

Mathematics in Industry

Volume 37

Series Editors

Hans Georg Bock, Interdisciplinary Center for Scientific Computing IWR,
Heidelberg University, Heidelberg, Germany

Frank de Hoog, CSIRO, Canberra, Australia

Avner Friedman, Ohio State University, Columbus, OH, USA

Arvind Gupta, University of British Columbia, Vancouver, BC, Canada

André Nachbin, IMPA, Rio de Janeiro, RJ, Brazil

Tohru Ozawa, Waseda University, Tokyo, Japan

William R. Pulleyblank, United States Military Academy, West Point, NY, USA

Torgeir Rusten, Det Norske Veritas, Høvik, Norway

Fadil Santosa, University of Minnesota, Minneapolis, MN, USA

Jin Keun Seo, Yonsei University, Seoul, Korea (Republic of)

Anna-Karin Tornberg, Royal Institute of Technology (KTH), Stockholm, Sweden

Mathematics in Industry focuses on the research and educational aspects of mathematics used in industry and other business enterprises. Books for *Mathematics in Industry* are in the following categories: research monographs, problem-oriented multi-author collections, textbooks with a problem-oriented approach, conference proceedings. Relevance to the actual practical use of mathematics in industry is the distinguishing feature of the books in the *Mathematics in Industry* series.

More information about this series at <https://link.springer.com/bookseries/4650>

Jong Chul Ye

Geometry of Deep Learning

A Signal Processing Perspective

 Springer

Jong Chul Ye
Korea Advanced Institute of Science
and Technology (KAIST),
Daejeon, Republic of Korea

ISSN 1612-3956 ISSN 2198-3283 (electronic)
Mathematics in Industry
ISBN 978-981-16-6045-0 ISBN 978-981-16-6046-7 (eBook)
<https://doi.org/10.1007/978-981-16-6046-7>

Mathematics Subject Classification: 68T01, 68T07

© The Editor(s) (if applicable) and The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2022

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

To Andy, Ella, and Joo

Preface

It was a very different, unprecedented, and weird start of the semester, and I did not know what to do. This semester, I was supposed to offer a new senior-level undergraduate class on *Advanced Intelligence* to jointly teach students at the Department of Bio/Brain Engineering and the Department of Mathematical Sciences. I had initially planned a standard method for teaching machine learning, the contents of which are practical, experience-based lectures with a lot of interaction with the students through many mini-projects and term projects. Unfortunately, the global pandemic of COVID-19 has completely changed the world and such interactive classes are no longer an option most of the time.

So, I thought about the best way to give online lectures to my students. I wanted my class to be different from other popular online machine learning courses but still provide up-to-date information about modern deep learning. However, not many options were available. Most existing textbooks are already outdated or very implementation oriented without touching the basics. One option would be to prepare presentation slides by adding all the up-to-date knowledge that I wanted to teach. However, for undergraduate-level courses, the presentation files are usually not enough for students to follow the class, and we need a textbook that students can read independently to understand the class. For this reason, I decided to write a reading material first and then create presentation files based on it, so that the students can learn independently before and after the online lectures. This was the start of my semester-long book project on *Geometry of Deep Learning*.

In fact, it has been my firm belief that a deep neural network is not a magic black box, but rather a source of endless inspiration for new mathematical discoveries. Also, I believed in the famous quote by Isaac Newton, “Standing on the shoulders of giants,” and looking for a mathematical interpretation of deep learning. For me as a medical imaging researcher, this topic was critical not only from a theoretical point of view but also for clinical decision-making, because we do not want to create false features that can be recognized as diseases.

In 2017, on a street in Lisbon, I had *Eureka!* moment in understanding hidden framelet structure in encoder-decoder neural networks. The resulting interpretation of the deep convolutional framelets, published in the *SIAM Journal of Imaging*

Science, has had a significant impact on the applied math community and has been one of the most downloaded papers since its publication. However, the role of the rectified linear unit (ReLU) was not clear in this work, and one of the reviewers in a medical imaging journal consistently asked me to explain the role of the ReLU in deep neural networks. At first, this looked like a question that went beyond the scope of the medical application paper, but I am grateful to the reviewer, as during the agony of preparing the answers to the question, I realized that the ReLU determines the input space partitioning, which is automatically adapted to the input space manifold. In fact, this finding led to a 2019 ICML paper, in which we revealed the combinatorial representation of framelets, which clearly shows the crucial connection with the classic compressed sensing (CS) approaches.

Looking back, I was pretty brave to start this book project, as these are just two pieces of my geometric understanding of deep learning. However, as I was preparing the reading material for each subject of deep learning, I found that there are indeed many exciting geometric insights that have not been fully discussed.

For example, when I wrote the chapter on backpropagation, I recognized the importance of the denominator layout convention in the matrix calculus, which led to the beautiful geometry of the backpropagation. Before writing this book, the normalization and attention mechanisms looked very heuristic to me, with no evidence of a systematic understanding that is even more confusing due to their similarities. For example, AdaIN, Transformer, and BERT were like dark recipes that researchers have developed with their own secret sauces. However, an in-depth study for the preparation of the reading material has revealed a very nice mathematical structure behind their intuition, which shows a close connection between them and their relationship to optimal transport theory.

Writing a chapter on the geometry of deep neural networks was another joy that broadened my insight. During my lecture, one of my students pointed out that some partitions can lead to a low-rank mapping. In retrospect, this was already in the equation, but it was not until my students challenged me that I recognized the beautiful geometry of the partition, which fits perfectly with fascinating empirical observations of the deep neural network.

The last chapter, on generative models and unsupervised learning, is something of which I am very proud. In contrast to the conventional explanation of the generative adversarial network (GAN), variational auto-encoder (VAE), and normalizing flows with probabilistic tools, my main focus was to derive them with geometric tools. In fact, this effort was quite rewarding, and this chapter clearly unified various forms of generative model as statistical distance minimization and optimal transport problems.

In fact, the focus of this book is to give students a geometric insight that can help them understand deep learning in a unified framework, and I believe that this is one of the first deep learning books written from such a perspective. As this book is based on the materials that I have prepared for my senior-level undergraduate class, I believe that this book can be used for one-semester-long senior-level undergraduate and graduate-level classes. In addition, my class was a code-shared course for

both bioengineering and math students, so that much of the content of the work is interdisciplinary, which tries to appeal to students in both disciplines.

I am very grateful to my TAs and students of the 2020 spring class of BiS400C and MAS480. I would especially like to thank my great team of TAs: Sangjoon Park, Yujin Oh, Chanyong Jung, Byeongsu Sim, Hyungjin Chung, and Gyutaek Oh. Sangjoon, in particular, has done a tremendous job as Head TA and provided organized feedback on the typographical errors and mistakes of this book. I would also like to thank my wonderful team at the Bio Imaging, Signal Processing and Learning laboratory (BISPL) at KAIST, who have produced ground-breaking research works that have inspired me.

Many thanks to my awesome son and future scientist, Andy Sangwoo, and my sweet daughter and future writer, Ella Jiwoo, for their love and support. You are my endless source of energy and inspiration, and I am so proud of you. Last, but not the least, I would like to thank my beloved wife, Seungjoo (Joo), for her endless love and constant support ever since we met. I owe you everything and you made me a good man. With my warmest thanks,

Daejeon, Korea
February, 2021

Jong Chul Ye

Contents

Part I Basic Tools for Machine Learning

1	Mathematical Preliminaries	3
1.1	Metric Space	3
1.2	Vector Space	4
1.3	Banach and Hilbert Space	6
1.3.1	Basis and Frames	7
1.4	Probability Space	9
1.5	Some Matrix Algebra	11
1.5.1	Kronecker Product	13
1.5.2	Matrix and Vector Calculus	15
1.6	Elements of Convex Optimization	17
1.6.1	Some Definitions	17
1.6.2	Convex Sets, Convex Functions	19
1.6.3	Subdifferentials	20
1.6.4	Convex Conjugate	21
1.6.5	Lagrangian Dual Formulation	24
1.7	Exercises	27
2	Linear and Kernel Classifiers	29
2.1	Introduction	29
2.2	Hard-Margin Linear Classifier	31
2.2.1	Maximum Margin Classifier for Separable Cases	31
2.2.2	Dual Formulation	33
2.2.3	KKT Conditions and Support Vectors	35
2.3	Soft-Margin Linear Classifiers	36
2.3.1	Maximum Margin Classifier with Noise	36
2.4	Nonlinear Classifier Using Kernel SVM	39
2.4.1	Linear Classifier in the Feature Space	39
2.4.2	Kernel Trick	40
2.5	Classical Approaches for Image Classification	42
2.6	Exercises	43

- 3 Linear, Logistic, and Kernel Regression** 45
 - 3.1 Introduction 45
 - 3.2 Linear Regression 46
 - 3.2.1 Ordinary Least Squares (OLS) 46
 - 3.3 Logistic Regression 48
 - 3.3.1 Logits and Linear Regression 48
 - 3.3.2 Multiclass Classification Using Logistic Regression 50
 - 3.4 Ridge Regression 51
 - 3.5 Kernel Regression 52
 - 3.6 Bias–Variance Trade-off in Regression 55
 - 3.6.1 Examples 56
 - 3.7 Exercises 58
- 4 Reproducing Kernel Hilbert Space, Representer Theorem** 61
 - 4.1 Introduction 61
 - 4.2 Reproducing Kernel Hilbert Space (RKHS) 62
 - 4.2.1 Feature Map and Kernels 63
 - 4.2.2 Definition of RKHS 65
 - 4.3 Representer Theorem 68
 - 4.4 Application of Representer Theorem 69
 - 4.4.1 Kernel Ridge Regression 69
 - 4.4.2 Kernel SVM 71
 - 4.5 Pros and Cons of Kernel Machines 73
 - 4.6 Exercises 74

Part II Building Blocks of Deep Learning

- 5 Biological Neural Networks** 79
 - 5.1 Introduction 79
 - 5.2 Neurons 80
 - 5.2.1 Anatomy of Neurons 80
 - 5.2.2 Signal Transmission Mechanism 80
 - 5.2.3 Synaptic Plasticity 82
 - 5.3 Biological Neural Network 84
 - 5.3.1 Visual System 84
 - 5.3.2 Hubel and Wiesel Model 86
 - 5.3.3 Jennifer Aniston Cell 88
 - 5.4 Exercises 90
- 6 Artificial Neural Networks and Backpropagation** 91
 - 6.1 Introduction 91
 - 6.2 Artificial Neural Networks 91
 - 6.2.1 Notation 91
 - 6.2.2 Modeling a Single Neuron 92
 - 6.2.3 Feedforward Multilayer ANN 95

- 6.3 Artificial Neural Network Training..... 96
 - 6.3.1 Problem Formulation 96
 - 6.3.2 Optimizers 97
- 6.4 The Backpropagation Algorithm 102
 - 6.4.1 Derivation of the Backpropagation Algorithm 102
 - 6.4.2 Geometrical Interpretation of BP Algorithm 105
 - 6.4.3 Variational Interpretation of BP Algorithm 106
 - 6.4.4 Local Variational Formulation 109
- 6.5 Exercises 110
- 7 Convolutional Neural Networks 113**
 - 7.1 Introduction..... 113
 - 7.2 History of Modern CNNs 114
 - 7.2.1 AlexNet 114
 - 7.2.2 GoogLeNet..... 115
 - 7.2.3 VGGNet..... 116
 - 7.2.4 ResNet..... 117
 - 7.2.5 DenseNet 117
 - 7.2.6 U-Net..... 118
 - 7.3 Basic Building Blocks of CNNs 119
 - 7.3.1 Convolution 119
 - 7.3.2 Pooling and Unpooling 120
 - 7.3.3 Skip Connection 124
 - 7.4 Training CNNs 125
 - 7.4.1 Loss Functions 125
 - 7.4.2 Data Split 126
 - 7.4.3 Regularization 127
 - 7.5 Visualizing CNNs 128
 - 7.6 Applications of CNNs..... 130
 - 7.7 Exercises 131
- 8 Graph Neural Networks 135**
 - 8.1 Introduction..... 135
 - 8.2 Mathematical Preliminaries..... 137
 - 8.2.1 Definition 138
 - 8.2.2 Graph Isomorphism 138
 - 8.2.3 Graph Coloring 139
 - 8.3 Related Works 139
 - 8.3.1 Word Embedding 140
 - 8.3.2 Loss Function 144
 - 8.4 Graph Embedding 146
 - 8.4.1 Matrix Factorization Approaches 146
 - 8.4.2 Random Walks Approaches 147
 - 8.4.3 Neural Network Approaches 148

8.5	WL Test, Graph Neural Networks	150
8.5.1	Weisfeiler–Lehman Isomorphism Test	150
8.5.2	Graph Neural Network as WL Test	152
8.6	Summary and Outlook	153
8.7	Exercises	153
9	Normalization and Attention	155
9.1	Introduction	155
9.1.1	Notation	156
9.2	Normalization	157
9.2.1	Batch Normalization	157
9.2.2	Layer and Instance Normalization	159
9.2.3	Adaptive Instance Normalization (AdaIN)	161
9.2.4	Whitening and Coloring Transform (WCT)	163
9.3	Attention	164
9.3.1	Metabotropic Receptors: Biological Analogy	164
9.3.2	Mathematical Modeling of Spatial Attention	166
9.3.3	Channel Attention	168
9.4	Applications	169
9.4.1	StyleGAN	169
9.4.2	Self-Attention GAN	170
9.4.3	Attentional GAN: Text to Image Generation	172
9.4.4	Graph Attention Network	172
9.4.5	Transformer	174
9.4.6	BERT	178
9.4.7	Generative Pre-trained Transformer (GPT)	182
9.4.8	Vision Transformer	185
9.5	Mathematical Analysis of Normalization and Attention	185
9.6	Exercises	189

Part III Advanced Topics in Deep Learning

10	Geometry of Deep Neural Networks	195
10.1	Introduction	195
10.1.1	Desiderata of Machine Learning	196
10.2	Case Studies	197
10.2.1	Single–Layer Perceptron	197
10.2.2	Frame Representation	198
10.3	Convolution Framelets	203
10.3.1	Convolution and Hankel Matrix	203
10.3.2	Convolution Framelet Expansion	205
10.3.3	Link to CNN	206
10.3.4	Deep Convolutional Framelets	208
10.4	Geometry of CNN	211
10.4.1	Role of Nonlinearity	211
10.4.2	Nonlinearity Is the Key for Inductive Learning	211

- 10.4.3 Expressivity 212
- 10.4.4 Geometric Meaning of Features 213
- 10.4.5 Geometric Understanding of Autoencoder 220
- 10.4.6 Geometric Understanding of Classifier 222
- 10.5 Open Problems 223
- 10.6 Exercises 225
- 11 Deep Learning Optimization 227**
 - 11.1 Introduction 227
 - 11.2 Problem Formulation 228
 - 11.3 Polyak–Łojasiewicz-Type Convergence Analysis 229
 - 11.3.1 Loss Landscape and Over-Parameterization 232
 - 11.4 Lyapunov-Type Convergence Analysis 235
 - 11.4.1 The Neural Tangent Kernel (NTK) 237
 - 11.4.2 NTK at Infinite Width Limit 238
 - 11.4.3 NTK for General Loss Function 240
 - 11.5 Exercises 241
- 12 Generalization Capability of Deep Learning 243**
 - 12.1 Introduction 243
 - 12.2 Mathematical Preliminaries 244
 - 12.2.1 Vapnik–Chervonenkis (VC) Bounds 246
 - 12.2.2 Rademacher Complexity Bounds 251
 - 12.2.3 PAC–Bayes Bounds 254
 - 12.3 Reconciling the Generalization Gap via Double Descent Model 256
 - 12.4 Inductive Bias of Optimization 260
 - 12.5 Generalization Bounds via Algorithm Robustness 261
 - 12.6 Exercises 265
- 13 Generative Models and Unsupervised Learning 267**
 - 13.1 Introduction 267
 - 13.2 Mathematical Preliminaries 269
 - 13.3 Statistical Distances 272
 - 13.3.1 f -Divergence 272
 - 13.3.2 Wasserstein Metric 274
 - 13.4 Optimal Transport 277
 - 13.4.1 Monge’s Original Formulation 277
 - 13.4.2 Kantorovich Formulation 278
 - 13.4.3 Entropy Regularization 281
 - 13.5 Generative Adversarial Networks 283
 - 13.5.1 Earliest Form of GAN 283
 - 13.5.2 f -GAN 285
 - 13.5.3 Wasserstein GAN (W-GAN) 288
 - 13.5.4 StyleGAN 290

- 13.6 Autoencoder-Type Generative Models 291
 - 13.6.1 ELBO 291
 - 13.6.2 Variational Autoencoder (VAE)..... 292
 - 13.6.3 β -VAE..... 296
 - 13.6.4 Normalizing Flow, Invertible Flow 298
- 13.7 Unsupervised Learning via Image Translation 301
 - 13.7.1 Pix2pix 302
 - 13.7.2 CycleGAN 303
 - 13.7.3 StarGAN 306
 - 13.7.4 Collaborative GAN 307
- 13.8 Summary and Outlook 311
- 13.9 Exercises 311
- 14 Summary and Outlook..... 315**
- 15 Bibliography 317**
- Index..... 327**