 is equivalent to the EM algorithm when applied to a generic mixture model

$$p(\mathbf{z}_n \mid \boldsymbol{\pi}, \boldsymbol{\alpha}) = \sum_{k=1}^K \pi_k p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k), \quad (6.5.1)$$

where $\boldsymbol{\pi} = (\pi_k)_{1 \leq k \leq K} \in \Delta_K$ are the mixture coefficients.

Proof. Given the mixture model (6.5.1), the EM algorithm aims at maximizing the log-likelihood function

$$L(\boldsymbol{\pi}, \boldsymbol{\alpha}) = \sum_{n \in \mathbb{Q}} \ln \left(\sum_{k=1}^K \pi_k p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k) \right) \quad (6.5.2)$$

with respect to $\boldsymbol{\pi}$ and $\boldsymbol{\alpha}$. The process involves two steps: **expectation** and **maximization**, and the algorithm iteratively generates sequences $\{\boldsymbol{\pi}^{(\ell)}\}_{\ell \in \mathbb{N}} \subset \Delta_K$ and, for every $k \in \{1, \dots, K\}$, $\{\boldsymbol{\alpha}_k^{(\ell)}\}_{\ell \in \mathbb{N}} \subset (0, +\infty)^K$.

During the **expectation step**, for a given iteration number $\ell \in \mathbb{N}$, we compute the expected responsibilities. For each query sample $n \in \mathbb{Q}$, we define $\mathbf{u}_n^{(\ell)} = (u_{n,k}^{(\ell)})_{1 \leq k \leq K}$ by

$$u_{n,k}^{(\ell)} = \frac{\pi_k^{(\ell)} p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k^{(\ell)})}{\sum_{i=1}^K \pi_i^{(\ell)} p(\mathbf{z}_n \mid \boldsymbol{\alpha}_i^{(\ell)})}. \quad (6.5.3)$$

This quantity corresponds to the probability of the data point n belonging to class k based on the current estimates of $\boldsymbol{\pi}^{(\ell)}$ and $\boldsymbol{\alpha}_k^{(\ell)}$.

In the **maximization step**, we derive an upper bound for the log-likelihood at the current iterate using the responsibilities calculated in the expectation step, along with Jensen's inequality. This majorization reads

$$L(\boldsymbol{\pi}, \boldsymbol{\alpha}) \leq q((\boldsymbol{\pi}, \boldsymbol{\alpha}); (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\alpha}^{(\ell)})), \quad (6.5.4)$$

where $q(\cdot; (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\alpha}^{(\ell)}))$ is defined, for all $\boldsymbol{\pi} \in \Delta_K$ and $\boldsymbol{\alpha} \in ((0, +\infty)^K)^K$, by

$$q((\boldsymbol{\pi}, \boldsymbol{\alpha}); (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\alpha}^{(\ell)})) = \sum_{n \in \mathbb{Q}} \sum_{k=1}^K u_{n,k}^{(\ell)} \ln \left(\frac{\pi_k p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k)}{u_{n,k}^{(\ell)}} \right).$$

This upper bound is separable and defines a tight majorant, i.e., $q((\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\alpha}^{(\ell)}); (\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\alpha}^{(\ell)})) = L(\boldsymbol{\pi}^{(\ell)}, \boldsymbol{\alpha}^{(\ell)})$. Next, one maximizes the majorant with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\pi}$ under the simplex constraints. This yields the expression

$$(\forall k \in \{1, \dots, K\}) \quad \pi_k^{(\ell+1)} = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}^{(\ell)}, \quad (6.5.5)$$

i.e., the mixing coefficients are the average of the responsibilities for each class over all data points in the query set. On the other hand, for each class $k \in \{1, \dots, K\}$, the parameters $\boldsymbol{\alpha}_k^{(\ell+1)}$ are set by solving the optimization problem

$$\underset{\boldsymbol{\alpha}_k \in (0, +\infty)^K}{\text{maximize}} \sum_{n \in \mathbb{Q}} u_{n,k}^{(\ell)} \ln p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k). \quad (6.5.6)$$

We can then show that the updates are identical to those performed in Algorithm 16 when $\lambda = |\mathbb{Q}|$

and $\mathbb{S} = \emptyset$. The identity of the updates on $\boldsymbol{\alpha}$ and $\boldsymbol{\pi}$ are obvious. For \mathbf{u} , note that Equation (6.5.3) can be rewritten

$$\begin{aligned} u_{n,k}^{(\ell+1)} &= \frac{\pi_k^{(\ell+1)} p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k^{(\ell+1)})}{\sum_{i=1}^K \pi_i^{(\ell+1)} p(\mathbf{z}_n \mid \boldsymbol{\alpha}_i^{(\ell+1)})}, \\ &= \frac{\exp \left(\ln \pi_k^{(\ell+1)} + \ln p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k^{(\ell+1)}) \right)}{\sum_{i=1}^K \exp \left(\ln \pi_i^{(\ell+1)} + \ln p(\mathbf{z}_n \mid \boldsymbol{\alpha}_i^{(\ell+1)}) \right)}, \end{aligned}$$

or equivalently,

$$\mathbf{u}_n = \text{softmax} \left((\ln \pi_k^{(\ell+1)} + \ln p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k^{(\ell+1)}))_k \right), \quad (6.5.7)$$

thus aligning with the update in Algorithm 16. \square

6.6 Experiments

We evaluated our method on 11 publicly accessible image classification datasets which were also utilized in CLIP [Radford et al., 2021]: ImageNet [Russakovsky et al., 2015], Caltech101 [Fei-Fei et al., 2006], OxfordPets [Parkhi et al., 2012], StanfordCars [Krause et al., 2013], Flowers102 [Nilsback and Zisserman, 2008], Food101 [Bossard et al., 2014], FGVC Aircraft [Maji et al., 2013], SUN397 [Xiao et al., 2010], DTD [Cimpoi et al., 2014], EuroSAT [Helber et al., 2019] and UCF101 [Soomro et al., 2012]. To ensure reproducibility, we adhere to the dataset splits provided by CoOp [Zhou et al., 2022a] and use the prompts employed in TIP-Adapter [Zhang et al., 2022]. All experiments are conducted using CLIP’s pre-trained ResNet50 visual encoder. The temperature in the probabilities (6.3.1) is fixed to $T = 30$.

6.6.1 Zero-shot

Tasks generation

For generating query sets in our transductive zero-shot setting, we employ a practical approach that maintains manageable batch sizes. At each new task (mini-batch), we randomly select the classes that will be represented in the query set, with the actual number of distinct classes ranging from 3 to 10, also selected at random. It is important to note that the set of classes occurring in each batch remain undisclosed, and vary randomly from one batch to another, ensuring that the clustering task is still performed over all K potential classes present in the whole dataset. Subsequently, we randomly select $|\mathbb{Q}| = 75$ images in to the chosen classes to constitute the query set. During transductive inference, the query set of each task is treated independently of the other randomly sampled tasks.

Comparative methods

We conduct a comparative evaluation of our clustering methodology, EM-Dirichlet, and its variant utilizing hard assignments, denoted as Hard EM-Dirichlet, against a range of clustering objective functions and algorithms: Hard and soft K -means [MacKay, 2003, p.286], EM for Gaussian mixtures with identity covariance (EM-Gaussian (cov. Id)) and with diagonal covariance (EM-Gaussian (cov. diag)) [Bishop

and Nasrabadi, 2006, p.438], and Hard KL K -Means [Cao et al., 2013]. Furthermore, our comparison provides a full ablation study of the terms in general objective function (6.3.5):

1. The log-likelihood model fitting term (6.3.6), which varies across Gaussian (employed in Hard K -means, Soft K -means, EM-Gaussian), and Dirichlet (in our method).
2. The entropic barrier (6.3.7) featured in both Soft K -means and the EM-based approaches.
3. The MDL partition-complexity term (6.3.8), incorporated exclusively in the EM methods.

Initialization is uniform across different clustering techniques, utilizing CLIP’s predictions from Equation (6.3.1). In all EM-based methods, the regularization parameter λ is set according to $\lambda = \frac{5}{K}|\mathbb{Q}|$, to maintain consistency across comparisons.

Results

We assess the clustering methods on zero-shot tasks, using both the visual embeddings and the combined text-vision feature vectors. We also include the zero-shot classification results from CLIP. In Table 6.1, we report average accuracy over 1,000 tasks using the graph cluster-to-classes assignment described in Section 6.4.3. Table 6.1 conveys several crucial messages:

- Clustering visual embeddings alone does not suffice to surpass inductive CLIP’s zero-shot performance. Incorporating textual information via probability features enhances the performance, even for methods initially designed for Gaussian distributions.
- Gaussian-based data-fitting approaches are sub-optimal for simplex clustering. Replacing the Gaussian metric with a Kullback-Leibler divergence is beneficial. Employing a Dirichlet data-fitting term within the EM framework significantly improves the results compared to EM-Gaussian methods, highlighting the necessity of accurate data distribution modeling.
- Introducing the partition complexity term (in the EM methods), which discourages overly balanced predictions, proves advantageous for the performance.
- Using an adapted transductive model like Hard EM-Dirichlet, accuracy improves considerably, showing a 9% rise across 11 datasets, and nearly 20% on ImageNet.

In Appendix 6.A.5, we show that zero-shot performance improves with larger query set sizes, indicating enhanced transduction efficiency with increasing mini-batch size.

6.6.2 Few-shot

Task generation

We follow the realistic transductive few-shot evaluation protocol proposed recently in [Martin et al., 2022]. Specifically, the query sets are constructed with a fixed number of effective classes $k_{\text{eff}} = 5$, from which $|\mathbb{Q}|$ samples are randomly selected. This approach aligns with established few-shot protocols in the literature [Snell et al., 2017, Liu et al., 2019, Veilleux et al., 2021]. These classes remain undisclosed during inference, ensuring the task is a K -way classification. The support set is created by uniformly

	Food101	EuroSAT	DTD	OxfordPets	Flowers102	Caltech101	UCF101	FGVC Aircraft	Stanford Cars	SUN397	ImageNet	Average
Zero-shot CLIP	77.1	36.5	42.9	85.1	66.1	84.4	61.7	17.1	55.8	58.6	58.3	58.5
Vis. emb.	Hard K-means	52.2	37.9	40.0	54.5	44.8	62.9	49.2	14.3	22.4	39.6	29.3
	Soft K-means	17.6	29.9	19.1	40.4	36.1	21.3	13.3	10.6	11.1	9.1	10.1
	EM-Gaussian (Id cov.)	14.0	14.5	9.4	6.9	5.3	30.3	7.4	1.9	2.5	5.3	8.3
	EM-Gaussian (diag cov.)	51.4	40.6	37.5	59.2	45.6	61.8	47.2	13.6	24.3	35.1	28.3
	Probabilities	49.5	35.2	38.7	62.4	44.5	52.2	46.6	14.5	29.6	41.4	31.0
	Hard K-means	41.8	21.5	18.3	56.6	34.3	50.5	30.2	7.2	34.8	18.8	19.1
	Soft K-means	21.4	14.5	16.5	21.1	23.1	33.6	19.3	6.8	18.5	18.7	19.1
	EM-Gaussian (Id cov.)	63.3	33.1	38.7	71.1	51.1	66.6	56.0	16.5	46.9	54.8	48.5
	EM-Gaussian (diag cov.)	72.2	34.9	40.8	73.0	61.1	72.0	60.6	17.7	56.2	61.8	61.0
	Hard KL K-means	88.2	33.0	47.7	87.3	71.5	88.4	69.0	19.2	65.5	77.3	76.9
	EM-Dirichlet	90.2	36.1	49.3	90.9	73.1	89.7	70.3	20.4	67.7	78.5	77.6
	Hard EM-Dirichlet											67.6

Table 6.1: Average accuracy of clustering methods over 1,000 zero-shot classification tasks. Inference is performed both on the visual embeddings and on the text-vision probability features.

selecting s images from each of the K classes. The ensuing results are derived performing few-shot tasks with 1, 2, 4, 8, and 16 shots. During inference on the test set, the size of the query set is set to $|\mathcal{Q}| = 75$, while for validation, the size is reduced to $|\mathcal{Q}| = 35$ due to data limitations.

Hyper-parameters

Parameter λ in EM-Dirichlet is set to the fixed value $\lambda = \frac{k_{\text{eff}}}{K} |\mathcal{Q}|$. Methods with tunable hyper-parameters are fine-tuned using the validation split provided with each dataset. In line with [Hu et al., 2022], validation is performed on five s -shot tasks across all datasets and for every shot number. These tasks, crafted as previously detailed, use support and query instances drawn from the validation set. The hyper-parameters are then optimized through a grid search to maximize accuracy on the validation set.

Results

We evaluate the accuracy of our proposed transductive methods, EM-Dirichlet and Hard EM-Dirichlet, against several recent transductive few-shot methods, including BD-CSPN [Liu et al., 2020], Laplacian Shot [Ziko et al., 2020], α -TIM [Veilleux et al., 2021], and PADDLE [Martin et al., 2022]. Additionally, we benchmark against two inductive few-shot methods designed for CLIP: TIP-Adapter [Zhang et al., 2022] and CoOp [Zhou et al., 2022b]. The results, averaged across 1,000 tasks with 4 shots, are presented in Table 6.2 and for the other number of shots in Appendix 6.A.6.

Our method surpasses competing approaches on the majority of datasets, with a more pronounced advantage observed on challenging datasets that have a large number of classes, such as SUN397 and ImageNet. The accuracy gap between our method and the inductive ones shows the benefits of transductive inference. On the other hand, the inferior performance of other transductive methods can be attributed to their lack of adaptability to simplex classification.

Interestingly, our results indicate that on some datasets such as Food101, our method performs better

in the zero-shot than in the few-shot setting. This is consistent with Radford et al. [Radford et al., 2021], suggesting that few labeled examples can negatively impact classification, possibly due to outliers or ambiguous examples in the support set.

Lastly, we observe that inductive methods outperform ours on the EuroSAT dataset. This might be due to the inclusion of text information in the vision-text features. While typically advantageous, it is possible that the text information introduces a confounding effect specific to this dataset.

	Food101	EuroSAT	DTD	OxfordPets	Flowers102	Caltech101	UCF101	FGVC Aircraft	Stanford Cars	SUN397	ImageNet	Average	Time (s)
Ind.	Tip-Adapter	76.7	72.5	54.7	86.4	83.2	88.8	72.1	23.7	63.9	66.7	62.7	68.3
	CoOp	76.3	63.2	52.2	86.2	81.0	87.7	67.0	22.2	61.3	63.4	59.9	65.5
Trans.	BDSCPN	74.7	46.1	45.2	81.3	74.2	82.0	59.0	18.0	48.1	54.5	49.2	57.5
	Laplacian Shot	76.6	53.0	52.6	88.4	85.5	86.8	67.0	22.2	60.4	63.8	56.3	64.8
	α -TIM	66.1	46.1	45.3	87.1	79.1	83.3	59.4	20.4	53.4	53.4	42.7	57.8
	PADDLE	71.8	45.9	50.0	84.7	82.3	81.9	63.7	21.3	56.1	60.6	52.1	60.9
	EM-Dirichlet	88.7	50.8	62.6	92.5	91.3	90.1	76.1	24.9	73.5	80.9	78.4	73.6
	Hard EM-Dirichlet	87.9	50.8	60.5	91.7	90.5	89.8	75.3	24.2	72.6	80.2	78.3	72.9

Table 6.2: Evaluation of our approach against two benchmarks – 1) inductive methods specifically designed for few-shot classification using CLIP, and 2) transductive few-shot methods applied to probability feature vector classification. The analysis encompasses 1,000 distinct 4 shots tasks. We also report average execution time for a single task, computed over 1,000 tasks, on the ImageNet dataset.

6.7 Conclusion

In conclusion, our study expands transductive inference to vision-language models like CLIP, previously unexplored in this domain. We demonstrate that the transductive methodology can boost image classification accuracy, including in zero-shot scenarios. Future work could apply our transductive CLIP approach to other tasks like segmentation and out-of-distribution detection.

6.A Appendices

6.A.1 About CLIP

An illustration of the framework is given in Figure 6.4.

6.A.2 Illustration of a Dirichlet distribution

Figure 6.3 presents examples of Dirichlet distributions on the unit simplex of \mathbb{R}^3 .

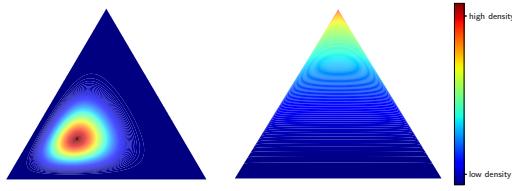


Figure 6.3: Examples of Dirichlet distributions on the simplex of \mathbb{R}^3 , for $\alpha = (10, 5.0, 5.0)$ (left) and $\alpha = (0.975, 0.975, 3.0)$ (right)

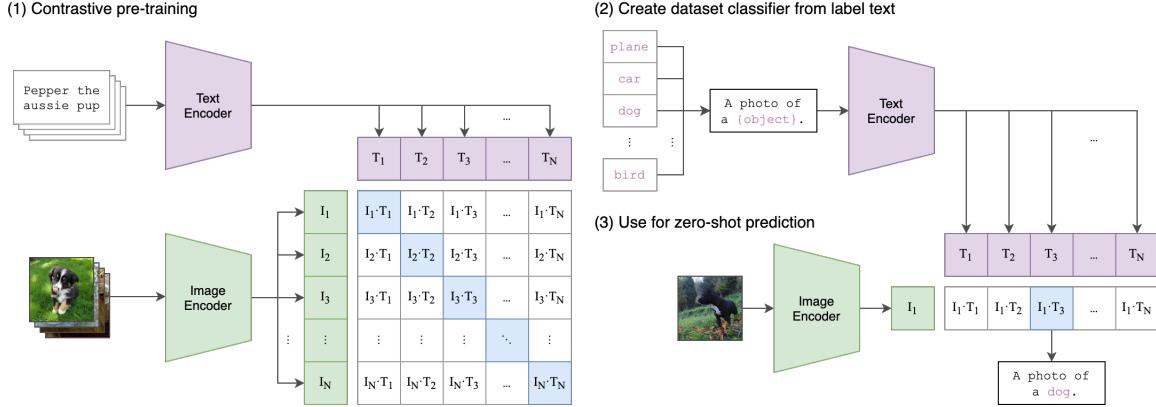


Figure 6.4: CLIP pre-training and zero-shot classification, image from [Radford et al., 2021]. The CLIP dataset comprises 400 million text-image pairs sourced from the web. CLIP itself is a neural network with two distinct encoders: a visual encoder (typically a ResNet or a Vision Transformer) and a textual encoder (a Transformer). Both encoders produce embeddings of the same dimension. (1) During training, these encoders are optimized to maximize the cosine similarity between each image and its corresponding text, while minimizing similarity for non-corresponding pairs. (2) For a new classification task on a different dataset with classes like plane, car, dog, etc., the CLIP classifier is formed by computing textual embeddings for prompts like “a photo of [class label]”. (3) To employ the zero-shot CLIP classifier on an image, its visual embedding is computed, and the class with the highest similarity with this embedding is assigned.

6.A.3 Majorization-Minimization algorithm

We provide the details for our new MM Algorithm 15 for minimizing (6.4.1). Our approach is based on constructing a quadratic bound of the function $\ln \Gamma(\cdot + 1)$, which is a consequence of the following lemma.

Lemma 6.3. ([Erdogan and Fessler, 2002]) *Let ψ be a twice-continuously differentiable function on $[0, +\infty[$. Assume that ψ'' is decreasing on $[0, +\infty[$. Let $z \in [0, +\infty[$ and let*

$$c_\psi(z) = \begin{cases} \frac{\psi''(0)}{2} & \text{if } z = 0 \\ \frac{\psi(0) - \psi(z) + \psi'(z)z}{z^2} & \text{otherwise.} \end{cases} \quad (6.A.1)$$

Then, for every $x \in [0, +\infty[$,

$$\psi(x) \leq \psi(z) + \psi'(z)(x - z) + \frac{1}{2}c_\psi(z)(x - z)^2. \quad (6.A.2)$$

We are now ready to prove Lemma 6.1.

Proof. We first observe that $\alpha_k \mapsto -\ln \Gamma \left(\sum_{i=1}^K \alpha_{k,i} \right)$ is concave. Consequently, we can upper-bound this term at β_k using its first-order Taylor expansion around β_k . Furthermore, considering the relation

$$\forall t \in (0, +\infty), \quad \ln \Gamma(t) = \varphi(t) - \ln t, \quad (6.A.3)$$

and given that the prerequisites of Lemma 6.3 are fulfilled by φ , the result in (6.4.4) follows immediately. \square

For a fixed value of $\beta_k \in (0, +\infty)^K$, the minimizer $\hat{\alpha}_k$ of the majorant given by Lemma 6.1 is such that, for every $i \in \{1, \dots, K\}$, $\hat{\alpha}_{k,i}$ is the unique positive root of the second order polynomial equation

$$c(\beta_{k,i})\alpha_{k,i}^2 + b_{k,i}(\beta_k)\alpha_{k,i} = 1, \quad (6.A.4)$$

with

$$b_{k,i}(\beta_k) = \varphi'(\beta_{k,i}) - (\ln \Gamma)' \left(\sum_{j=1}^K \beta_{k,j} \right) - c(\beta_{k,i})\beta_{k,i} - \left(\sum_{n=1}^N u_{n,k} \right)^{-1} \sum_{n=1}^N u_{n,k} \ln z_{n,i}. \quad (6.A.5)$$

Hence,

$$\hat{\alpha}_{k,i} = \frac{-b_{k,i}(\beta_k) + \sqrt{(b_{k,i}(\beta_k))^2 + 4c(\beta_{k,i})}}{2c(\beta_{k,i})}, \quad (6.A.6)$$

which yields the MM updates described in Algorithm 15.

In Table 6.3, we compare the convergence speed of the MM Algorithm 15 and the Block MM Algorithm 16, using our majorant (6.4.4) versus the one proposed by Minka in [Minka, 2000]. For Algorithm 15, the convergence criterion is defined as $\frac{\|\alpha^{(m+1)} - \alpha^{(m)}\|^2}{\|\alpha^{(m)}\|^2} \leq \varepsilon$, and for Algorithm 16 as $\frac{\|\alpha^{(\ell+1)} - \alpha^{(\ell)}\|^2}{\|\alpha^{(\ell)}\|^2} \leq \varepsilon$, where $\varepsilon = 10^{-13}$. Our MM algorithm is approximately twice as fast as Minka's.

	Algo. 15	Algo. 16
Minka's	2.04×10^{-1}	2.09
Ours	7.62×10^{-2}	1.04

Table 6.3: Time before reaching the convergence criterion in seconds, for Algorithm 15 and 16. The displayed time is the average execution time per task, computed over 1,000 tasks, on the ImageNet dataset with 4 shots.

6.A.4 Estimation step on assignments in our algorithm

We provide more details on the derivation of the closed-form update of variable \mathbf{u}_n at each iteration $\ell \in \mathbb{N}$. Consider the function F given by

$$F(\mathbf{u}_n) = - \sum_{k=1}^K u_{n,k} \ln(p(z_n | \alpha_k)) + \iota_{\Delta_K}(\mathbf{u}_n) - \frac{\lambda}{|\mathcal{Q}|} (\ln(\boldsymbol{\pi}^{(\ell+1)}) + \mathbf{1})^\top (\mathbf{u}_n - \mathbf{u}_n^{(\ell)}) + \sum_{k=1}^K u_{n,k} \ln u_{n,k}, \quad (6.A.7)$$

where ι_{Δ_K} is the indicator function of the simplex Δ_K , assigning zero to points within the simplex and $+\infty$ elsewhere.

Let us see how to compute the minimizer of (6.A.7) via the proximal operator (see [Bauschke and Combettes, 2019, Eq. 24.2] for a definition). We define the function ψ on \mathbb{R}^K as

$$\psi(\mathbf{x}) = \begin{cases} \sum_{k=1}^K x_k \ln(x_k) - \frac{x_k^2}{2}, & \text{if } \mathbf{x} \in \Delta_K, \\ +\infty, & \text{otherwise.} \end{cases} \quad (6.A.8)$$

The proximal operator of ψ , which is well-established as the softmax function, allows for the practical computation of the minimizer [Combettes and Pesquet, 2020, Example 2.23]. Since F is proper, lower semi continuous and convex, finding the minimizer of F is equivalent to finding \mathbf{u}_n such that $0 \in \partial F(\mathbf{u}_n)$. This reads

$$\begin{aligned} 0 \in \partial F(\mathbf{u}_n) &\iff 0 \in -\ln(p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k)) - \frac{\lambda}{|\mathcal{Q}|} (\ln(\boldsymbol{\pi}^{(\ell+1)}) + 1) + \partial\psi(\mathbf{u}_n) + \mathbf{u}_n, \\ &\iff \mathbf{u}_n = \text{softmax} \left(\left(\ln p(\mathbf{z}_n \mid \boldsymbol{\alpha}_k) + \frac{\lambda}{|\mathcal{Q}|} \ln \pi_k^{(\ell+1)} \right)_k \right), \end{aligned}$$

where we used the characterization of the proximity operator [Bauschke and Combettes, 2019, Prop. 16.44].

6.A.5 Zero-shot performance as a function of the size the query set

We point to Figure 6.5 which displays the accuracy of our methods EM-Dirichlet and Hard EM-Dirichlet in the zero-shot setting versus the number of samples in the query set.

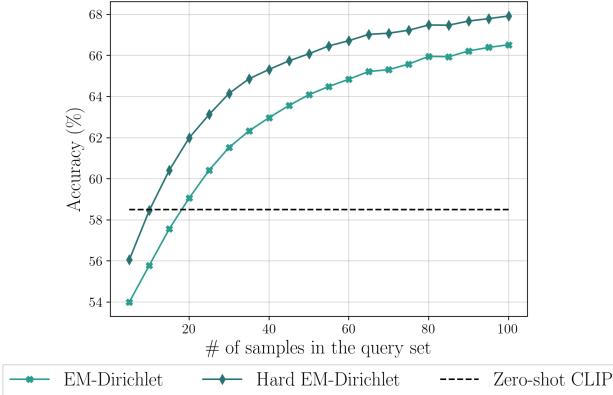


Figure 6.5: Average accuracy on the 11 datasets as a function of the number of samples in the query set, over 1,000 tasks generated following the protocol described in Section 6.6.1. As anticipated, the efficiency of transduction increases with the number of samples in the query set.

6.A.6 Additional results in the few-shot setting

In addition to the results in the 4-shot case presented in Table 6.2, we provide the results for other number of shots. Figure 6.6 displays the accuracy as a function of the number of shots. This analysis includes our

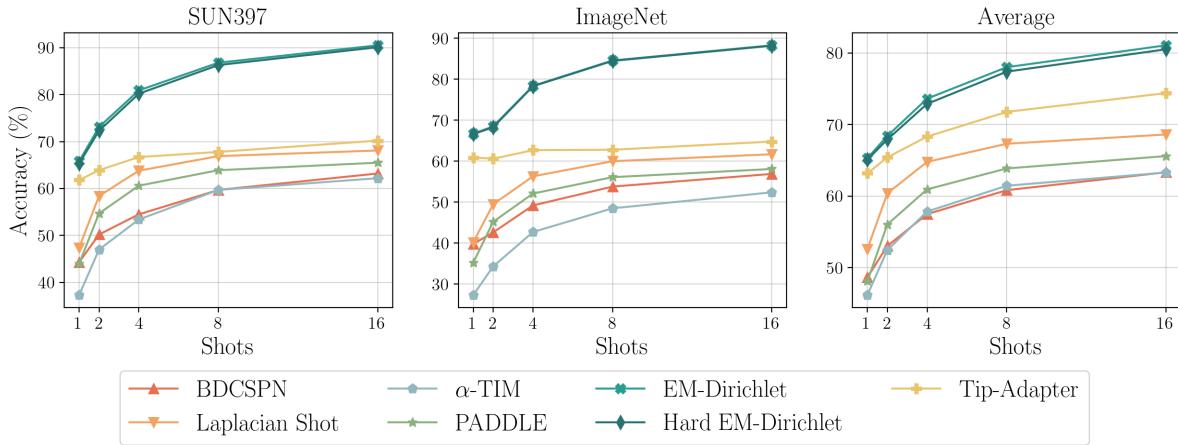


Figure 6.6: Accuracy versus shots for seven methods from Table 6.2 on SUN397, ImageNet, and the average across the 11 datasets.

methods EM-Dirichlet and Hard EM-Dirichlet, other transductive methods (BDCSPN, Laplacian Shot, α -TIM, PADDLE), and the inductive Tip-Adapter method. We did not evaluate CoOp because of the prohibitive time required to run the method, as underlined in Table 6.2. We observe that our method significantly outperforms its closest competitor, TIP, on the challenging SUN397 and ImageNet datasets, as well as on the average of the 11 datasets. This gap gets even wider when the number of shots increases. Complete results for all datasets are given in Figure 6.7.

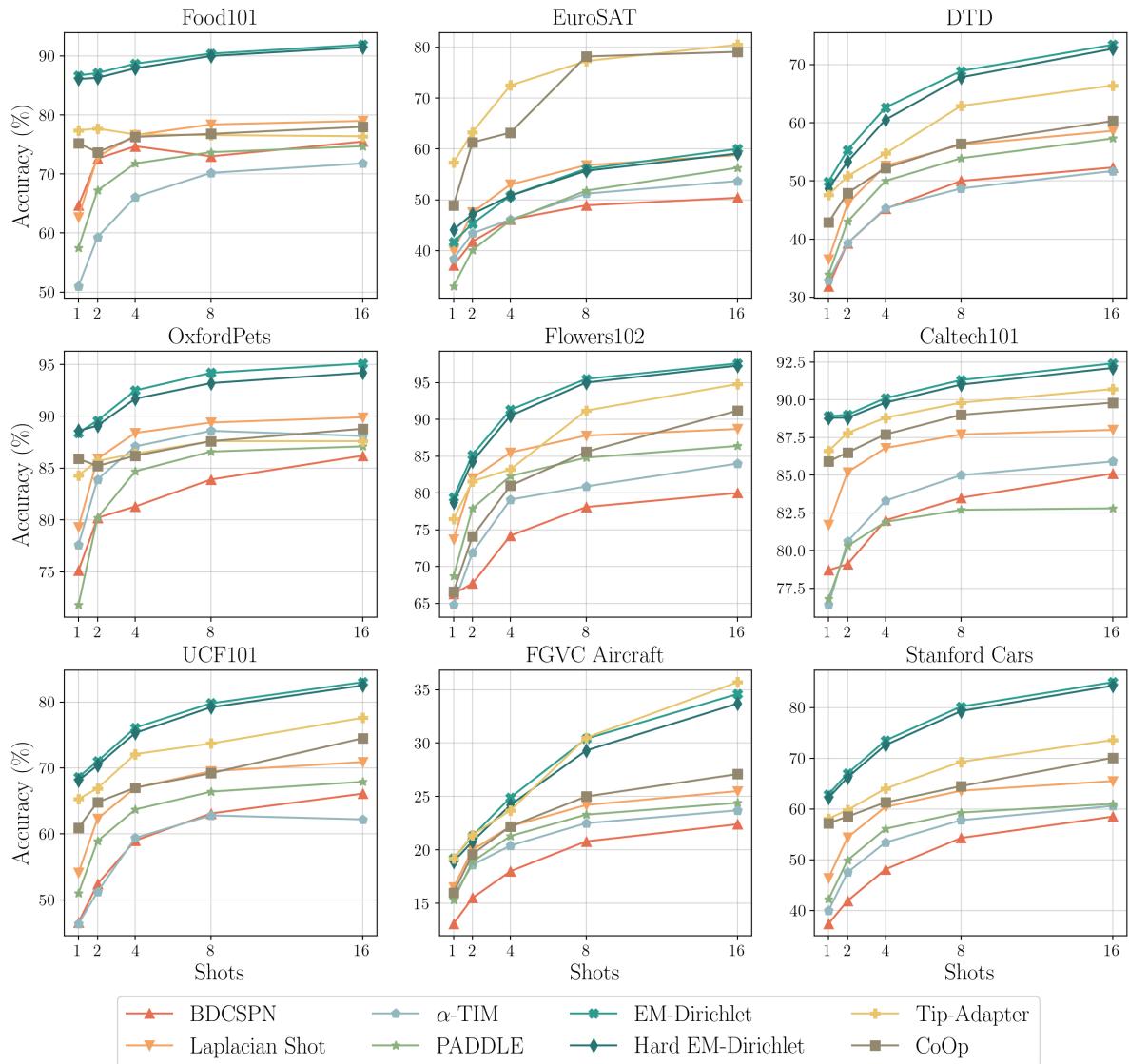


Figure 6.7: Accuracy versus shots for seven methods from Table 6.2 on 9 datasets.

- Chapter 7 -

Conclusion and perspectives

7.1 Summary of contributions

In this thesis, we developed new optimization-based strategies for addressing challenges in image recovery and few-shot classification.

In Chapter 3, we introduced a novel algorithm for constrained, smooth, large-scale problems, called Penalized Majoration-Minimization Subspace algorithm (P-MMS). The main innovation of this algorithm is that it effectively enables the application of MM methods in scenarios with constraints, by successfully integrating a subspace MM algorithm within an inexact exterior penalty framework. To address the well-known ill-conditioning issue in exterior penalty methods, a trust-region-like approach was proposed. We provided an analysis of the algorithm convergence and demonstrated its computational efficiency on two large-scale image recovery applications.

Chapter 4 focuses on 3D image restoration in the context of multi-photon microscopy. A comprehensive restoration pipeline was presented, encompassing all stages from image acquisition to final output. A novel protocol for PSF estimation was developed, distinguishing itself by the use of large beads for calibration. This constitutes a breakthrough in the practice of microscopy imaging where the traditional methods are typically restricted to the use of sub-resolved beads. We addressed this inverse problem by formulating a minimization task incorporating a Gaussian prior and employed an alternating proximal algorithm to solve it. The chapter also introduced a novel restoration methodology, combining a constrained formulation and an additive heteroscedastic noise model. A significant advantage of the proposed method is its ability to perform relatively quick restorations in the computational demanding context of 3D imaging contexts. This makes it a practical tool for physicists and biologists. This was successfully demonstrated using real mouse muscle data, showcasing the method speed and efficacy in producing usable and reliable restoration results for complex 3D images.

In Chapter 5, we used our knowledge on variational methods to explore the recent domain of few-shot learning. We participated in the endeavor of making classification methods applicable to more difficult real-life usecases than those often considered for academic purposes. Given the observation that all transductive few-shot methods are evaluated on unrealistic tasks (i.e., perfectly balanced and with a limited number of different classes), and that these methods are biased to perform well on such tasks, we proposed a new, more realistic evaluation protocol. For the unbalanced classification challenge, a

criterion inspired from Minimum Description Length inference was introduced as well as a simple yet efficient block coordinate descent algorithm. The method performance was validated on datasets such as mini-Imagenet, tiered-Imagenet, and iNat.

In Chapter 6, we extended the discussion on zero-shot and few-shot classification using recent large language-vision models, in particular the popular CLIP one. We proposed the first transductive inference approach in this field. We first designed informative text-vision probability features for the transductive inference. This leads to a classification challenge on the unit simplex which was addressed with a flexible EM-like criterion for Dirichlet mixtures. We proposed an algorithm based on the MM framework to minimize the criterion. In particular, we introduced a new tight majorant for the Dirichlet negative log-likelihood, which has the advantage of having a minimizer that can be expressed in a closed-form.. The chapter concluded by applying our method to both zero-shot and few-shot tasks across 11 different datasets. This demonstrated *i*) the benefits of employing a transductive setting rather than an inductive one for CLIP zero-shot classification, and *ii*) the impact of using probabilities as feature vectors and accurately modeling the data distribution on the unit simplex set with Dirichlet distributions.

7.2 Perspectives

Based on the works exposed in this manuscript, we discuss several potential directions for future work.

Relaxation of the convexity assumption in P-MMS

In Chapter 3, a convexity assumption was imposed on the penalized function at each iteration to guarantee the convergence of the P-MMS algorithm. Relaxing this assumption would allow for the application of the algorithm to a wider range of imaging problems. This could be achieved relying better on the Kurdyka-Łojasiewicz property. The primary challenge, however, lies in the fact that each function within the sequence of penalized functions might fulfill the KL condition with a different reparametrization function.

Further numerical aspects of P-MMS

One of the main drawbacks of the P-MMS algorithm is the need to set the sequences of penalty / precision parameters. Though the choices do not impact the convergence of the algorithm toward the optimum, they do affect its efficiency. It would be useful to develop a more systematic approach to setting these parameters, similarly to what is done for barrier methods [Nocedal and Wright, 2006, Chap. 19, p.572]. Another possibility would be to deploy a GPU-based implementation of the method coupled with an unrolling strategy [Monga et al., 2021, Gharbi et al., 2022].

Real world applications of the few-shot classification developed in Chapter 5

Classification of unbalanced data is frequently encountered in real-world applications, especially in medical imaging where the distribution of samples per class, such as types of tumors, can be highly unbalanced. The preliminary work presented in Appendix 8.B involves applying the method proposed

in Chapter 5 to classify patches in histopathology slides. In this context, the transductive approach is particularly interesting because one can infer from multiple unlabeled patches within a localized area in the slide. Another promising application task for our method is segmentation, which can be regarded as a pixel-level classification task. Here, the unbalance in the number of pixels that correspond to different objects (such as background versus foreground) within an image raises an interesting challenge.

Establishing theoretical bounds on the few-shot performance

It would be interesting to establish theoretical bounds on the best achievable performance with Maximum Likelihood based methods given the number of samples in the query set and the number of shots, for instance under the assumption that the data is distributed according to a Gaussian mixture.

Extensions of few-shot benchmarks

Extending benchmarks to include more complex and realistic settings is a logical extension to Chapter 5. Few-shot benchmarks consider tasks where the classes represented in the query set can be totally unrelated (eg. girafe, micro-wave and tulip) as underlined in [Bennequin et al., 2021]. Presence in large amount of such (easy) tasks during the evaluation might hide the actual performance of the methods on more realistic tasks, comprising more subtle inter-class variations. It would also be pertinent to enlarge the number of classes possibly present in the query set, instead of keeping it fixed to 5.

About the Dirichlet maximum likelihood problem

In Chapter 6, we proposed a new MM algorithm for the minimization of negative Dirichlet log-likelihood. However, the problem of the existence of a minimizer remains open. This question is not so evident and relates to the well-known problem of the existence of Maximum-Likelihood estimators for the exponential family [Wedderburn, 1976]. If we can prove the existence of a minimizer, then a convergence analysis of the algorithm could be conducted. An idea to do so would be to establish a link between the proposed algorithm and Bregman MM strategies.

Transduction with vision-language models for other downstream tasks

In Chapter 6, we focused on the use of vision-language models for zero-shot and few-shot classification. We could extend the transductive methodology to other downstream tasks, as this has not been explored yet. For instance, CLIP-based transductive approaches could be applied to zero-shot segmentation tasks.

Dealing with outliers in the few-shot problem

During this thesis, the potential presence of outliers in the support set was not addressed. However, when dealing with real-world data, considering outliers is crucial, as they can frequently arise due to annotation errors or distributions shifts between two difference source of data. For cases with a reasonably sized support set, one strategy to tackle this issue could involve selecting relevant examples from the support set by incorporating a binary variable into the supervision term of our few-shot model. This

binary variable, to be estimated along with other parameters, would be set to 0 for out-of-distribution examples and 1 otherwise. This idea aligns with the methodology previously investigated in [Boudiaf et al., 2023] for addressing the few-shot open-set recognition problem.

Appendices

8.A A non-convex proximal approach for centroid-based classification

In this work, we propose a novel variational approach for supervised classification based on transform learning. Our approach consists of formulating an optimization problem on both the transform matrix and the centroids of the classes in a low-dimensional transformed space. The loss function is based on the distance to the centroids, which can be chosen in a flexible manner. To avoid trivial solutions or highly correlated clusters, our model incorporates a penalty term on the centroids, which encourages them to be separated. The resulting non-convex and non-smooth minimization problem is then solved by a primal-dual alternating minimization strategy. We assess the performance of our method on a bunch of supervised classification problems and compare it to state-of-the-art methods.

8.A.1 Introduction

Given labeled multidimensional observations split into k classes of data having similar characteristics, the problem of supervised classification consists in learning a classifier which associates each data with a cluster. However, classification of high-dimensional data remains limited by both computational issues and the predictive power of traditional methods, vectors tending to become indiscernible as the dimension grows [Aggarwal, 2005, Houle et al., 2010]. When facing such situations, a common strategy is to map those data to a lower-dimensional subspace without losing essential information regarding the discriminative characteristics of the original variables. A popular approach is to perform a Principal Component Analysis (PCA) prior to classification [Everitt et al., 2011]. However, this approach is not always relevant in practice since it only accounts for the second-order correlations within the data and does not take into account all the available information [Chang, 1983]. A generalization of the PCA approach consists in searching for a low-dimensional transform that enjoys some optimality properties. For instance, one may search for a matrix that maximizes the separation between classes in the transformed space, such as in *Linear Discriminant Analysis* [De la Torre and Kanade, 2006]. Ideally, the clusters in the transformed space should be as far as possible from each other, and the data points compactly distributed within each cluster.

A simple way to tackle the supervised classification problem is to formulate it as a least squares

optimization problem on the transform matrix \mathbf{W} [Feng et al., 2016, Naseem et al., 2010]:

$$\underset{\mathbf{W}}{\text{minimize}} \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|_{\text{F}}^2, \quad (8.\text{A}.1)$$

where $\|\cdot\|_{\text{F}}$ is the Frobenius norm, $\mathbf{X} \in \mathbb{R}^{m \times d}$ is a matrix whose rows correspond to the m samples belonging to the feature space \mathbb{R}^d , and $\mathbf{Y} \in \{0, 1\}^{m \times k}$ denote the matrix of one-hot encoded labels, where $k \geq 2$ is the number of classes. Each row vector $(\mathbf{Y}_i)_{1 \leq i \leq m}$ of \mathbf{Y} contains only one non-zero coefficient, which indicates the cluster of the sample \mathbf{X}_i . Many authors have proposed to add an ℓ_1 norm sparse regularizer on the transformation matrix \mathbf{W} in Problem (8.A.1), therefore retrieving the standard LASSO loss function [Tibshirani, 1996, Majumdar and Ward, 2009, Shao et al., 2018]. However, in [Nie et al., 2010], it was observed that using the Frobenius norm in (8.A.1) makes the approach sensitive to outliers and it was suggested to replace it by an $\ell_{1,2}$ norm. In the same spirit, an ℓ_1 data-fit function was used in [Zhou et al., 2019], leading to the minimization of the sum of two non-smooth convex functions.

A more flexible criterion proposed in [Barlaud et al., 2019] and inspired from the k -means algorithm, optimizes jointly the transform matrix \mathbf{W} and the centroid matrix \mathbf{M} through a data-fidelity term of the general form $(\mathbf{W}, \mathbf{M}) \mapsto f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W})$, where f is a function defined on $\mathbb{R}^{m \times \ell}$. Yet, as highlighted by the authors, such an approach requires adding adequate regularization and penalty terms to break separability and avoid the trivial undesirable solution $(\mathbf{M}, \mathbf{W}) = (0, 0)$.

In this Chapter, we present a new transform-based variational approach for supervised classification. Our contribution is twofold. The first one is to propose a suitable criterion on both the transform matrix and the centroids. Our formulation differs from the one in [Barlaud et al., 2019] as it adds a term to the loss favoring centroid separation. Although the introduction of this term makes the approach more effective, it leads to a loss of the convexity property, thus making the optimization problem more challenging. Secondly, we propose an alternating proximal minimization algorithm to solve this problem. Sub-iterations for computing the involved proximity operator are performed using a primal-dual formulation.

The Chapter is organized as follows. Section 8.A.2 presents the centroid-based approach for classification and the difficulties it raises. In Section 8.A.3 we formulate a novel criterion for separating the centroids in the transformed space. The algorithm we implement to address this problem is described in Section 8.A.4. Finally, comparisons in terms of accuracy of our approach with state-of-the-art methods are conducted in Section 8.A.5, before drawing some conclusions in Section 8.A.6.

8.A.2 Centroid-based classification

Let $(\mathbf{X}_i)_{1 \leq i \leq m}$ be vectors in \mathbb{R}^d containing the rows of data matrix $\mathbf{X} \in \mathbb{R}^{m \times d}$, corresponding to data samples. Centroid based classification consists in mapping \mathbf{X} to a lower dimensional space \mathbb{R}^ℓ using a linear transform $\mathbf{W} \in \mathbb{R}^{d \times \ell}$. We also introduce k *centroid* vectors, each of which can be thought of as the center of the samples $\mathbf{W}^\top \mathbf{X}_i$ belonging to the same cluster. The matrix of centroids $(\mathbf{M}_j)_{1 \leq j \leq k}$ is denoted by $\mathbf{M} = [\mathbf{M}_1, \dots, \mathbf{M}_k]^\top \in \mathbb{R}^{k \times \ell}$.

The centroid-based classification approach amounts to minimizing the following loss function with

respect to the matrix variables \mathbf{W} and \mathbf{M} :

$$(\forall \mathbf{M} \in \mathbb{R}^{k \times \ell}) (\forall \mathbf{W} \in \mathbb{R}^{d \times \ell}) \quad \mathcal{L}(\mathbf{M}, \mathbf{W}) = f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) + g(\mathbf{W}) + h(\mathbf{M}), \quad (8.A.2)$$

where $f: \mathbb{R}^{m \times \ell} \rightarrow]-\infty, +\infty]$, $g: \mathbb{R}^{d \times \ell} \rightarrow]-\infty, +\infty]$ and $h: \mathbb{R}^{k \times \ell} \rightarrow]-\infty, +\infty]$ satisfy the following assumption:

Assumption 8.1. Functions f , g , and h are convex, proper, lower semi-continuous. Moreover, f is row-wise separable, i.e.

$$(\forall \mathbf{Z} = [\mathbf{Z}_1, \dots, \mathbf{Z}_m]^\top \in \mathbb{R}^{m \times \ell}) \quad f(\mathbf{Z}) = \sum_{i=1}^m \varphi(\mathbf{Z}_i), \quad (8.A.3)$$

for some function $\varphi: \mathbb{R}^\ell \rightarrow]-\infty, +\infty]$.

In formulation (8.A.2), f corresponds to the data-fit function which plays a crucial role in learning from the training set (\mathbf{X}, \mathbf{Y}) , while g (resp. h) is a regularization function on \mathbf{W} (resp. \mathbf{M}).

Example 8.1.

- A classical choice for f corresponds to $f = \|\cdot\|_F^2$ [Maggu et al., 2018, Bach and Harchaoui, 2007]. Note that, in this case, the data-fit term in (8.A.2) is reminiscent of the standard k-means approach since

$$f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) = \sum_{j=1}^k \sum_{i \in C_j} \|\mathbf{M}_j - \mathbf{W}^\top \mathbf{X}_i\|_2^2, \quad (8.A.4)$$

where $(C_j)_{1 \leq j \leq k}$ denote the classes. An alternative choice proposed in [Zhou et al., 2019] is $f = \|\cdot\|_1$ and was shown to be more robust to outliers.

- A sparsity-promoting regularization is often employed for \mathbf{W} [Xu et al., 2014b, Wei et al., 2016, Hara and Maehara, 2017], corresponding to $g = \alpha \|\cdot\|_1$, where $\alpha > 0$. It may also be convenient to use an elastic net regularisation $g = \alpha \|\cdot\|_1 + \frac{\beta}{2} \|\cdot\|_F^2$ with $\beta > 0$ to guarantee the existence of a solution to the minimization problem.
- Choices for function h are discussed in the rest of the Chapter.

Given the transform matrix \mathbf{W} and the centroid matrix \mathbf{M} obtained by minimizing (8.A.2), a new sample \mathbf{X}_{m+1} can be assigned to a class j^* , where j^* satisfies

$$j^* \in \operatorname{Argmin}_{j \in \{1, \dots, k\}} \varphi(\mathbf{M}_j - \mathbf{W}^\top \mathbf{X}_{m+1}). \quad (8.A.5)$$

(If such a j^* is not unique, one arbitrarily assigns to \mathbf{X}_{m+1} the class corresponding to the smallest j^* satisfying (8.A.5).)

While model (8.A.2) has the great advantage of leading to a convex optimization problem, one can observe that the trivial solution $(\mathbf{M}, \mathbf{W}) = (0, 0)$ is obtained when f , g are chosen as in Example 8.1 and h is a nonnegative function vanishing at 0. Therefore, function h should be cleverly chosen so to avoid this trivial solution. For instance, Barlaud et al. [Barlaud et al., 2019] circumvented this issue by introducing a regularization h of the form

$$(\forall \mathbf{M} \in \mathbb{R}^{k \times k}) \quad h(\mathbf{M}) = \frac{\rho}{2} \|\mathbf{M} - \mathbf{I}_k\|_F^2, \quad (8.A.6)$$

where $\rho > 0$ is a multiplicative constant. Such a choice for function h penalizes the alignment of the centroids. Although it avoids retrieving the trivial solution, it forces the data to be mapped to vectors of the same dimension $\ell = k$ as the number of classes, which may not be a suitable choice to fully separate the clusters. In the following, we propose a new expression for h which favors the separation of the clusters in a transformed space of arbitrary dimension $\ell \leq m$.

8.A.3 A non-convex formulation to centroid-based classification

8.A.3.1 Separation of the centroids

We propose to resort to the following modified loss function:

$$(\forall \mathbf{M} \in \mathbb{R}^{k \times \ell}) (\forall \mathbf{W} \in \mathbb{R}^{d \times \ell}) \quad \mathcal{L}(\mathbf{M}, \mathbf{W}) = f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) + g(\mathbf{W}) - \gamma h(\mathbf{M}), \quad (8.A.7)$$

where f , g and h satisfy Assumption 8.1, and $\gamma > 0$ is a penalty parameter. We opt for a particular choice of function h which pushes the centroids to be distant from each others, namely

$$(\forall \mathbf{M} \in \mathbb{R}^{k \times \ell}) \quad h(\mathbf{M}) = \sum_{1 \leq i < j \leq k} \|\mathbf{M}_j - \mathbf{M}_i\|_1. \quad (8.A.8)$$

Note that for this choice of function h , Model (8.A.7) is non-convex. In order to handle this term more easily, it is reexpressed as

$$(\forall \mathbf{M} \in \mathbb{R}^{k \times \ell}) \quad h(\mathbf{M}) = \|\mathbf{A}\mathbf{M}\|_1, \quad (8.A.9)$$

where A is a matrix of dimension $q \times k$, where $q = \ell(\ell - 1)/2$.

8.A.3.2 Bounding the problem

The loss function defined in (8.A.7) may have no minimizer. Let us show this, in the particular case when f , g and h are homogeneous functions with the same scaling factor, in the sense that there exists a function $a: \mathbb{R}_+ \rightarrow \mathbb{R}_+$ such that $a(0) = 0$, $\lim_{\eta \rightarrow +\infty} a(\eta) = +\infty$ and, for every $\eta \in \mathbb{R}_+$,

$$\begin{cases} (\forall \mathbf{Z} \in \mathbb{R}^{m \times \ell}) & f(\eta \mathbf{Z}) = a(\eta) f(\mathbf{Z}), \\ (\forall \mathbf{W} \in \mathbb{R}^{d \times \ell}) & g(\eta \mathbf{W}) = a(\eta) g(\mathbf{W}), \\ (\forall \mathbf{M} \in \mathbb{R}^{k \times \ell}) & h(\eta \mathbf{M}) = a(\eta) h(\mathbf{M}). \end{cases} \quad (8.A.10)$$

Then, if there exists a pair (\mathbf{M}, \mathbf{W}) such that

$$f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) + g(\mathbf{W}) - \gamma h(\mathbf{M}) < 0, \quad (8.A.11)$$

Criterion (8.A.7) is unbounded from below. Otherwise, $(\mathbf{M}, \mathbf{W}) = (0, 0)$ is a trivial solution.

These considerations lead us to bound the centroid matrix \mathbf{M} . To do so, we constrain each of the centroids $(\mathbf{M}_j)_{1 \leq j \leq k}$ to lie in a closed ball of radius $\delta > 0$. Taking into account this constraint on the centroids, the minimization problem to be solved reads

$$\underset{(\mathbf{M}, \mathbf{W})}{\text{minimize}} \quad f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) + g(\mathbf{W}) - \gamma h(\mathbf{M}) + \iota_C(\mathbf{M}), \quad (8.A.12)$$

where

$$C = \{\mathbf{M} \in \mathbb{R}^{k \times \ell} \mid (\forall j \in \{1, \dots, k\}) \quad \|\mathbf{M}_j\|_2 \leq \delta\}, \quad (8.A.13)$$

and ι_C denotes the indicator function of C .¹

8.A.3.3 An equivalent formulation

The objective function in (8.A.12) is convex with respect to \mathbf{W} , but it is non-convex with respect to \mathbf{M} . This last issue can be overcome by rewriting the ℓ_1 -norm through its dual norm, i.e.

$$(\forall \mathbf{M} \in \mathbb{R}^{k \times \ell}) \quad h(\mathbf{M}) = \max_{\mathbf{U} \in \mathcal{B}_\infty} \langle \mathbf{A}\mathbf{M}, \mathbf{U} \rangle, \quad (8.A.14)$$

where \mathcal{B}_∞ is the ℓ_∞ -unit ball of $\mathbb{R}^{q \times k}$, and $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product. Therefore, Problem (8.A.12) is equivalent to

$$\underset{(\mathbf{M}, \mathbf{W}, \mathbf{U})}{\text{minimize}} \quad f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) + g(\mathbf{W}) - \gamma \langle \mathbf{A}\mathbf{M}, \mathbf{U} \rangle + \iota_C(\mathbf{M}) + \iota_{\mathcal{B}_\infty}(\mathbf{U}). \quad (8.A.15)$$

The above problem is multi-convex, i.e. convex with respect to each variable \mathbf{M} , \mathbf{W} , or \mathbf{U} , when the others are set. This property suggests to resort to an alternating proximal algorithm for solving Problem (8.A.15), as described in the following section.

8.A.4 Proposed algorithm

8.A.4.1 Proximal alternating algorithm

To address Problem (8.A.15), one could think of using a standard alternating minimization approach. However, it is well-known that such an alternating minimization procedure requires quite restrictive conditions to guarantee its convergence toward a local minimizer (see for example [Chan and Wong, 2000]), even in a convex setting. Instead, we use an alternating proximal algorithm which was initially proposed in [Bolte et al., 2010], for which sound convergence guarantees were established. We first recall the definition of the proximity operator.

Based on the expression of the objective function in Problem (8.A.15), we will also define the following functions:

$$\phi_{\mathbf{U}}(\mathbf{M}, \mathbf{W}) = f(\mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}) + g(\mathbf{W}) - \gamma \langle \mathbf{M}, \mathbf{A}^\top \mathbf{U} \rangle + \iota_C(\mathbf{M}) \quad (8.A.16)$$

$$\phi_{(\mathbf{M}, \mathbf{W})}(\mathbf{U}) = -\gamma \langle \mathbf{A}\mathbf{M}, \mathbf{U} \rangle + \iota_{\mathcal{B}_\infty}(\mathbf{U}). \quad (8.A.17)$$

The chosen algorithm is given below, where λ and ν are parameters of the algorithm:

We can derive the proximity operator of $\nu\phi_{(\mathbf{M}, \mathbf{W})}$ from the projection onto the ball \mathcal{B}^∞ :

$$\text{prox}_{\nu\phi_{(\mathbf{M}, \mathbf{W})}}(\mathbf{U}) = \text{proj}_{\mathcal{B}^\infty}(\mathbf{U} + \nu\gamma\mathbf{A}\mathbf{M}) \quad (8.A.18)$$

However, there is no closed form expression for $\text{prox}_{\lambda\phi_{\mathbf{U}}}$.

¹ $\iota_C(x) = 0$ if $x \in C$, $\iota_C(x) = +\infty$ otherwise.

Algorithm 17: Alternating proximal algorithm

```

input :  $\mathbf{M}_0, \mathbf{W}_0, \mathbf{U}_0, \lambda > 0, \nu > 0$ 
for  $n = 0, 1, \dots$  do
     $(\mathbf{M}_{n+1}, \mathbf{W}_{n+1}) = \text{prox}_{\lambda \phi_{\mathbf{U}_n}}(\mathbf{M}_n, \mathbf{W}_n)$ 
     $\mathbf{U}_{n+1} = \text{prox}_{\nu \phi_{(\mathbf{M}_{n+1}, \mathbf{W}_{n+1})}}(\mathbf{U}_n)$ 
end

```

		texture	sonar	pima	wdbc	banana	magic	satimage	titanic	bupa	AVG
APDNC	Train	87.0	85.4	74.7	94.1	57.3	77.1	78.3	77.3	59.7	76.8
	Test	86.4	72.0	73.0	94.0	54.9	77.2	77.3	77.3	58.2	74.5
Barlaud et al.	Train	72.8	83.1	76.5	88.0	56.0	66.0	74.7	77.6	69.3	73.8
	Test	72.3	68.1	75.5	87.2	54.4	65.7	72.1	77.6	67.8	71.2
NCM	Train	74.5	72.7	73.4	93.9	57.7	77.1	78.7	75.4	60.0	73.7
	Test	73.7	70.2	72.8	93.7	57.4	76.9	78.4	74.6	60.0	73.1

Table 8.1: Classification rate of APDNC compared to the state-of-the-art.

8.A.4.2 Computation of the proximal operators

For every $(\bar{\mathbf{M}}, \bar{\mathbf{W}}) \in \mathbb{R}^{k \times d} \times \mathbb{R}^{d \times \ell}$, we must compute

$$\text{prox}_{\lambda \phi_{\mathbf{U}}}(\bar{\mathbf{M}}, \bar{\mathbf{W}}) = \underset{\mathbf{M}, \mathbf{W}}{\operatorname{argmin}} \phi_{\mathbf{U}}(\mathbf{M}, \mathbf{W}) + \frac{1}{2\lambda} (\|\mathbf{M} - \bar{\mathbf{M}}\|_{\text{F}}^2 + \|\mathbf{W} - \bar{\mathbf{W}}\|_{\text{F}}^2).$$

Function $\phi_{\mathbf{U}}$ can be split as $\phi_{\mathbf{U}}(\mathbf{M}, \mathbf{W}) = f(L(\mathbf{M}, \mathbf{W})) + r(\mathbf{M}, \mathbf{W})$, where $L: (\mathbf{M}, \mathbf{W}) \mapsto \mathbf{Y}\mathbf{M} - \mathbf{X}\mathbf{W}$ and r is a separable function of its two arguments \mathbf{M} and \mathbf{W} . L is a linear operator whose adjoint operator is

$$(\forall \mathbf{Z} \in \mathbb{R}^{m \times \ell}) \quad L^*: \mathbf{Z} \mapsto (\mathbf{Y}^\top \mathbf{Z}, -\mathbf{X}^\top \mathbf{Z}). \quad (8.A.19)$$

and whose spectral norm is $\|L\|_{\text{S}} = \|\mathbf{X}\mathbf{X}^\top + \mathbf{Y}\mathbf{Y}^\top\|_{\text{S}}^{1/2}$. Since the objective function involved in the definition of the proximity operator is strongly convex, we can employ an accelerated Primal-Dual algorithm [Chambolle and Pock, 2011, Alg. 2] to compute $\text{prox}_{\lambda \phi_{\mathbf{U}}}(\bar{\mathbf{M}}, \bar{\mathbf{W}})$. This algorithm makes use of the conjugate function f^* of f , and two parameters τ_p and σ_p which are dynamically updated along the iterations.

8.A.4.3 Overall algorithm

Algorithm 18 is embedded as a sub-iteration in Algorithm 17. The resulting algorithm will be subsequently referred to as APDNC (alternating primal-dual non-convex minimization). Note that the initial variable \mathbf{M}_0 and \mathbf{W}_0 in Algorithm 18 are warm-restarted after each iteration of the main loop of Algorithm 17. This has been observed to be beneficial to the convergence speed.

Algorithm 18: Accelerated primal-dual algorithm

```

input :  $\bar{\mathbf{M}}, \bar{\mathbf{W}}, \mathbf{U}, \mathbf{M}_0, \mathbf{W}_0, \mathbf{V}_0, \tau_0, \lambda, \gamma$ 
initialization :  $\sigma_0 = \frac{1}{\tau_0 \|L\|_S^2}$ 
for  $p = 0, 1, \dots$  do
     $\theta_p = \left( \sqrt{1 + \frac{2\tau_p}{\lambda}} \right)^{-1}$ 
     $\mathbf{M}_{p+1} = \text{proj}_C \left( \frac{\lambda}{1+\lambda} (\mathbf{M}_p + \gamma \mathbf{A}^\top \mathbf{U} - \tau_p \mathbf{Y}^\top \mathbf{V}_p - \lambda^{-1} \bar{\mathbf{M}}) \right)$ 
     $\mathbf{W}_{p+1} = \text{prox}_{\frac{\lambda\tau_p}{1+\lambda} g} \left( \frac{\lambda}{1+\lambda} (\mathbf{W}_p + \tau_p \mathbf{X}^\top \mathbf{V}_p - \lambda^{-1} \bar{\mathbf{W}}) \right)$ 
     $\mathbf{Z}_p = L((1 + \theta_p)(\mathbf{M}_{p+1}, \mathbf{W}_{p+1}) - (\mathbf{M}_p, \mathbf{W}_p))$ 
     $\tau_{p+1} = \theta_p \tau_p, \sigma_{p+1} = \frac{\sigma_p}{\theta_p}$ 
     $\mathbf{V}_{p+1} = \text{prox}_{\sigma_{p+1} f^*}(\mathbf{V}_p + \sigma_{p+1} \mathbf{Z}_p)$ 
end

```

8.A.5 Numerical experiments

In this section, we evaluate the performance of our method on a set of supervised classification problems and we compare it to state-of-the-art methods.

Settings. The experimental results we present hereafter were obtained using $f = \|\cdot\|_1$ and $g = \alpha \|\cdot\|_1 + \frac{\beta}{2} \|\cdot\|_F^2$. For all our experiments, we set the parameters of Algorithm 18 to $\nu = 1$ and the regularization parameters to $\alpha = 10^{-3}$, $\beta = 10^{-6}$. Parameter λ in the proximity operator, and penalty parameter γ on the centroid-separation term were chosen to reach the best classification performance for each dataset. Regarding the initialization of the primal variables, we set \mathbf{W}_0 by performing a PCA on \mathbf{X} , and then we computed \mathbf{M}_0 by using the centroids of the transformed data. The dual variables were initialized as $\mathbf{V}_0 = \text{sign}(\mathbf{Y}\mathbf{M}_0 - \mathbf{X}\mathbf{W}_0)$, $\mathbf{U}_0 = \text{sign}(\mathbf{A}\mathbf{M}_0)$.

Results. We used several datasets from Knowledge Extraction Evolutionary Learning (KEEL) [Alcalá-Fdez et al., 2011], which include both simulated and real-word datasets. We used the 10-fold version, which allows us to perform cross-validation on our model. We compared our method with two state-of-the-art centroid-based methods, namely the Nearest Matrix Classification method (NCM) [Mensink et al., 2013] and the method of Barlaud et al. [Barlaud et al., 2019, Alg. 7]. For more comparisons, refer to [Suárez-Díaz et al., 2021]. Table 8.1 reports the obtained results. One can observe that our method gives comparable results to its competitors and even outperforms them on some datasets.

8.A.6 Conclusion

We have proposed a sound variational formulation of centroid-based classification in a transformed domain. The resulting non-convex optimization problem has been tackled with recent proximal techniques. The performance obtained on standard datasets show the good performance of our approach. An advantage of this method is that it allows a flexible choice for the data fit term and the regularisation ones, which could be useful when dealing with difficult (e.g., corrupted) datasets.

8.B A transductive few-shot learning approach for classification of digital histopathological slides from liver cancer

This paper presents a new approach for classifying 2D histopathology patches using few-shot learning. The method is designed to tackle a significant challenge in histopathology, which is the limited availability of labeled data. By applying a sliding window technique to histopathology slides, we illustrate the practical benefits of transductive learning (i.e., making joint predictions on patches) to achieve consistent and accurate classification. Our approach involves an optimization-based strategy that actively penalizes the prediction of a large number of distinct classes within each window. We conducted experiments on histopathological data to classify tissue classes in digital slides of liver cancer, specifically hepatocellular carcinoma. The initial results show the effectiveness of our method and its potential to enhance the process of automated cancer diagnosis and treatment, all while reducing the time and effort required for expert annotation.

8.B.1 Introduction

In clinical settings, histopathology images are a critical primary source of information for pathologists to perform cancer diagnostics and choose treatment strategies. With the widespread adoption of digital pathology, it has become a standard practice to digitize histology slides into high-resolution images called Whole Slide Images (WSIs). WSIs have initiated a new era offering considerable opportunities for using AI assistance systems [Dimitriou et al., 2019, Xiang et al., 2021]. In particular, supervised deep learning methods based on conventional neural networks (CNNs) have made great strides in cancer research [Shin et al., 2016, Saillard et al., 2020]. However, the success of classical supervised learning approaches depends on the availability of extensive annotated training data. Unlike natural images, which can be annotated via crowd-sourcing, histopathology necessitates expert pathologists' accurate annotations of gigapixel-sized images. Due to the time-consuming nature of the labeling process, histopathology datasets tend to be limited in size, which poses significant challenges for training machine learning models [Cooper et al., 2023]. Moreover, WSIs can exhibit variability due to staining techniques, tissue preparation, and image quality [Van Den Bent, 2010], affecting supervised model performance. Furthermore, supervised learning models can encounter difficulties when confronted with imbalanced data, a common scenario in histopathology. Non-uniform class distribution may produce biased results and compromise the model performance [Johnson and Khoshgoftaar, 2019].

Few-shot learning methods address the limitations found in traditional supervised learning techniques, providing efficient models capable of generalizing from a small set of labeled examples. These methods not only prove to be scalable, but also significantly reduce costs and time consumption.

Transductive few-shot learning [Martin et al., 2022, Veilleux et al., 2021], a particularly appealing category within this field, has a distinct advantage. Unlike supervised classification methods, which often treat each data sample independently, transductive methods make predictions on a set of samples collectively. This is especially useful when dealing with localized regions in medical imaging. It allows us to leverage homogeneity and spatial coherence across multiple patches in such a region to enhance the classification accuracy and reliability.

In this work, we introduce a novel transductive few-shot learning approach for histopathological image classification. To our knowledge, it is the first of this kind in the field [Szołomicka and Markowska-Kaczmar, 2023]. Our main contributions are summarized below.

- We apply a sliding window technique to WSIs, establishing a practical scenario where the advantages of transductive few-shot learning are clearly demonstrated.
- Inspired by previous work [Martin et al., 2022], we develop an optimization-based method for few-shot classification of histopathological patches.
- We validate our approach by tests on the most frequent liver cancer (i.e., hepatocellular carcinoma, HCC), showcasing its effectiveness and confirming its high potential for practical application.

The paper is organized as follows. Section 8.B.2 describes the medical context. In Section 8.B.3 we detail the few-shot methodology and the proposed algorithm. Finally, Section 8.B.4 is dedicated to the experimental results and Section 8.B.5 to the conclusion.

8.B.2 Medical multiclass problem

We used HCC WSIs stained with HES (Hemaloxylin-Eosin-Saffron) and digitized at $40\times$ magnification. More precisely, we aim to classify local tissues into the following five classes:

1. Non-Tumor Liver (NT): Liver sections that are not affected by HCC but may be affected by cirrhosis.
2. Hemorragic tissue (RE): a non-tumoral pattern characterized by blood cell suffusion.
3. Tumor tissue with macro-trabecular architecture (AM): An aggressive pejorative tumor type characterized by trabeculae of more than ten cells thick.
4. Tumor tissue with Vessels Encapsulating Tumor Clusters architecture (VE): An aggressive pejorative tumor type characterized by tumor cells arranged in small clusters and surrounded by endothelial cells.
5. Conventional trabecular architecture (AN): A non-pejorative tumoral pattern commonly found in HCC patients.

The distinction between tumor and non-tumor areas and the evaluation of pejorative tumor areas provide insightful information to medical doctors.

In this paper, 28 patients from a previously formed cohort of 108 patients with HCC were selected from usable HES slides from Kremlin-Bicêtre Hospital, France, and manually annotated by two skilled pathologists in the five above categories. Annotated WSIs were then tiled into 1728×1728 patches. Data distribution per class is displayed in Table 8.2.

Class	NT	RE	AM	VE	AN
Percentage	26%	14%	8%	12%	40%

Table 8.2: Data distribution per class.

8.B.3 Proposed method

8.B.3.1 Problem formulation

Few-shot methods typically involve a two-step process [Chen et al., 2019]: first, a neural network, pre-trained on a comprehensive and generic dataset, extracts features from the images of interest. Then, a specifically designed classifier is applied to these extracted features to perform the classification task.

We start by introducing the notation for the few-shot classification challenge at hand. The pre-trained network encoder, denoted by Φ , is crucial for feature extraction. Typically, it has been trained on a dataset $\mathcal{D}_{\text{base}}$ encompassing a broad spectrum of images, potentially inclusive of various WSIs from a multitude of organs and medical facilities. Still, it may not precisely encapsulate the exact categories of our specific classification tasks.

The few-shot dataset consists of N images spanning across K distinct classes. In our context of few-shot classification for histopathological images, K equals 5. Within the dataset, a subset, referred to as the *support set* with index set $\mathbb{S} \subseteq \{1, \dots, N\}$, encompasses the feature samples $(\mathbf{x}_n)_{n \in \mathbb{S}}$ and their respective one-hot-encoded labels $(\mathbf{y}_n)_{n \in \mathbb{S}}$.² The support set is constituted by *s-shots* (labeled examples) for each class. In contrast, the *query set* with indices in $\mathbb{Q} = \{1, \dots, N\} \setminus \mathbb{S}$, comprises a batch of unlabeled samples $(\mathbf{x}_n)_{n \in \mathbb{Q}}$.

The goal is to accurately predict the labels for the samples of the query set under the supervision of the support set. To achieve this, the representations $(\mathbf{z}_n = \Phi(\mathbf{x}_n))_{1 \leq n \leq N}$ generated by the feature extractor are fed into our few-shot classifier.

8.B.3.2 Transductive methodology

One of the primary advantages of few-shot learning methodologies, when contrasted with traditional supervised learning techniques, lies in their ability to collectively infer from an entire batch of $|\mathbb{Q}| > 1$ query instances simultaneously rather than evaluating each instance independently. In the lexicon of few-shot learning, this methodology is referred to as *transductive* learning [Bronskill et al., 2020, Hu et al., 2021]. Transductive few-shot methods are designed to make joint predictions for the entire batch of query samples within each specific few-shot task. This approach takes full advantage of the statistical properties inherent to the query set of a task, employing shared information across instances to enhance generalization and accuracy. Empirical studies have demonstrated that batch-based inference on unlabeled instances, as opposed to individual sample evaluation, results in substantial improvements in prediction accuracy [Joachims, 1999].

In the field of microscopy analysis, where spatial pattern recognition is crucial, transductive few-shot approaches exhibit significant potential. Commonly, in a single WSI, it is observed that architectures belonging to the same class tend to cluster spatially, forming homogeneous regions. To leverage this spatial coherence, our strategy involves selecting a window of dimensions $S \times S$ on the microscope slide, as depicted in Figure 8.1. Each window comprises overlapping mini-patches, each of dimensions $s \times s$, constituting the query set for our few-shot task. The underlying assumption here is that each window encapsulates a few (typically, one or two) distinct classes, allowing each mini-patch to serve as

²For every $n \in \mathbb{S}$ and $k \in \{1, \dots, K\}$, $y_{n,k} = 1$ if \mathbf{x}_n is an instance of class k , and $y_{n,k} = 0$ otherwise.

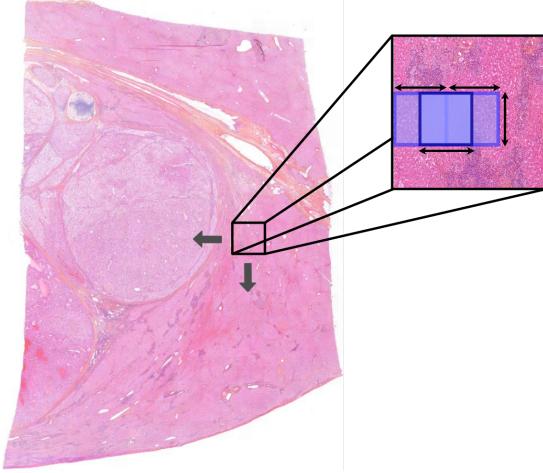


Figure 8.1: Scanning of the slide with a sliding window.

an additional (unlabeled) instance of these classes. By sliding the window across the entire WSI, we facilitate comprehensive predictions across its entirety.

8.B.3.3 Minimization problem

Our method estimates the optimal class assignments of each small patch within the window while limiting the number of predicted classes, thereby acknowledging the spatial coherence in such samples.

By refining the method presented in [Martin et al., 2022], we approach the few-shot classification challenge through a minimization problem, seeking optimal solutions for the one-hot-encoded assignments $\mathbf{U} = (\mathbf{u}_n)_{1 \leq n \leq |\mathbb{Q}|} \in (\Delta_K)^{|\mathbb{Q}|}$ and the class centroids $\mathbf{W} = (\mathbf{w}_k)_{1 \leq k \leq K} \in (\mathbb{R}^d)^K$, where Δ_K represents the unit simplex set in \mathbb{R}^K . The problem is mathematically formulated as

$$\begin{aligned} & \underset{\mathbf{U}, \mathbf{W}}{\text{minimize}} \quad f(\mathbf{U}, \mathbf{W}) + g(\mathbf{U}) + \lambda h(\mathbf{U}), \\ & \text{subject to} \quad (\forall n \in \mathbb{Q}) \quad \mathbf{u}_n \in \Delta_K, \\ & \quad (\forall n \in \mathbb{S}) \quad \mathbf{u}_n = y_n, \end{aligned} \tag{8.B.1}$$

with λ is a positive regularization parameter. Here, f represents the data-fidelity term, reflecting the assumption that the data follows a multivariate Gaussian distribution and integrating supervision from the support set. Formally, we define

$$f(\mathbf{U}, \mathbf{W}) = \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} (\mathbf{w}_k - \mathbf{z}_n)^\top \hat{\mathbf{S}}_k (\mathbf{w}_k - \mathbf{z}_n) - \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N u_{n,k} \ln \det(\hat{\mathbf{S}}_k) \tag{8.B.2}$$

where, for every $k \in \{1, \dots, K\}$, $\hat{\mathbf{S}}_k$ is a symmetric positive matrix corresponding to a sparse approximation of inverse of the empirical covariance matrix of class k , computed from the support set with a Graphical Lasso approach [Friedman et al., 2008]. In addition, g represents an entropic barrier on the

assignments, facilitating closed-form updates in the forthcoming algorithm. It is expressed as

$$g(\mathbf{U}) = \sum_{k=1}^K \sum_{n \in \mathbb{Q}} u_{n,k} \ln u_{n,k}. \quad (8.B.3)$$

Finally, the penalty function h is central to our approach: it acts as a partition complexity term, encouraging a minimal number of classes to be predicted within the window:

$$h(\mathbf{U}) = - \sum_{k=1}^K \pi_k \ln(\pi_k), \quad (8.B.4)$$

where, for every $k \in \{1, \dots, K\}$, $\pi_k = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}$ denotes the proportion of samples of class k in the query set.

8.B.3.4 Algorithm

To address the minimization problem outlined in Equation (8.B.1), we propose an algorithm that alternates minimization steps with respect to the variables \mathbf{U} and \mathbf{W} . Our iterative approach, detailed in Algorithm 19, shares similarities with the technique presented in [Martin et al., 2022], the primary distinction being the introduction of inverse covariance matrices. Given these similarities, we direct the reader to [Martin et al., 2022] for more details on our methodology and the convergence guarantees of the algorithm.

Algorithm 19: PADDLE-Cov

Initialize $\mathbf{W}^{(0)}$ as the means computed on the support set and for all $k \in \{1, \dots, K\}$,

$$\pi_k^{(0)} = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}^{(0)}.$$

for $\ell = 1, 2, \dots$, **do**

$$\mathbf{u}_n^{(\ell)} = \text{softmax} \left(\left(-\frac{1}{2} (\mathbf{w}_k - \mathbf{z}_n)^\top \hat{\mathbf{S}}_k (\mathbf{w}_k - \mathbf{z}_n) + \frac{1}{2} \ln \det(\hat{\mathbf{S}}_k) + \frac{\lambda}{|\mathbb{Q}|} \ln \pi_k^{(\ell)} \right)_k \right), \forall n \in \mathbb{Q},$$

$$\mathbf{w}_k^{(\ell+1)} = \frac{\sum_{n=1}^N u_{n,k}^{(\ell+1)} \mathbf{z}_n}{\sum_{n=1}^N u_{n,k}^{(\ell+1)}}, \forall k \in \{1, \dots, K\},$$

$$\pi_k^{(\ell+1)} = \frac{1}{|\mathbb{Q}|} \sum_{n \in \mathbb{Q}} u_{n,k}^{(\ell+1)}, \quad \forall k \in \{1, \dots, K\}.$$

8.B.4 Experiments

8.B.4.1 Experimental setting

In our experimental setup, we leverage the pre-trained model from [Ciga et al., 2022], trained on diverse histopathological images. We structure our few-shot tasks using a sliding window of dimensions 5184×5184 , containing mini-patches of size 1728×1728 downsampled to a resolution of 512×512 . This results in query sets of 25 samples each. The support set comprises the annotated patches of the 28 train patients,

	Accuracy (%)	F1-score (%)
SimpleShot [Wang et al., 2019]	48.9	46.4
Baseline [Chen et al., 2019]	74.4	72.0
α -TIM [Veilleux et al., 2021]	56.0	56.9
PADDLE [Martin et al., 2022]	51.0	48.9
PADDLE-Cov ($\lambda = 0$)	77.3	73.8
PADDLE-Cov	79.3	75.5

Table 8.3: Evaluation of our approach against other few-shot methods for histopathological patch classification regarding accuracy and F1-score.

and we set the penalty parameter λ to 1250 using validation slides. Preprocessing includes Reinhard color normalization to mitigate staining variability [Reinhard et al., 2001].

8.B.4.2 Results

Validation on annotated test data

Our initial evaluation focuses on the entire collection of labeled patches, which we refer to as windows, from the test set slides from 13 patients. Notably, each window is exclusively composed of mini-patches associated with a single class. In this context, we benchmark our approach against two *inductive* few-shot methodologies, SimpleShot [Wang et al., 2019] and Baseline [Chen et al., 2019], which conduct inference on each mini-patch independently, as well as the state-of-the-art transductive method α -TIM [Veilleux et al., 2021]. In addition we provide an ablation of the terms in our classifying objective (8.B.1), evaluating the original PADDLE method (with identity covariances) and the PADDLE-Cov for $\lambda = 0$. The outcomes of this comparative analysis are given in Table 8.3. Our method surpasses the other approaches, highlighting the benefits of using an appropriate Gaussian metric and of transductive inference.

Inference on a Whole Slide Image (WSI)

In our second evaluation, we aim to compare the predictions made by our 5-class few-shot classifier trained on 28 patients with those of a 3-class fully supervised model on WSIs. The 3-class fully supervised model is a CNN based on ResNet34, which was trained using 800K patches, based on the 87 patients cohort, to classify tissues as non-tumor (NT), non-pejorative tumor (AN), or pejorative tumor (VE+AM). Creating a 5-class supervised model was hardly achievable, as two classes are notably under-represented. Figures 8.2 and 8.3 display the predictions of both models on a WSI where the colored squares represent the annotations (ground truth) made by the pathologists.

Both models reliably identify non-tumoral (green squares) and pejorative regions (orange/brown squares), while the conventional trabecular architecture (yellow squares) is better detected by the few-shot model on the WSI in Figure 8.2. Moreover, training the few-shot model on 5 classes enables detailed detection of the architectures, which the 3-class model can not achieve. In particular, our model accurately distinguishes the VE architecture within the pejorative regions, providing 100% certainty in differentiating VE from AM in both WSIs. Additionally, it detected hemorrhagic regions (RE, purple squares), which were logically misclassified by the fully supervised model in Figure 8.2. Lastly, the

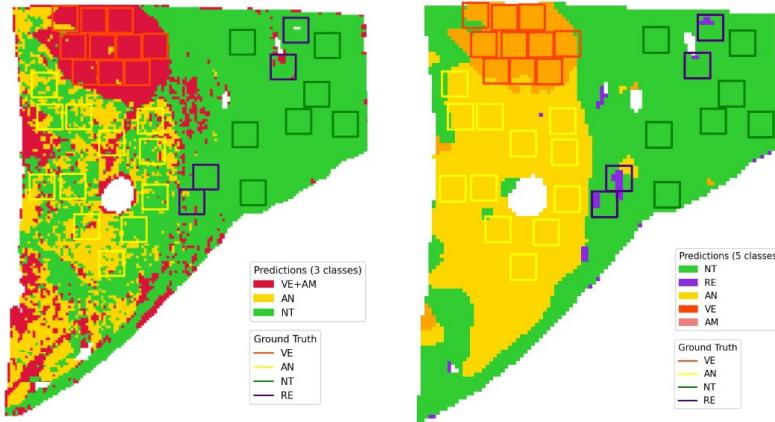


Figure 8.2: (left) Predictions made by the 3-class fully supervised model. (right) Predictions made by the few-shot 5-class model.

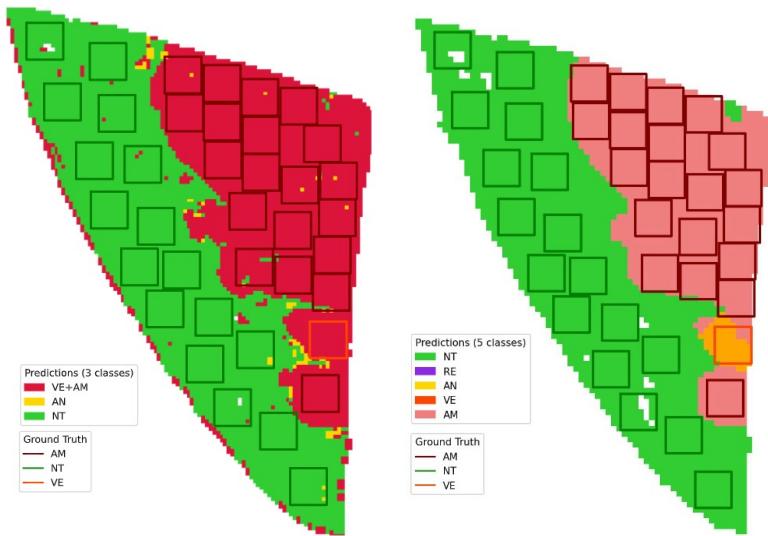


Figure 8.3: (left) Predictions made by the 3-class fully supervised model. (right) Predictions made by the few-shot 5-class model.

5-class model exhibits remarkable proficiency in defining homogeneous regions across the entire WSIs, unlike the 3-class model, which analyzes individual patches independently. The 5-class model contextual understanding allows for consideration of interdependencies between neighboring patches, leading to a more cohesive interpretation of the data.

8.B.5 Conclusion

To wrap up, we have introduced an innovative transductive few-shot learning method tailored to classify histopathological images. This approach effectively overcomes significant obstacles, notably data scarcity and class imbalance. Our study emphasizes the adaptability and promise of our method in the domain of biomedical imaging. Its success not only emphasizes the feasibility of our approach in tackling practical challenges but also paves the way for its wider application in various medical imaging scenarios.

8.B.6 Ethics approval

The study conformed to the General Data Protection Regulation (GDPR) and was approved by the Institutional Review Board of Mondor Hospital (IRB#00011558) (notification number: 2022-135).

8.B.7 Acknowledgments

We would like to thank Dr Laura Claude of the Department of Pathology of the CHU de Rouen, the surgical team of the Centre Hépato-Biliaire of the Hôpital Paul Brousse, and the technicians of the Department of Pathology of the Hôpital Bicêtre.

List of Figures

2.1	Illustration of a k -way s -shot classification task.	30
2.2	Illustration of the pretraining + classifier few-shot paradigm.	31
2.3	Illustration of the subdifferential	37
2.4	Exemples of a Majorization-Minimization update.	42
2.5	An illustration of the majorants obtained with the descent Lemma.	45
2.6	How local curvature can improve majorant's accuracy.	46
3.1	Weakly convex functions satisfy the assumption on the coercivity of the penalized function.	73
3.2	Visual results obtained with the P-MMS algorithm on the tight frame deconvolution problem.	81
3.3	Distance to the optimum versus time, for the tight frame deconvolution problem.	82
3.4	Illustration of PET imaging system.	82
3.5	Illustration of the new majorant derived for the Anscombe data-fit term.	85
3.6	Reconstruction results for the PET problem.	86
3.7	Distance to the optimum versus time, for the PET reconstruction problem.	86
4.1	Visualisation of a real $1\mu\text{m}$ polystyrene bead in a liquide.	94
4.2	Accuracy at estimating the PSF kernel as a function of the parameter in the generalized exponential distribution.	103
4.3	Estimated FWHM and angles for beads in a water solution.	104
4.4	SNR level reached with our approach and with a penalized least-squares approach	111
4.5	Evolution of the SNR with the iterations of the algorithm.	112
4.6	FWHM and angles estimated for the beads in the mouse muscle.	113
4.7	Profiles of the PSF kernels estimated.	114
4.8	Linear regression for the estimation of noise parameters.	115
4.9	Visualisation of the 3D restoration of a mouse muscle.	116
5.1	Overview of the proposed realistic transductive few-shot framework.	120
5.2	Relaxation of the label cost function in 2D.	123
5.3	Few-shot accuracy as a function of the number of effective classes, for <i>mini</i> and <i>tiered-ImageNet</i> , with a ResNet18.	129
5.4	Few-shot accuracy as a function of the number of effective classes, for <i>mini</i> and <i>tiered-ImageNet</i> , with a WRN28-10 network.	130
5.5	Few-shot accuracy as a function of the number of effective classes, for <i>iNat</i> .	131

5.6	Time before convergence of the proposed method.	132
5.7	Evolution of the optimal parameter as a function of the size of the query set.	134
5.8	Evolution of the accuracy as a function of λ	135
6.1	Summary of our approach for transductive few-shot CLIP classification	139
6.2	Illustration of the bipartite matching for class assignment	145
6.3	Illustration of Dirichlet distributions	151
6.4	CLIP pre-training and zero-shot classification	151
6.5	Zero-shot performance as a function of the size the query set	153
6.6	Results in the few-shot setting (1)	154
6.7	Results in the few-shot setting (2)	155
8.1	Scanning of the slide with a sliding window.	170
8.2	Predictions made by the fully supervised model and the few-shot model (1)	173
8.3	Predictions made by the fully supervised model and the few-shot model (2)	173

List of Tables

2.1	Summary of the vocabulary used in few-shot classification. In the typical (artificial) tasks considered in the literature, the support set is constituted of an equal number of labeled samples of each class. This number is then called <i>shot</i> . We might relax this assumption across the manuscript.	30
5.1	Results for the few-shot inference on <i>mini</i> -Imagenet and <i>tiered</i> -Imagenet.	128
5.2	Results for the few-shot inference on <i>iNat</i> .	130
5.3	Ablation study on the partition complexity term.	131
6.1	Accuracy for zero-shot CLIP clustering	149
6.2	Accuracy for few-shot CLIP methods	150
6.3	Time before convergence: comparison of the two majorants	152
8.1	Classification rate of APDNC compared to the state-of-the-art.	165
8.2	Data distribution per class.	168
8.3	Evaluation of our approach against other few-shot methods for histopathological patch classification regarding accuracy and F1-score.	172

Bibliography

- [Abboud et al., 2017] Abboud, F., Chouzenoux, E., Pesquet, J.-C., Chenot, J.-H., and Laborelli, L. (2017). Dual block-coordinate forward-backward algorithm with application to deconvolution and deinterlacing of video sequences. *Journal of Mathematical Imaging and Vision*, 59(3):415–431.
- [Abergel and Moisan, 2017] Abergel, R. and Moisan, L. (2017). The Shannon total variation. *Journal of Mathematical Imaging and Vision*, 59(2):341–370.
- [Afonso et al., 2010a] Afonso, M. V., Bioucas-Dias, J. M., and Figueiredo, M. A. T. (2010a). An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems. *IEEE Transactions on Image Processing*, 20(3):681–695.
- [Afonso et al., 2010b] Afonso, M. V., Bioucas-Dias, J. M., and Figueiredo, M. A. T. (2010b). A fast algorithm for the constrained formulation of compressive image reconstruction and other linear inverse problems. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4034–4037. IEEE.
- [Aggarwal, 2005] Aggarwal, C. C. (2005). On k-anonymity and the curse of dimensionality. In *Proceedings of the 31st Very Large Data Bases Conference*, volume 5, pages 901–909, Trondheim, Norway.
- [Albert, 1972] Albert, A. E. (1972). *Regression and the Moore-Penrose pseudoinverse*. Academic Press.
- [Alcalá-Fdez et al., 2011] Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., and Herrera, F. (2011). Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic and Soft Computing*, 17.
- [Ali et al., 2019] Ali, M., Gao, J., and Antolovich, M. (2019). Parametric classification of Bingham distributions based on Grassmann manifolds. *IEEE Transactions on Image Processing*, 28(12):5771–5784.
- [Allain et al., 2006] Allain, M., Idier, J., and Goussard, Y. (2006). On global and local convergence of half-quadratic algorithms. *IEEE Transactions on Image Processing*, 15(5):1130–1142.
- [Anscombe, 1948] Anscombe, F. J. (1948). The transformation of Poisson, binomial and negative-binomial data. *Biometrika Trus*, 35(3/4):246–254.
- [Antoni et al., 2023] Antoni, J., Idier, J., and Bourguignon, S. (2023). A Bayesian interpretation of the L-curve. *Inverse Problems*, 39(6):065016.

- [Attouch et al., 2010] Attouch, H., Bolte, J., Redont, P., and Soubeyran, A. (2010). Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka-Łojasiewicz inequality. *Mathematics of Operations Research*, 35(2):438–457.
- [Avron et al., 2010] Avron, H., Sharf, A., Greif, C., and Cohen-Or, D. (2010). ℓ_1 -sparse reconstruction of sharp point set surfaces. *ACM Transactions on Graphics (TOG)*, 29(5):1–12.
- [Bach and Harchaoui, 2007] Bach, F. and Harchaoui, Z. (2007). Diffrac: a discriminative and flexible framework for clustering. *Advances in Neural Information Processing Systems (NeurIPS)*, 20:49–56.
- [Bahmani et al., 2013] Bahmani, S., Raj, B., and Boufounos, P. T. (2013). Greedy sparsity-constrained optimization. *Journal of Machine Learning Research (JMLR)*, 14(Mar):807–841.
- [Barlaud et al., 2019] Barlaud, M., Chambolle, A., and Caillau, J.-B. (2019). Robust supervised classification and feature selection using a primal-dual method. *arXiv preprint arXiv:1902.01600*.
- [Bauckhage, 2020] Bauckhage, C. (2020). NumPy/sciPy recipes for data science: Projections onto the standard simplex. Technical report, University of Bonn, Tech. Rep.
- [Bauschke and Combettes, 2011] Bauschke, H. H. and Combettes, P. L. (2011). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer.
- [Bauschke and Combettes, 2019] Bauschke, H. H. and Combettes, P. L. (2019). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer New York, 2nd, corrected printing edition.
- [Beck and Teboulle, 2009] Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.
- [Beck and Tetruashvili, 2013] Beck, A. and Tetruashvili, L. (2013). On the convergence of block coordinate descent type methods. *SIAM Journal on Optimization*, 23(4):2037–2060.
- [Belzungue, 2018] Belzungue, M. A. (2018). High-resolution heterogeneous digital PET [18F] FDG brain phantom based on the bigbrain atlas. *Zenodo [Online]*, 5281.
- [Bennequin et al., 2021] Bennequin, E., Bouvier, V., Tami, M., Toubhans, A., and Hudelot, C. (2021). Bridging few-shot learning and adaptation: new challenges of support-query shift. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 554–569. Springer.
- [Bertocchi et al., 2019] Bertocchi, C., Chouzenoux, E., Corbineau, M.-C., Pesquet, J.-C., and Prato, M. (2019). Deep unfolding of a proximal interior point method for image restoration. *Inverse Problems*.
- [Bertsekas, 1995] Bertsekas, D. (1995). *Nonlinear Programming*. Athena Scientific Optimization and Computation Series. Athena Scientific.
- [Bertsekas, 2014] Bertsekas, D. P. (2014). *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press.
- [Besson et al., 2017] Besson, A., Perdios, D., Martinez, F., Chen, Z., Carrillo, R. E., Ardit, M., Wiaux, Y., and Thiran, J.-P. (2017). Ultrafast ultrasound imaging as an inverse problem: Matrix-free sparse image reconstruction. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 65(3):339–355.

- [Bezdek, 2013] Bezdek, J. C. (2013). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Springer Science & Business Media.
- [Bioucas-Dias and Figueiredo, 2009] Bioucas-Dias, J. M. and Figueiredo, M. A. (2009). Total variation restoration of speckled images using a split-Bregman algorithm. In *16th IEEE International Conference on Image Processing (ICIP)*, pages 3717–3720. IEEE.
- [Bishop and Nasrabadi, 2006] Bishop, C. M. and Nasrabadi, N. M. (2006). *Pattern Recognition and Machine Learning*, volume 4. Springer.
- [Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022.
- [Bohra et al., 2019] Bohra, P., Garg, D., Gurumoothy, K. S., and Rajwade, A. (2019). Variance-stabilization-based compressive inversion under Poisson or Poisson–Gaussian noise with analytical bounds. *Inverse Problems*, 35(10):105006.
- [Bolte et al., 2010] Bolte, J., Combettes, P. L., and Pesquet, J.-C. (2010). Alternating proximal algorithm for blind image recovery. In *Proceedings of the 17th IEEE International Conference on Image Processing*, pages 1673–1676, Hong Kong, Hong Kong. IEEE.
- [Bolte et al., 2006] Bolte, J., Daniilidis, A., and Lewis, A. (2006). A nonsmooth Morse–Sard theorem for subanalytic functions. *Journal of Mathematical Analysis and Applications*, 321(2):729–740.
- [Bolte et al., 2007] Bolte, J., Daniilidis, A., and Lewis, A. (2007). The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17(4):1205–1223.
- [Bolte et al., 2014] Bolte, J., Sabach, S., and Teboulle, M. (2014). Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494.
- [Bonettini et al., 2018] Bonettini, S., Prato, M., and Rebegoldi, S. (2018). A block coordinate variable metric linesearch based proximal gradient method. *Computational Optimization and Applications*, 71(1):5–52.
- [Bonettini et al., 2008] Bonettini, S., Zanella, R., and Zanni, L. (2008). A scaled gradient projection method for constrained image deblurring. *Inverse Problems*, 25(1):015002.
- [Bonnans et al., 2006] Bonnans, J.-F., Gilbert, J. C., Lemaréchal, C., and Sagastizábal, C. A. (2006). *Numerical Optimization: Theoretical and Practical Aspects*. Springer Science & Business Media.
- [Bossard et al., 2014] Bossard, L., Guillaumin, M., and Van Gool, L. (2014). Food-101—mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)*, pages 446–461, Zurich, Switzerland. Springer.
- [Boudiaf et al., 2023] Boudiaf, M., Bennequin, E., Tami, M., Toubhans, A., Piantanida, P., Hudelot, C., and Ayed, I. B. (2023). Open-set likelihood maximization for few-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

- [Boudiaf et al., 2021] Boudiaf, M., Kervadec, H., Masud, Z. I., Piantanida, P., Ben Ayed, I., and Dolz, J. (2021). Few-shot segmentation without meta-learning: A good transductive inference is all you need? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13979–13988.
- [Boudiaf et al., 2020] Boudiaf, M., Ziko, I., Rony, J., Dolz, J., Piantanida, P., and Ben Ayed, I. (2020). Information maximization for few-shot learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:2445–2457.
- [Boyd et al., 2011] Boyd, S., Parikh, N., and Chu, E. (2011). *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Now Publishers Inc.
- [Boykov et al., 2015] Boykov, Y., Isack, H., Olsson, C., and Ben Ayed, I. (2015). Volumetric bias in segmentation and reconstruction: Secrets and solutions. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1769–1777.
- [Branch et al., 1999] Branch, M. A., Coleman, T. F., and Li, Y. (1999). A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. *SIAM Journal on Scientific Computing*, 21(1):1–23.
- [Bredies et al., 2010] Bredies, K., Kunisch, K., and Pock, T. (2010). Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526.
- [Bronskill et al., 2020] Bronskill, J., Gordon, J., Requeima, J., Nowozin, S., and Turner, R. (2020). Tasknorm: Rethinking batch normalization for meta-learning. In *International Conference on Machine Learning (ICML)*, pages 1153–1164. PMLR.
- [Bruce et al., 2002] Bruce, L. M., Koger, C. H., and Li, J. (2002). Dimensionality reduction of hyperspectral data using discrete wavelet transform feature extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 40(10):2331–2338.
- [Byrd et al., 2000] Byrd, R. H., Gilbert, J.-C., and Nocedal, J. (2000). A trust region method based on interior point techniques for nonlinear programming. *Mathematical Programming*, 89(1):149–185.
- [Byrd et al., 1999] Byrd, R. H., Hribar, M. E., and Nocedal, J. (1999). An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900.
- [Byrd et al., 1995] Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208.
- [Cao et al., 2013] Cao, J., Wu, Z., Wu, J., and Xiong, H. (2013). SAIL: Summation-bAsed Incremental Learning for information-theoretic text clustering. *IEEE Transactions on Cybernetics*, 43(2):570–584.
- [Carlavan and Blanc-Féraud, 2011] Carlavan, M. and Blanc-Féraud, L. (2011). Sparse Poisson noisy image deblurring. *IEEE Transactions on Image Processing*, 21(4):1834–1846.
- [Carrillo et al., 2012] Carrillo, R. E., McEwen, J. D., and Wiaux, Y. (2012). Sparsity Averaging Reweighted Analysis (SARA): a novel algorithm for radio-interferometric imaging. *Monthly Notices of the Royal Astronomical Society*, 426(2):1223–1234.

- [Castillo, 2019] Castillo, E. (2019). Quadratic penalty method for intensity-based deformable image registration and 4DCT lung motion recovery. *Medical Physics*, 46(5):2194–2203.
- [Celis et al., 1984] Celis, M. R., Dennis, J. E., and Tapia, R. A. (1984). A trust region strategy for equality constrained optimization. Technical report, Rice University Houston TX Department of Mathematical Sciences.
- [Centonze and White, 1998] Centonze, V. E. and White, J. G. (1998). Multiphoton excitation provides optical sections from deeper within scattering specimens than confocal imaging. *Biophysical Journal*, 75(4):2015–2024.
- [Chalvidal et al., 2023] Chalvidal, M., Chouzenoux, E., Fest, J.-B., and Lefort, C. (2023). Block delayed Majorize-Minimize subspace algorithm for large scale image restoration. *Inverse Problems*, 39(4):044002.
- [Chambolle, 2004] Chambolle, A. (2004). An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1):89–97.
- [Chambolle and Dossal, 2015] Chambolle, A. and Dossal, C. (2015). On the convergence of the iterates of the "fast iterative shrinkage/thresholding algorithm". *Journal of Optimization Theory and Applications*, 166(3):968–982.
- [Chambolle and Pock, 2011] Chambolle, A. and Pock, T. (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145.
- [Chan et al., 2013] Chan, R. H., Tao, M., and Yuan, X. (2013). Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers. *SIAM Journal on Imaging Sciences*, 6(1):680–697.
- [Chan et al., 2005] Chan, T., Esedoglu, S., Park, F., Yip, A., et al. (2005). Recent developments in total variation image restoration. *Mathematical Models of Computer Vision*, 17(2):17–31.
- [Chan and Mulet, 1999] Chan, T. F. and Mulet, P. (1999). On the convergence of the lagged diffusivity fixed point method in total variation image restoration. *SIAM Journal on Numerical Analysis*, 36(2):354–367.
- [Chan and Wong, 2000] Chan, T. F. and Wong, C.-K. (2000). Convergence of the alternating minimization algorithm for blind deconvolution. *Linear Algebra and its Applications*, 316(1-3):259–285.
- [Chang, 1983] Chang, W.-C. (1983). On using principal components before separating a mixture of two multivariate normal distributions. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 32(3):267–275.
- [Charbonnier et al., 1997] Charbonnier, P., Blanc-Féraud, L., Aubert, G., and Barlaud, M. (1997). Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, 6(2):298–311.

- [Chen et al., 2022] Chen, G., Yao, W., Song, X., Li, X., Rao, Y., and Zhang, K. (2022). Prompt learning with optimal transport for vision-language models. In *International Conference on Learning Representations (ICLR)*.
- [Chen and Rockafellar, 1997] Chen, G. H. and Rockafellar, R. T. (1997). Convergence rates in forward–backward splitting. *SIAM Journal on Optimization*, 7(2):421–444.
- [Chen et al., 2001] Chen, S. S., Donoho, D. L., and Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159.
- [Chen et al., 2019] Chen, W.-Y., Liu, Y.-C., Kira, Z., Wang, Y.-C. F., and Huang, J.-B. (2019). A closer look at few-shot classification. In *International Conference on Learning Representations (ICLR)*.
- [Chen et al., 2012] Chen, X., Ng, M. K., and Zhang, C. (2012). Non-lipschitz ℓ_p -regularization and box constrained model for image restoration. *IEEE Transactions on Image Processing*, 21(12):4709–4721.
- [Cherni et al., 2016] Cherni, A., Chouzenoux, É., and Delsuc, M.-A. (2016). Proximity operators for a class of hybrid sparsity + entropy priors application to dosy NMR signal reconstruction. In *International Symposium on Signal, Image, Video and Communications (ISIVC)*, pages 120–125, Tunis, Tunisia. IEEE.
- [Cherni et al., 2020] Cherni, A., Chouzenoux, E., Duval, L., and Pesquet, J.-C. (2020). SPOQ ℓ_p -over- ℓ_q regularization for sparse signal recovery applied to mass spectrometry. *IEEE Transactions on Signal Processing*, 68:6070–6084.
- [Chouzenoux et al., 2019a] Chouzenoux, E., Corbineau, M.-C., and Pesquet, J.-C. (2019a). A proximal interior point algorithm with applications to image processing. *Journal of Mathematical Imaging and Vision*, pages 1–22.
- [Chouzenoux et al., 2023] Chouzenoux, E., Corbineau, M.-C., Pesquet, J.-C., and Scrivanti, G. (2023). A variational approach for joint image recovery and feature extraction based on spatially varying generalised Gaussian models. <https://hal.science/hal-03591742>.
- [Chouzenoux et al., 2010] Chouzenoux, E., Idier, J., and Moussaoui, S. (2010). A Majorize–Minimize strategy for subspace optimization applied to image restoration. *IEEE Transactions on Image Processing*, 20(6):1517–1528.
- [Chouzenoux et al., 2013a] Chouzenoux, É., Jezierska, A., Pesquet, J.-C., and Talbot, H. (2013a). A Majorize–Minimize subspace approach for $\ell_2 - \ell_0$ image regularization. *SIAM Journal on Imaging Sciences*, 6:563–591.
- [Chouzenoux et al., 2015] Chouzenoux, E., Jezierska, A., Pesquet, J.-C., and Talbot, H. (2015). A convex approach for image restoration with exact Poisson–Gaussian likelihood. *SIAM Journal on Imaging Sciences*, 8(4):2662–2682.
- [Chouzenoux et al., 2019b] Chouzenoux, E., Lau, T. T.-K., Lefort, C., and Pesquet, J.-C. (2019b). Optimal multivariate Gaussian fitting with applications to PSF modeling in two-photon microscopy imaging. *Journal of Mathematical Imaging and Vision*, 61(7):1037–1050.

- [Chouzenoux et al., 2022] Chouzenoux, E., Martin, S., and Pesquet, J.-C. (2022). A local MM subspace method for solving constrained variational problems in image recovery. *Journal of Mathematical Imaging and Vision*.
- [Chouzenoux and Pesquet, 2016] Chouzenoux, E. and Pesquet, J.-C. (2016). Convergence rate analysis of the Majorize–Minimize subspace algorithm. *IEEE Signal Processing Letters*, 23(9):1284–1288.
- [Chouzenoux et al., 2014] Chouzenoux, E., Pesquet, J.-C., and Repetti, A. (2014). Variable metric forward–backward algorithm for minimizing the sum of a differentiable function and a convex function. *Journal of Optimization Theory and Applications*, 162(1):107–132.
- [Chouzenoux et al., 2016] Chouzenoux, E., Pesquet, J.-C., and Repetti, A. (2016). A block coordinate variable metric forward–backward algorithm. *Journal of Global Optimization*, 66(3):457–485.
- [Chouzenoux et al., 2013b] Chouzenoux, E., Zolyniak, F., Gouillart, E., and Talbot, H. (2013b). A Majorize–Minimize memory gradient algorithm applied to X-ray tomography. In *IEEE International Conference on Image Processing*. IEEE.
- [Cidade et al., 2000] Cidade, G. A., Anteneodo, C., Roberty, N. C., and Silva Neto, A. J. (2000). A generalized approach for atomic force microscopy image restoration with Bregman distances as Tikhonov regularization terms. *Inverse Problems in Engineering*, 8(5):457–472.
- [Ciga et al., 2022] Ciga, O., Xu, T., and Martel, A. L. (2022). Self supervised contrastive learning for digital histopathology. *Machine Learning with Applications*, 7:100198.
- [Cimpoi et al., 2014] Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. (2014). Describing textures in the wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Cole et al., 2011] Cole, R. W., Jinadasa, T., and Brown, C. M. (2011). Measuring and interpreting point spread functions to determine confocal microscope resolution and ensure quality control. *Nature Protocols*, 6(12):1929–1941.
- [Coleman and Li, 1996] Coleman, T. F. and Li, Y. (1996). An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6(2):418–445.
- [Combettes et al., 2011] Combettes, P. L., Dūng, Đ., and Vũ, B. C. (2011). Proximity for sums of composite functions. *Journal of Mathematical Analysis and applications*, 380(2):680–688.
- [Combettes and Pesquet, 2004] Combettes, P. L. and Pesquet, J.-C. (2004). Image restoration subject to a total variation constraint. *IEEE Transactions on Image Processing*, 13(9):1213–1222.
- [Combettes and Pesquet, 2011] Combettes, P. L. and Pesquet, J.-C. (2011). Proximal splitting methods in signal processing. In *Fixed-point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer.
- [Combettes and Pesquet, 2020] Combettes, P. L. and Pesquet, J.-C. (2020). Deep neural network structures solving variational inequalities. *Set-Valued and Variational Analysis*, 28(3):491–518.
- [Combettes and Pesquet, 2021] Combettes, P. L. and Pesquet, J.-C. (2021). Fixed point strategies in data science. *IEEE Transactions on Signal Processing*, 69:3878–3905.

- [Combettes and Trussell, 1989] Combettes, P. L. and Trussell, H. J. (1989). Methods for digital restoration of signals degraded by a stochastic impulse response. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3):393–401.
- [Combettes and Vũ, 2014] Combettes, P. L. and Vũ, B. C. (2014). Variable metric forward–backward splitting with applications to monotone inclusions in duality. *Optimization*, 63(9):1289–1318.
- [Combettes and Wajs, 2005] Combettes, P. L. and Wajs, V. R. (2005). Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation. A SIAM Interdisciplinary Journal*, 4(4):1168–1200.
- [Condat, 2012] Condat, L. (2012). A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2):460–479.
- [Condat, 2014] Condat, L. (2014). Semi-local total variation for regularization of inverse problems. In *22nd European Signal Processing Conference (EUSIPCO)*, pages 1806–1810. IEEE.
- [Condat, 2015] Condat, L. (2015). Fast projection onto the simplex and the ℓ_1 -ball. *Mathematical Programming*, 158(1-2):575–585.
- [Condat, 2017] Condat, L. (2017). Discrete total variation: New definition and minimization. *SIAM Journal on Imaging Sciences*, 10(3):1258–1290.
- [Conn et al., 1991] Conn, A. R., Gould, N. I. M., and Toint, P. (1991). A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Numerical Analysis*, 28(2):545–572.
- [Conn et al., 2000] Conn, A. R., Gould, N. I. M., and Toint, P. L. (2000). *Trust Region Methods*. Society for Industrial and Applied Mathematics.
- [Cooper et al., 2023] Cooper, M., Ji, Z., and Krishnan, R. G. (2023). Machine learning in computational histopathology: Challenges and opportunities. *Genes, Chromosomes and Cancer*.
- [Corless et al., 1996] Corless, R. M., Gonnet, G. H., Hare, D. E. G., Jeffrey, D. J., and Knuth, D. E. (1996). On the Lambert W function. *Advances in Computational Mathematics*, 5:329–359.
- [Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297.
- [Cox, 1958] Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 20(2):215–232.
- [Cragg and Levy, 1969] Cragg, E. and Levy, A. (1969). Study on a supermemory gradient method for the minimization of functions. *Journal of Optimization Theory and Applications*, 4(3):191–205.
- [Crivaro et al., 2011] Crivaro, M., Enjieu-Kadji, H., Hatanaka, R., Nakauchi, S., Bosch, J., Judin, J., Riéra, J., and Kawashima, R. (2011). Multiphoton fluorescent images with a spatially varying background signal: a ML deconvolution method. *Journal of Microscopy*, 242(3):311–324.

- [Crouse, 2016] Crouse, D. F. (2016). On implementing 2D rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696.
- [Cui and Guo, 2021] Cui, W. and Guo, Y. (2021). Parameterless transductive feature re-representation for few-shot learning. In *International Conference on Machine Learning (ICML)*, pages 2212–2221. PMLR.
- [Curtis et al., 2017] Curtis, F. E., Gould, N. I. M., Robinson, D. P., and Toint, P. L. (2017). An interior-point trust-funnel algorithm for nonlinear optimization. *Mathematical Programming*, 161(1-2):73–134.
- [Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. Ieee.
- [Danielyan et al., 2014] Danielyan, A., Wu, Y.-W., Shih, P.-Y., Dembitskaya, Y., and Semyanov, A. (2014). Denoising of two-photon fluorescence images with block-matching 3D filtering. *Methods*, 68(2):308–316.
- [Dao et al., 2015] Dao, L., Glancy, B., Lucotte, B., Chang, L.-C., Balaban, R. S., and Hsu, L. (2015). A model-based approach for microvasculature structure distortion correction in two-photon fluorescence microscopy images. *Journal of Microscopy*, 260(2):180–193.
- [Daubechies et al., 2004] Daubechies, I., Defrise, M., and De Mol, C. (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11):1413–1457.
- [De la Torre and Kanade, 2006] De la Torre, F. and Kanade, T. (2006). Discriminative cluster analysis. In *International Conference on Machine Learning (ICML)*, pages 241–248, Pittsburgh, USA.
- [De Monvel et al., 2003] De Monvel, J. B., Scarfone, E., Le Calvez, S., and Ulfendahl, M. (2003). Image-adaptive deconvolution for three-dimensional deep biological imaging. *Biophysical Journal*, 85(6):3991–4001.
- [De Moraes Marim et al., 2008] De Moraes Marim, M., Zhang, B., Olivo-Marin, J.-C., and Zimmer, C. (2008). Improving single particle localization with an empirically calibrated Gaussian kernel. In *5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1003–1006. IEEE.
- [Debarnot and Weiss, 2021] Debarnot, V. and Weiss, P. (2021). Deepblur: Blind identification of space variant PSF. In *IEEE 18th International Symposium on Biomedical Imaging (ISBI)*.
- [Delong et al., 2012] Delong, A., Osokin, A., Isack, H. N., and Boykov, Y. (2012). Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, 96:1–27.
- [Depriester, 2018] Depriester, D. (2018). Computing Euler angles with Bunge convention from rotation matrix. 10.13140/RG.2.2.34498.48321/5.
- [Descloux et al., 2016] Descloux, A., Amitonova, L. V., and Pinkse, P. W. H. (2016). Aberrations of the point spread function of a multimode fiber due to partial mode excitation. *Optics Express*, 24(16):18501–18512.

- [Dey et al., 2006] Dey, N., Blanc-Feraud, L., Zimmer, C., Roux, P., Kam, Z., Olivo-Marin, J.-C., and Zerubia, J. (2006). Richardson-Lucy algorithm with total variation regularization for 3D confocal microscope deconvolution. *Microscopy Research and Technique*, 69(4):260–266.
- [Dhillon et al., 2019] Dhillon, G. S., Chaudhari, P., Ravichandran, A., and Soatto, S. (2019). A baseline for few-shot image classification. *arXiv preprint arXiv:1909.02729*.
- [Di Pillo and Grippo, 1989] Di Pillo, G. and Grippo, L. (1989). Exact penalty functions in constrained optimization. *SIAM Journal on control and optimization*, 27(6):1333–1360.
- [Diaspro, 2001] Diaspro, A. (2001). *Confocal and Two-Photon Microscopy: Foundations, Applications and Advances*. Wiley-Liss Inc., U.S.
- [Diaspro et al., 2006] Diaspro, A., Bianchini, P., Vicidomini, G., Faretta, M., Ramoino, P., and Usai, C. (2006). Multi-photon excitation microscopy. *Biomedical Engineering Online*, 5:1–14.
- [Difato et al., 2004] Difato, F., Mazzone, F., Scaglione, S., Fato, M., Beltrame, F., Kubínová, L., Janáček, J., Ramoino, P., Vicidomini, G., and Diaspro, A. (2004). Improvement in volume estimation from confocal sections after image deconvolution. *Microscopy Research and Technique*, 64(2):151–155.
- [Dimitriou et al., 2019] Dimitriou, N., Arandjelovic, O., and Caie, P. D. (2019). Deep learning for Whole Slide Image analysis: An overview. *Clinical Orthopaedics and Related Research*, abs/1910.11097.
- [Doi et al., 2018] Doi, A., Oketani, R., Nawa, Y., and Fujita, K. (2018). High-resolution imaging in two-photon excitation microscopy using in situ estimations of the point spread function. *Biomedical Optics Express*, 9(1):202–213.
- [Dong et al., 2003] Dong, C.-Y., Koenig, K., and So, P. (2003). Characterizing point spread functions of two-photon fluorescence microscopy in turbid medium. *Journal of Biomedical Optics*, 8(3):450–459.
- [Donoho, 1995] Donoho, D. L. (1995). De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627.
- [Du et al., 2012] Du, L., Li, X., and Shen, Y.-D. (2012). Robust nonnegative matrix factorization via half-quadratic minimization. In *IEEE 12th International Conference on Data Mining*, pages 201–210. IEEE.
- [Dupe et al., 2009] Dupe, F.-X., Fadili, J. M., and Starck, J.-L. (2009). A proximal iteration for deconvolving Poisson noisy images using sparse representations. *IEEE Transactions on Image Processing*, 18(2):310–321.
- [Durand et al., 2009] Durand, S., Fadili, J., and Nikolova, M. (2009). Multiplicative noise removal using ℓ_1 fidelity on frame coefficients. *Journal of Mathematical Imaging and Vision*, 36(3):201–226.
- [Dusch et al., 2007] Dusch, E., Dorval, T., Vincent, N., Wachsmuth, M., and Genovesio, A. (2007). Three-dimensional point spread function model for line-scanning confocal microscope with high-aperture objective. *Journal of Microscopy*, 228(2):132–138.
- [Dussault, 1995] Dussault, J.-P. (1995). Numerical stability and efficiency of penalty algorithms. *SIAM Journal on Numerical Analysis*, 32(1):296–317.

- [Elad et al., 2007] Elad, M., Matalon, B., and Zibulevsky, M. (2007). Coordinate and subspace optimization methods for linear least squares with non-quadratic regularization. *Applied and Computational Harmonic Analysis*, 23(3):346–367.
- [Erdogan and Fessler, 2002] Erdogan, H. and Fessler, J. A. (2002). Monotonic algorithms for transmission tomography. In *5th IEEE EMBS International Summer School on Biomedical Imaging*, pages 14–pp. IEEE.
- [Everitt et al., 2011] Everitt, B. S., Landau, S., Leese, M., and Stahl, D. (2011). *Cluster Analysis*. Wiley Series in Probability and Statistics. Wiley.
- [Fei-Fei et al., 2006] Fei-Fei, L., Fergus, R., and Perona, P. (2006). One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611.
- [Feng et al., 2016] Feng, Q., Zhou, Y., and Lan, R. (2016). Pairwise linear regression classification for image set retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4865–4872, Hawai, USA.
- [Fiacco and McCormick, 1990] Fiacco, A. V. and McCormick, G. P. (1990). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. SIAM.
- [Figueiredo and Nowak, 2003] Figueiredo, M. A. and Nowak, R. D. (2003). An EM algorithm for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 12(8):906–916.
- [Finn et al., 2017] Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, pages 1126–1135. PMLR.
- [Fletcher, 1985] Fletcher, R. (1985). A penalty method for nonlinear constraints. In *Numerical Optimization 1984*, pages 26–40. SIAM Publications.
- [Fletcher and Reeves, 1964] Fletcher, R. and Reeves, C. M. (1964). Function minimization by conjugate gradients. *The Computer Journal*, 7(2):149–154.
- [Florescu et al., 2014] Florescu, A., Chouzenoux, E., Pesquet, J.-C., Ciuci, P., and Ciocchina, S. (2014). A Majorize-Minimize memory gradient method for complex-valued inverse problems. *Signal Processing*, 103:285–295.
- [Foare et al., 2019] Foare, M., Pustelnik, N., and Condat, L. (2019). Semi-linearized proximal alternating minimization for a discrete Mumford–Shah model. *IEEE Transactions on Image Processing*, 29:2176–2189.
- [Foi, 2009] Foi, A. (2009). Clipped noisy images: Heteroskedastic modeling and practical denoising. *Signal Processing*, 89(12):2609–2629.
- [Foi et al., 2008] Foi, A., Trimeche, M., Katkovnik, V., and Egiazarian, K. (2008). Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754.

- [Fortun et al., 2018] Fortun, D., Guichard, P., Hamel, V., Sorzano, C. O. S., Banterle, N., Gönczy, P., and Unser, M. (2018). Reconstruction from multiple particles for 3D isotropic resolution in fluorescence microscopy. *IEEE Transactions on Medical Imaging*, 37(5):1235–1246.
- [Friedman et al., 2008] Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical Lasso. *Biostatistics*, 9(3):432–441.
- [Fu et al., 2006] Fu, H., Ng, M. K., Nikolova, M., and Barlow, J. L. (2006). Efficient minimization methods of mixed $\ell_2 - \ell_1$ and $\ell_1 - \ell_1$ norms for image restoration. *SIAM Journal on Scientific Computing*, 27(6):1881–1902.
- [Gao et al., 2023] Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., and Qiao, Y. (2023). CLIP-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, pages 1–15.
- [Geman and Reynolds, 1992] Geman, D. and Reynolds, G. (1992). Constrained restoration and the recovery of discontinuities. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 14(03):367–383.
- [Gharbi et al., 2022] Gharbi, M., Villa, S., Chouzenoux, E., and Pesquet, J.-C. (2022). Unrolled primal-dual deep network for sparse signal restoration. hal-03988686.
- [Ghojogh and Crowley, 2019] Ghojogh, B. and Crowley, M. (2019). Linear and quadratic discriminant analysis: Tutorial. *arXiv preprint arXiv:1906.02590*.
- [Göbel et al., 2007] Göbel, W., Kampa, B. M., and Helmchen, F. (2007). Imaging cellular network dynamics in three dimensions using fast 3D laser scanning. *Nature Methods*, 4(1):73–79.
- [Golub et al., 1999] Golub, G. H., Hansen, P. C., and O’Leary, D. P. (1999). Tikhonov regularization and total least squares. *SIAM journal on matrix analysis and applications*, 21(1):185–194.
- [Gould, 1989] Gould, N. I. M. (1989). On the convergence of a sequential penalty function method for constrained minimization. *SIAM Journal on Numerical Analysis*, 26(1):107–128.
- [Gould and Toint, 2010] Gould, N. I. M. and Toint, P. L. (2010). Nonlinear programming without a penalty function or a filter. *Mathematical Programming*, 122(1):155–196.
- [Grapiglia et al., 2013] Grapiglia, G. N., Yuan, J., and Yuan, Y.-X. (2013). A subspace version of the Powell–Yuan trust-region algorithm for equality constrained optimization. *Journal of the Operations Research Society of China*, 1(4):425–451.
- [Guo, 2011] Guo, H. (2011). A simple algorithm for fitting a Gaussian function [DSP tips and tricks]. *IEEE Signal Processing Magazine*, 28(5):134–137.
- [Guo et al., 2020] Guo, Y., Codella, N. C., Karlinsky, L., Codella, J. V., Smith, J. R., Saenko, K., Rosing, T., and Feris, R. (2020). A broader study of cross-domain few-shot learning. In *European Conference on Computer Vision (ECCV)*, pages 124–141, Glasgow, UK. Springer.
- [Hamsici and Martinez, 2007] Hamsici, O. C. and Martinez, A. M. (2007). Spherical-homoscedastic distributions: The equivalency of spherical and normal distributions in classification. *Journal of Machine Learning Research*, 8(7).

- [Hara and Maehara, 2017] Hara, S. and Maehara, T. (2017). Enumerate Lasso solutions for feature selection. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, volume 31, San Francisco, USA.
- [Harizanov et al., 2013] Harizanov, S., Pesquet, J.-C., and Steidl, G. (2013). Epigraphical projection for solving least squares Anscombe transformed constrained optimization problems. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 125–136, Berlin, Heidelberg. Springer.
- [Hasnat et al., 2017] Hasnat, M. A., Bohné, J., Milgram, J., Gentric, S., and Chen, L. (2017). Von Mises-Fisher mixture model-based deep learning: Application to face verification. *arXiv preprint arXiv:1706.04264*.
- [He and Wang, 1990] He, D.-C. and Wang, L. (1990). Texture unit, texture spectrum, and texture analysis. *IEEE Transactions on Geoscience and Remote Sensing*, 28(4):509–512.
- [He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.
- [He et al., 2013] He, R., Zheng, W.-S., Tan, T., and Sun, Z. (2013). Half-quadratic-based iterative minimization for robust sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(2):261–275.
- [Helber et al., 2019] Helber, P., Bischke, B., Dengel, A., and Borth, D. (2019). EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226.
- [Helstrom, 1967] Helstrom, C. W. (1967). Image restoration by the method of least squares. *JOSA*, 57(3):297–303.
- [Högbom, 1974] Högbom, J. A. (1974). Aperture synthesis with a non-regular distribution of interferometer baselines. *Astronomy and Astrophysics Supplement*, 15:417.
- [hong et al., 2021] hong, Z., Friedman, D., and Chen, D. (2021). Factual probing is [mask]: Learning vs. learning to recall. In *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- [Houle et al., 2010] Houle, M. E., Kriegel, H.-P., Kröger, P., Schubert, E., and Zimek, A. (2010). Can shared-neighbor distances defeat the curse of dimensionality? In *Proceedings of the 22nd International Conference on Scientific and Statistical Database Management*, pages 482–500, Heidelberg, Germany. Springer.
- [Hu et al., 2022] Hu, S. X., Li, D., Stühmer, J., Kim, M., and Hospedales, T. M. (2022). Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9068–9077.
- [Hu et al., 2021] Hu, Y., Gripon, V., and Pateux, S. (2021). Leveraging the feature distribution in transfer-based few-shot learning. In *International Conference on Artificial Neural Networks*, pages 487–499. Springer.

- [Huang, 2005] Huang, J. (2005). Maximum likelihood estimation of Dirichlet distribution parameters. Technical report, Carnegie Mellon University.
- [Huang et al., 2021] Huang, Y., Chouzenoux, E., and Elvira, V. (2021). Probabilistic modeling and inference for sequential space-varying blur identification. *IEEE Transactions on Computational Imaging*, 7:531–546.
- [Huang et al., 2022] Huang, Y., Chouzenoux, E., and Pesquet, J.-C. (2022). Unrolled variational Bayesian algorithm for image blind deconvolution. *IEEE Transactions on Image Processing*, 32:430–445.
- [Idier, 2001] Idier, J. (2001). Convex half-quadratic criteria and interacting auxiliary variables for image restoration. *IEEE transactions on image processing*, 10(7):1001–1009.
- [Ivanov et al., 2013] Ivanov, V. K., Vasin, V. V., and Tanana, V. P. (2013). *Theory of Linear Ill-Posed Problems and its Applications*. De Gruyter.
- [Jia et al., 2021] Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning (ICML)*, pages 4904–4916. PMLR.
- [Joachims, 1999] Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *International Conference on Machine Learning (ICML)*, volume 99, pages 200–209.
- [Johnson and Khoshgoftaar, 2019] Johnson, J. M. and Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1):1–54.
- [Jonard et al., 2022] Jonard, M., Leventoux, Y., Colas, M., Février, S., Mansuryan, T., Cornette, J., Tonello, A., Krupa, K., Duclère, J.-R., and Lefort, C. (2022). 3D multiphoton characterization of χ^2 nonlinearity induced in a multimode fiber through optical poling. In *2022 Conference on Lasers and Electro-Optics (CLEO)*, pages 1–2. IEEE.
- [Kearns et al., 1998] Kearns, M., Mansour, Y., and Ng, A. Y. (1998). An information-theoretic analysis of hard and soft assignment methods for clustering. *Learning in Graphical Models*, pages 495–520.
- [Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, volume abs/1412.6980.
- [Kirshner et al., 2013] Kirshner, H., Aguet, F., Sage, D., and Unser, M. (2013). 3-D PSF fitting for fluorescence microscopy: implementation and localization application. *Journal of Microscopy*, 249(1):13–25.
- [Komodakis and Pesquet, 2015] Komodakis, N. and Pesquet, J.-C. (2015). Playing with duality: an overview of recent primal-dual approaches for solving large-scale optimization problems. *IEEE Signal Processing Magazine*, 32(6):31–54.
- [Konnov, 2019] Konnov, I. V. (2019). An approximate penalty method with descent for convex optimization problems. *Russian Mathematics*, 63(7):41–55.

- [Krause et al., 2013] Krause, J., Stark, M., Deng, J., and Fei-Fei, L. (2013). 3D object representations for fine-grained categorization. In *IEEE/CVF International Conference on Computer Vision Workshops*, pages 554–561.
- [Krist et al., 2011] Krist, J. E., Hook, R. N., and Stoehr, F. (2011). 20 years of Hubble space telescope optical modeling using Tiny Tim. In *Optical Modeling and Performance Predictions V*, volume 8127, pages 166–181. SPIE.
- [Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 25.
- [Kurdyka, 1998] Kurdyka, K. (1998). On gradients of functions definable in o-minimal structures. In *Annales de l’Institut Fourier*, volume 48, pages 769–783.
- [Lapin et al., 2017] Lapin, M., Hein, M., and Schiele, B. (2017). Analysis and optimization of loss functions for multiclass, top-k, and multilabel classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(7):1533–1554.
- [Larson, 2011] Larson, A. M. (2011). Multiphoton microscopy. *Nature Photonics*, 5(1):1–1.
- [Lazarou et al., 2021] Lazarou, M., Stathaki, T., and Avrithis, Y. (2021). Iterative label cleaning for transductive and semi-supervised few-shot learning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8751–8760.
- [Lee et al., 2019] Lee, J. H., Jung, Y. M., Yuan, Y.-X., and Yun, S. (2019). A subspace SQP method for equality constrained optimization. *Computational Optimization and Applications. An International Journal*, 74(1):177–194.
- [Lefort et al., 2021] Lefort, C., Chalvidal, M., Parenté, A., Blanquet, V., Massias, H., Magnol, L., and Chouzenoux, E. (2021). FAMOUS: a fast instrumental and computational pipeline for multiphoton microscopy applied to 3D imaging of muscle ultrastructure. *Journal of Physics D: Applied Physics*, 54(27):274005.
- [Lehtinen et al., 2018] Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T. (2018). Noise2Noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*.
- [Li et al., 2015] Li, J., Shen, Z., Yin, R., and Zhang, X. (2015). A reweighted ℓ^2 method for image restoration with Poisson and mixed Poisson-Gaussian noise. *Inverse Problems and Imaging*, 9(3):875.
- [Li et al., 2018] Li, J., Xue, F., and Blu, T. (2018). Accurate 3D PSF estimation from a wide-field microscopy image. In *IEEE 15th International Symposium on Biomedical Imaging (ISBI)*, pages 501–504. IEEE.
- [Li et al., 2022] Li, Y., Liang, F., Zhao, L., Cui, Y., Ouyang, W., Shao, J., Yu, F., and Yan, J. (2022). Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. In *International Conference on Learning Representations (ICLR)*.

- [Lichtenstein et al., 2020] Lichtenstein, M., Sattigeri, P., Feris, R., Giryes, R., and Karlinsky, L. (2020). TAFSSL: Task-adaptive feature sub-space learning for few-shot classification. In *European Conference on Computer Vision (ECCV)*, pages 522–539. Springer.
- [Liu and Nocedal, 1989] Liu, D. C. and Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528.
- [Liu et al., 2020] Liu, J., Song, L., and Qin, Y. (2020). Prototype rectification for few-shot learning. In *European Conference on Computer Vision (ECCV)*, pages 741–756. Springer.
- [Liu et al., 2019] Liu, Y., Lee, J., Park, M., Kim, S., Yang, E., Hwang, S. J., and Yang, Y. (2019). Learning to propagate labels: Transductive propagation network for few-shot learning. In *International Conference on Learning Representations (ICLR)*.
- [Lloyd, 1982] Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137.
- [Lojasiewicz, 1963] Lojasiewicz, S. (1963). Une propriété topologique des sous-ensembles analytiques réels. *Les équations aux dérivées partielles*, 117:87–89.
- [Loris et al., 2009] Loris, I., Bertero, M., De Mol, C., Zanella, R., and Zanni, L. (2009). Accelerating gradient projection methods for ℓ_1 -constrained signal recovery by steplength selection rules. *Applied and Computational Harmonic Analysis*, 27(2):247–254.
- [Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, volume 2, pages 1150–1157. Ieee.
- [Luenberger, 1997] Luenberger, D. G. (1997). *Optimization by Vector Space Methods*. John Wiley & Sons.
- [Luo and Tseng, 1993] Luo, Z.-Q. and Tseng, P. (1993). Error bounds and convergence analysis of feasible descent methods: a general approach. *Annals of Operations Research*, 46(1):157–178.
- [MacKay, 2003] MacKay, D. J. C. (2003). *Information Theory, Inference and Learning Algorithms*. Cambridge University Press.
- [Maggu et al., 2018] Maggu, J., Chouzenoux, E., Chierchia, G., and Majumdar, A. (2018). Convolutional transform learning. In *Proceedings of the 25th International Conference on Neural Information Processing*, pages 162–174, Siem Reap, Cambodia. Springer.
- [Mairal, 2015] Mairal, J. (2015). Incremental Majorization-Minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, 25(2):829–855.
- [Mairal et al., 2008] Mairal, J., Leordeanu, M., Bach, F., Hebert, M., and Ponce, J. (2008). Discriminative sparse image models for class-specific edge detection and image interpretation. In *European Conference on Computer Vision (ECCV)*, pages 43–56, Marseille, France. Springer.
- [Maji et al., 2013] Maji, S., Kannala, J., Rahtu, E., Blaschko, M., and Vedaldi, A. (2013). Fine-grained visual classification of aircraft. Technical report, Oxford University.

- [Majumdar and Ward, 2009] Majumdar, A. and Ward, R. K. (2009). Fast group sparse classification. *Canadian Journal of Electrical and Computer Engineering*, 34(4):136–144.
- [Malgouyres, 2002] Malgouyres, F. (2002). Minimizing the total variation under a general convex constraint for image restoration. *IEEE Transactions on Image Processing*, 11(12):1450–1456.
- [Mallat, 1999] Mallat, S. (1999). *A Wavelet Tour of Signal Processing*. Elsevier.
- [Marquardt, 1963] Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441.
- [Martin et al., 2022] Martin, S., Boudiaf, M., Chouzenoux, E., Pesquet, J.-C., and Ayed, I. (2022). Towards practical few-shot query sets: Transductive minimum description length inference. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:34677–34688.
- [Mensink et al., 2013] Mensink, T., Verbeek, J., Perronnin, F., and Csurka, G. (2013). Distance-based image classification: generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637.
- [Miele and Cantrell, 1969] Miele, A. and Cantrell, J. W. (1969). Study on a memory gradient method for the minimization of functions. *Journal of Optimization Theory and Applications*, 3(6):459–470.
- [Minka, 2000] Minka, T. (2000). Estimating a Dirichlet distribution. Technical report, MIT.
- [Monga et al., 2021] Monga, V., Li, Y., and Eldar, Y. C. (2021). Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38(2):18–44.
- [Murtagh and Saunders, 1978] Murtagh, B. A. and Saunders, M. A. (1978). Large-scale linearly constrained optimization. *Mathematical Programming*, 14(1):41–72.
- [Musse et al., 2001] Musse, O., Heitz, F., and Armspach, J.-P. (2001). Topology preserving deformable image matching using constrained hierarchical parametric models. *IEEE Transactions on Image Processing*, 10(7):1081–1093.
- [Narayanan, 1991] Narayanan, A. (1991). Algorithm AS 266: maximum likelihood estimation of the parameters of the dirichlet distribution. *Applied Statistics*, 40(2):365–374.
- [Narkiss and Zibulevsky, 2005] Narkiss, G. and Zibulevsky, M. (2005). Sequential subspace optimization method for large-scale unconstrained optimization. Technical report, Israel Institute of Technology.
- [Naseem et al., 2010] Naseem, I., Togneri, R., and Bennamoun, M. (2010). Linear regression for face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(11):2106–2112.
- [Nesterov and Polyak, 2006] Nesterov, Y. and Polyak, B. T. (2006). Cubic regularization of Newton method and its global performance. *Mathematical Programming*, 108(1):177–205.
- [Nie et al., 2010] Nie, F., Huang, H., Cai, X., and Ding, C. (2010). Efficient and robust feature selection via joint $\ell_{2,1}$ -norms minimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 23. Citeseer.

- [Nikolova, 2004] Nikolova, M. (2004). Weakly constrained minimization: application to the estimation of images and signals involving constant regions. *Journal of Mathematical Imaging and Vision*, 21(2):155–175.
- [Nikolova, 2013] Nikolova, M. (2013). Description of the minimizers of least squares regularized with ℓ_0 -norm. uniqueness of the global minimizer. *SIAM Journal on Imaging Sciences*, 6(2):904–937.
- [Nikolova and Ng, 2005] Nikolova, M. and Ng, M. K. (2005). Analysis of half-quadratic minimization methods for signal and image recovery. *SIAM Journal on Scientific Computing*, 27(3):937–966.
- [Nilsback and Zisserman, 2008] Nilsback, M.-E. and Zisserman, A. (2008). Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE.
- [Niu et al., 2022] Niu, S.-Y., Guo, L.-Z., Li, Y., Zhang, Z., Wang, T.-D., Liu, K.-C., Li, Y.-J., Tsao, Y., and Liu, T.-M. (2022). Boundary-preserved deep denoising of stochastic resonance enhanced multiphoton images. *IEEE Journal of Translational Engineering in Health and Medicine*, 10:1–12.
- [Nocedal and Wright, 2006] Nocedal, J. and Wright, S. (2006). *Numerical Optimization*. Springer Science & Business Media.
- [Pal and Heumann, 2022] Pal, S. and Heumann, C. (2022). Clustering compositional data using Dirichlet mixture model. *PLOS One*, 17(5):e0268438.
- [Parkhi et al., 2012] Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. V. (2012). Cats and dogs. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Pearson, 1901] Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572.
- [Phan and Gillis, 2023] Phan, D. N. and Gillis, N. (2023). An inertial block Majorization Minimization framework for nonsmooth nonconvex optimization. *Journal of Machine Learning Research*, 24:1–41.
- [Pizurica et al., 2006] Pizurica, A., Wink, A. M., Vansteenkiste, E., Philips, W., and Roerdink, B. J. (2006). A review of wavelet denoising in MRI and ultrasound brain imaging. *Current Medical Imaging*, 2(2):247–260.
- [Powell, 1983] Powell, M. J. D. (1983). Variable metric methods for constrained optimization. In *Mathematical Programming the State of the Art*, pages 288–311. Springer.
- [Prato et al., 2013] Prato, M., La Camera, A., Bonettini, S., and Bertero, M. (2013). A convergent blind deconvolution method for post-adaptive-optics astronomical imaging. *Inverse Problems*, 29(6):065017.
- [Pustelnik, 2010] Pustelnik, N. (2010). *Méthodes proximales pour la résolution de problèmes inverses: application à la tomographie par émission de positrons*. PhD thesis, Université Paris-Est.
- [Pustelnik et al., 2016] Pustelnik, N., Benazza-Benayia, A., Zheng, Y., and Pesquet, J.-C. (2016). Wavelet-based image deconvolution and reconstruction. In *Wiley Encyclopedia of Electrical and Electronics Engineering*, pages 1–34. American Cancer Society.

- [Qi et al., 2021] Qi, G., Yu, H., Lu, Z., and Li, S. (2021). Transductive few-shot classification on the oblique manifold. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8412–8422.
- [Radford et al., 2021] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pages 8748–8763. PMLR.
- [Ravi and Larochelle, 2016] Ravi, S. and Larochelle, H. (2016). Optimization as a model for few-shot learning. In *International Conference on Learning Representations (ICLR)*.
- [Reinhard et al., 2001] Reinhard, E., Adhikhmin, M., Gooch, B., and Shirley, P. (2001). Color transfer between images. *IEEE Computer Graphics and Applications*, 21(5):34–41.
- [Ren et al., 2018] Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J. B., Larochelle, H., and Zemel, R. S. (2018). Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*.
- [Repetti et al., 2022] Repetti, A., Terris, M., Wiaux, Y., and Pesquet, J.-C. (2022). Dual forward-backward unfolded network for flexible plug-and-play. In *30th European Signal Processing Conference (EUSIPCO)*, pages 957–961. IEEE.
- [Rockafellar and Wets, 2009] Rockafellar, R. T. and Wets, R. J.-B. (2009). *Variational Analysis*, volume 317. Springer Science & Business Media.
- [Rossi and Barbaro, 2022] Rossi, F. and Barbaro, F. (2022). Mixture of von Mises-Fisher distribution with sparse prototypes. *Neurocomputing*, 501:41–74.
- [Rudin et al., 1992] Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268.
- [Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252.
- [Saillard et al., 2020] Saillard, C., Schmauch, B., Laifa, O., Moarii, M., Toldo, S., Zaslavskiy, M., Pronier, E., Laurent, A., Amaddeo, G., Regnault, H., et al. (2020). Predicting survival after hepatocellular carcinoma resection using deep learning on histological slides. *Hepatology*, 72(6):2000–2013.
- [Samuylov et al., 2019] Samuylov, D. K., Purwar, P., Székely, G., and Paul, G. (2019). Modeling Point Spread Function in fluorescence microscopy with a sparse Gaussian mixture: Tradeoff between accuracy and efficiency. *IEEE Transactions on Image Processing*, 28(8):3688–3702.
- [Sarder and Nehorai, 2006] Sarder, P. and Nehorai, A. (2006). Deconvolution methods for 3D fluorescence microscopy images. *IEEE Signal Processing Magazine*, 23(3):32–45.
- [Scott et al., 2021] Scott, T. R., Gallagher, A. C., and Mozer, M. C. (2021). Von Mises-Fisher loss: An exploration of embedding geometries for supervised learning. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10612–10622.

- [Setzer et al., 2010] Setzer, S., Steidl, G., and Teuber, T. (2010). Deblurring Poissonian images by split Bregman techniques. *Journal of Visual Communication and Image Representation*, 21(3):193–199.
- [Sghaier et al., 2022] Sghaier, M., Chouzenoux, E., Pesquet, J.-C., and Muller, S. (2022). A novel task-based reconstruction approach for digital breast tomosynthesis. *Medical Image Analysis*, 77:102341.
- [Shanno and Marsten, 1982] Shanno, D. F. and Marsten, R. E. (1982). Conjugate gradient methods for linearly constrained nonlinear programming. In *Mathematical Programming Studies*, pages 149–161. Springer Berlin Heidelberg.
- [Shao et al., 2018] Shao, Y., Sang, N., Gao, C., and Ma, L. (2018). Spatial and class structure regularized sparse representation graph for semi-supervised hyperspectral image classification. *Pattern Recognition*, 81:81–94.
- [Shen et al., 2021] Shen, X., Xiao, Y., Hu, S. X., Sbai, O., and Aubry, M. (2021). Re-ranking for image retrieval and transductive few-shot classification. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:25932–25943.
- [Shibata et al., 2016] Shibata, T., Tanaka, M., and Okutomi, M. (2016). Gradient-domain image reconstruction framework with intensity-range and base-structure constraints. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2745–2753.
- [Shin et al., 2016] Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., and Summers, R. M. (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging*, 35(5):1285–1298.
- [Shin et al., 2020] Shin, T., Razeghi, Y., L. R. L. I., Wallace, E., and Singh, S. (2020). Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Empirical Methods in Natural Language Processing (EMNLP)*.
- [Shultz et al., 1985] Shultz, G. A., Schnabel, R. B., and Byrd, R. H. (1985). A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties. *SIAM Journal on Numerical Analysis*, 22(1):47–67.
- [Snell et al., 2017] Snell, J., Swersky, K., and Zemel, R. (2017). Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.
- [Soomro et al., 2012] Soomro, K., Zamir, A. R., and Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. Technical report, Center for Research in Computer Vision, University of Central Florida. arXiv:1212.0402.
- [Soubies et al., 2015] Soubies, E., Blanc-Féraud, L., and Aubert, G. (2015). A continuous exact ℓ_0 penalty (cel0) for least squares regularized problem. *SIAM Journal on Imaging Sciences*, 8(3):1607–1639.
- [Soubies et al., 2017] Soubies, E., Blanc-Féraud, L., and Aubert, G. (2017). A unified view of exact continuous penalties for $\ell_2 - \ell_0$ minimization. *SIAM Journal on Optimization*, 27(3):2034–2060.
- [Stallinga and Rieger, 2010] Stallinga, S. and Rieger, B. (2010). Accuracy of the Gaussian point spread function model in 2D localization microscopy. *Optics Express*, 18(24):24461–24476.

- [Ströhl and Kaminski, 2015] Ströhl, F. and Kaminski, C. F. (2015). A joint Richardson-Lucy deconvolution algorithm for the reconstruction of multifocal structured illumination microscopy data. *Methods and Applications in Fluorescence*, 3(1):014002.
- [Suárez-Díaz et al., 2021] Suárez-Díaz, J. L., García, S., and Herrera, F. (2021). A tutorial on distance metric learning: mathematical foundations, algorithms, experimental analysis, prospects and challenges. *Neurocomputing*, 425:300–322.
- [Sun et al., 2006] Sun, Q.-Y., Wang, C.-Y., and Shi, Z.-J. (2006). Global convergence of a modified gradient projection method for convex constrained problems. *Acta Mathematicae Applicatae Sinica. English Series*, 22(2):227–242.
- [Sun et al., 2017] Sun, Y., Babu, P., and Palomar, D. (2017). Majorization-Minimization algorithms in signal processing, communications, and machine learning. *IEEE Transactions on Signal Processing*, 65:794–816.
- [Sung et al., 2018] Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., and Hospedales, T. M. (2018). Learning to compare: Relation network for few-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1199–1208.
- [Szołomicka and Markowska-Kaczmar, 2023] Szołomicka, J. and Markowska-Kaczmar, U. (2023). An overview of few-shot learning methods in analysis of histopathological images. *Advances in Smart Healthcare Paradigms and Applications: Outstanding Women in Healthcare—Volume 1*, pages 87–113.
- [Tang et al., 2019] Tang, M., Marin, D., Ben Ayed, I., and Boykov, Y. (2019). Kernel cuts: Kernel and spectral clustering meet regularization. *International Journal of Computer Vision*, 127:477–511.
- [Tao et al., 2023] Tao, R., Chen, H., and Savvides, M. (2023). Boosting transductive few-shot fine-tuning with margin-based uncertainty weighting and probability regularization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15752–15761.
- [Thai et al., 2013] Thai, T. H., Cogranne, R., and Retraint, F. (2013). Camera model identification based on the heteroscedastic noise model. *IEEE Transactions on Image Processing*, 23(1):250–263.
- [Tian et al., 2023] Tian, L., Feng, J., Chai, X., Chen, W., Wang, L., Liu, X., and Chen, B. (2023). Prototypes-oriented transductive few-shot learning with conditional transport. In *IEEE International Conference on Computer Vision (ICCV)*, pages 16317–16326.
- [Tian et al., 2020] Tian, Y., Wang, Y., Krishnan, D., Tenenbaum, J. B., and Isola, P. (2020). Rethinking few-shot image classification: a good embedding is all you need? In *European Conference on Computer Vision (ECCV)*.
- [Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.
- [Tikhonov, 1963] Tikhonov, A. N. (1963). On the solution of ill-posed problems and the method of regularization. In *Doklady Akademii Nauk*, volume 151, pages 501–504. Russian Academy of Sciences.
- [Toint, 1988] Toint, P. L. (1988). Global convergence of a class of trust-region methods for nonconvex minimization in Hilbert space. *IMA Journal of Numerical Analysis*, 8(2):231–252.

- [Trosten et al., 2023] Trosten, D. J., Chakraborty, R., Løkse, S., Wickstrøm, K. K., Jenssen, R., and Kampffmeyer, M. C. (2023). Hubs and hyperspheres: Reducing hubness and improving transductive few-shot learning with hyperspherical embeddings. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7527–7536.
- [Tseng, 2001] Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494.
- [Ulyanov et al., 2018] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). Deep image prior. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 9446–9454.
- [Van Den Bent, 2010] Van Den Bent, M. J. (2010). Interobserver variation of the histopathological diagnosis in clinical trials on glioma: a clinician’s perspective. *Acta Neuropathologica*, 120(3):297–304.
- [Van den Dries and Miller, 1996] Van den Dries, L. and Miller, C. (1996). Geometric categories and o-minimal structures. *Duke Mathematical Journal*, 84(2):497 – 540.
- [Van Kempen et al., 1997] Van Kempen, G. M., van Vliet, L. J., Verveer, P. J., and van der Voort, H. T. (1997). A quantitative comparison of image restoration methods for confocal microscopy. *Journal of Microscopy*, 185(3):354–365.
- [Vapnik, 1999] Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999.
- [Vatankhah et al., 2014] Vatankhah, S., Renaut, R. A., and Ardestani, V. E. (2014). Regularization parameter estimation for underdetermined problems by the χ^2 principle with application to 2D focusing gravity inversion. *Inverse Problems*, 30(8):085002.
- [Veilleux et al., 2021] Veilleux, O., Boudiaf, M., Piantanida, P., and Ben Ayed, I. (2021). Realistic evaluation of transductive few-shot learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 34.
- [Vicidomini et al., 2009] Vicidomini, G., Boccacci, P., Diaspro, A., and Bertero, M. (2009). Application of the split-gradient method to 3D image deconvolution in fluorescence microscopy. *Journal of Microscopy*, 234(1):47–61.
- [Villa et al., 2013] Villa, S., Salzo, S., Baldassarre, L., and Verri, A. (2013). Accelerated and inexact forward-backward algorithms. *SIAM Journal on Optimization*, 23(3):1607–1633.
- [Vinyals et al., 2016] Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. (2016). Matching networks for one shot learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 29.
- [Von Tiedemann et al., 2006] Von Tiedemann, M., Fridberger, A., Ulfendahl, M., and De Monvel, J. B. (2006). Image adaptive point-spread function estimation and deconvolution for in vivo confocal microscopy. *Microscopy Research and Technique*, 69(1):10–20.
- [Vorontsov and Jefferies, 2017] Vorontsov, S. V. and Jefferies, S. M. (2017). A new approach to blind deconvolution of astronomical images. *Inverse Problems*, 33(5):055004.

- [Vũ, 2013] Vũ, B. C. (2013). A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Advances in Computational Mathematics*, 38(3):667–681.
- [Wahba, 1977] Wahba, G. (1977). Practical approximate solutions to linear operator equations when the data are noisy. *SIAM Journal on Numerical Analysis*, 14(4):651–667.
- [Wald and Schuster, 2017] Wald, A. and Schuster, T. (2017). Sequential subspace optimization for non-linear inverse problems. *Journal of Inverse and Ill-Posed Problems*, 25(1):99–117.
- [Wang et al., 2017] Wang, R., Li, H.-C., Pizurica, A., Li, J., Plaza, A., and Emery, W. J. (2017). Hyperspectral unmixing using double reweighted sparse regression and total variation. *IEEE Geoscience and Remote Sensing Letters*, 14(7):1146–1150.
- [Wang et al., 2019] Wang, Y., Chao, W.-L., Weinberger, K. Q., and Van der Maaten, L. (2019). SimpleShot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv preprint arXiv:1911.04623*.
- [Wang and Ma, 2009] Wang, Y. and Ma, S. (2009). A fast subspace method for image deblurring. *Applied Mathematics and Computation*, 215(6):2359–2377.
- [Wang et al., 2020] Wang, Y., Xu, C., Liu, C., Zhang, L., and Fu, Y. (2020). Instance credibility inference for few-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12836–12845.
- [Wang et al., 2022] Wang, Z., Wu, Z., Agarwal, D., and Sun, J. (2022). MedCLIP: Contrastive learning from unpaired medical images and text. *Conference on Empirical Methods in Natural Language Processing*.
- [Wang et al., 2004] Wang, Z. H., Wen, Z. W., and Yuan, Y.-X. (2004). A subspace trust region method for large scale unconstrained optimization. In *Numerical Linear Algebra and Optimization*, pages 264–274. Science Press.
- [Wang and Yuan, 2006] Wang, Z.-H. and Yuan, Y.-X. (2006). A subspace implementation of quasi-Newton trust region methods for unconstrained optimization. *Numerische Mathematik*, 104(2):241–269.
- [Wedderburn, 1976] Wedderburn, R. (1976). On the existence and uniqueness of the maximum likelihood estimates for certain generalized linear models. *Biometrika*, 63(1):27–32.
- [Wei et al., 2016] Wei, X., Cao, B., and Yu, P. S. (2016). Nonlinear joint unsupervised feature selection. In *Proceedings of the 16th SIAM International Conference on Data Mining*, pages 414–422, Miami, USA. SIAM.
- [Wertheimer and Hariharan, 2019] Wertheimer, D. and Hariharan, B. (2019). Few-shot learning with localization in realistic settings. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6558–6567.
- [Wu et al., 2011] Wu, C., Zhang, J., and Tai, X.-C. (2011). Augmented Lagrangian method for total variation restoration with non-quadratic fidelity. *Inverse problems and imaging*, 5(1):237–261.

- [Wu et al., 2005] Wu, K., Otoo, E., and Shoshani, A. (2005). Optimizing connected component labeling algorithms. In *Medical Imaging 2005: Image Processing*, volume 5747, pages 1965–1976. SPIE.
- [Xiang et al., 2021] Xiang, K., Jiang, B., and Shang, D. (2021). The overview of the deep learning integrated into the medical imaging of liver: A review. *Hepatology International*, 15:868–880.
- [Xiao et al., 2010] Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. (2010). Sun database: Large-scale scene recognition from abbey to zoo. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3485–3492. IEEE.
- [Xu et al., 2016] Xu, C., Lin, Z., Zhao, Z., and Zha, H. (2016). Relaxed majorization-minimization for non-smooth and non-convex optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.
- [Xu et al., 2014a] Xu, L., Ren, J. S., Liu, C., and Jia, J. (2014a). Deep convolutional neural network for image deconvolution. *Advances in Neural Information Processing Systems (NeurIPS)*, 27.
- [Xu and Yin, 2013] Xu, Y. and Yin, W. (2013). A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences*, 6(3):1758–1789.
- [Xu et al., 2014b] Xu, Z., Huang, G., Weinberger, K. Q., and Zheng, A. X. (2014b). Gradient boosted feature selection. In *Proceedings of the 20th International Conference on Knowledge Discovery and Data Mining*, pages 522–531, New York, USA.
- [Youla and Webb, 1982] Youla, D. C. and Webb, H. (1982). Image restoration by the method of convex projections: Part 1 theory. *IEEE Transactions on Medical Imaging*, 1(2):81–94.
- [Young et al., 2011a] Young, P., Clendenon, S., Byars, J., Decca, R., and Dunn, K. (2011a). The effects of spherical aberration on multiphoton fluorescence excitation microscopy. *Journal of Microscopy*, 242(2):157–165.
- [Young et al., 2011b] Young, P., Clendenon, S., Byars, J., and Dunn, K. (2011b). The effects of refractive index heterogeneity within kidney tissue on multiphoton fluorescence excitation microscopy. *Journal of Microscopy*, 242(2):148–156.
- [Yu et al., 2017] Yu, C., Zhao, J., Wang, Y., Wang, C., and Geng, W. (2017). Separation and imaging diffractions by a sparsity-promoting model and subspace trust-region algorithm. *Geophysical Journal International*, 208(3):1756–1763.
- [Yuan and Boykov, 2010] Yuan, J. and Boykov, Y. (2010). TV-based multi-label image segmentation with label cost prior. In *British Machine Vision Conference*, volume 2, page 4.
- [Yuan, 2008] Yuan, Y.-X. (2008). Subspace methods for large scale nonlinear equations and nonlinear least squares. *Optimization and Engineering*, 10(2):207–218.
- [Yuan, 2015] Yuan, Y.-X. (2015). Recent advances in trust region algorithms. *Mathematical Programming*, 151(1):249–281.

- [Zanella et al., 2009] Zanella, R., Boccacci, P., Zanni, L., and Bertero, M. (2009). Efficient gradient projection methods for edge-preserving removal of Poisson noise. *Inverse Problems*, 25(4):045010.
- [Zhang et al., 2007] Zhang, B., Zerubia, J., and Olivo-Marin, J.-C. (2007). Gaussian approximations of fluorescence microscope point-spread function models. *Applied Optics*, 46(10):1819–1829.
- [Zhang et al., 2014] Zhang, B., Zhu, Z., and Li, S. (2014). A modified spectral conjugate gradient projection algorithm for total variation image restoration. *Applied Mathematics Letters. An International Journal of Rapid Publication*, 27:26–35.
- [Zhang et al., 2017] Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2017). Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155.
- [Zhang et al., 2022] Zhang, R., Zhang, W., Fang, R., Gao, P., Li, K., Dai, J., Qiao, Y., and Li, H. (2022). Tip-adapter: Training-free adaption of clip for few-shot classification. In *European Conference on Computer Vision (ECCV)*, pages 493–510. Springer.
- [Zhou et al., 2022a] Zhou, K., Yang, J., Loy, C. C., and Liu, Z. (2022a). Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16816–16825.
- [Zhou et al., 2022b] Zhou, K., Yang, J., Loy, C. C., and Liu, Z. (2022b). Learning to prompt for vision-language models. In *International Journal of Computer Vision*, volume 130, pages 2337–2348. Springer.
- [Zhou et al., 2019] Zhou, Y., Caillau, J.-B., Antonini, M., and Barlaud, M. (2019). Robust classification with feature selection using alternating minimization and Douglas-Rachford splitting method. *hal preprint hal-01993753*.
- [Zhu and Koniusz, 2023] Zhu, H. and Koniusz, P. (2023). Transductive few-shot learning with prototype-based label propagation by iterative graph refinement. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23996–24006.
- [Zhu and Yuille, 1996] Zhu, S. C. and Yuille, A. (1996). Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900.
- [Zhu and Zhang, 2013] Zhu, X. and Zhang, D. (2013). Efficient parallel Levenberg-Marquardt model fitting towards real-time automated parametric imaging microscopy. *PLOS One*, 8(10):e76665.
- [Zibulevsky, 2008] Zibulevsky, M. (2008). SESOP-TN: Combining sequential subspace optimization with truncated Newton method. Technical report, Computer Science Department, Technion.
- [Zibulevsky and Elad, 2010] Zibulevsky, M. and Elad, M. (2010). $\ell_1 - \ell_2$ optimization in signal and image processing. *IEEE Signal Processing Magazine*, 27(3):76–88.
- [Ziko et al., 2021] Ziko, I., Boudiaf, M., Dolz, J., Granger, E., and Ben Ayed, I. (2021). Transductive few-shot learning: clustering is all you need? *arXiv e-prints*, pages arXiv–2106.

- [Ziko et al., 2020] Ziko, I., Dolz, J., Granger, E., and Ayed, I. B. (2020). Laplacian regularized few-shot learning. In *International Conference on Machine Learning (ICML)*, pages 11660–11670. PMLR.
- [Zipfel et al., 2003] Zipfel, W. R., Williams, R. M., and Webb, W. W. (2003). Nonlinear magic: multi-photon microscopy in the biosciences. *Nature Biotechnology*, 21(11):1369–1377.

- *Je n'ai jamais vu de berger allemand qui aimait les épinards... !*
- *C'est qu'elle ne se voit pas comme un chien.*
- *Ah... Et elle se voit comme quoi ?*
- *On dirait qu'elle se considère comme un être particulier, qui se situe au-delà de ces catégories.*
- *Un super-dog ?*
- *Par exemple.*
- *Et c'est pour cela qu'elle aimerait les épinards ?*
- *Non, il n'y a aucun rapport.*

Haruki Murakami, 1Q84 (Livre 1)