EXPLOITING THE LATENT SPACE OF DEEP GENERATIVE MODELS FOR BETTER GENERATION, REPRESENTATION LEARNING, AND GENERATIVE TRANSFER

ARNAB KUMAR MONDAL

AMAR NATH AND SHASHI KHOSLA SCHOOL OF INFORMATION TECHNOLOGY

INDIAN INSTITUTE OF TECHNOLOGY DELHI

JULY 2024

Exploiting the Latent Space of Deep Generative Models for Better Generation, Representation Learning, and Generative Transfer

by

Arnab Kumar Mondal
Amar Nath and Shashi Khosla School of Information Technology

Submitted
in fulfillment of the requirements of the degree of Doctor of Philosophy

to the

Indian Institute of Technology Delhi

July 2024

THESIS CERTIFICATE

This is to certify that the thesis titled **Exploiting the Latent Space of Deep Generative Models for Better Generation, Representation Learning, and Generative Transfer** for dissertations, submitted by **Arnab Kumar Mondal (2018ANZ8380)**, to the **Indian Institute of Technology, Delhi**, for the award of the degree of **Doctor of Philosophy**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma unless otherwise stated explicitly. In particular, work done in chapter 3 was done jointly with an undergraduate and a master's student, and the part done by the collaborators appeared in their respective bachelor's and master's thesis.

Prathosh A.P.
Assistant Professor,
Dept. of Electrical Communication Engg.,
Indian Institute of Science (IISc),
Bengaluru, India - 560012

Parag Singla
Professor,
Dept. of Computer Science and Engg.,
Indian Institute of Technology Delhi,
New Delhi, India - 110016

Acknowledgements

Embarking on the journey of a Ph.D. after leaving my government job of five years was one of the most challenging decisions I have ever made. This transition was fraught with uncertainties and difficulties, but none of it would have been possible without the unwavering support of my beloved wife, Shweta. Her encouragement, understanding, and constant motivation have been the pillars upon which I have built this academic pursuit. Shweta's sacrifices, patience, and belief in my dreams have played a crucial role in overcoming the hurdles of this journey. Additionally, she meticulously proofread the final manuscripts of all my papers when they were near submission, ensuring their quality and coherence.

I vividly remember my first interaction with my supervisor, Professor Prathosh AP. I went to meet him before joining the program. After meeting him, I felt a sense of reassurance and confidence that I would be able to navigate the complexities of my Ph.D. under his exceptional guidance. His expertise, insightful advice, and unwavering support have been invaluable throughout this journey. He spent countless hours sitting alongside me, meticulously polishing and guiding my work. His commitment to my success went beyond mere supervision. He has been a friend, philosopher, and guide in the truest sense of the phrase.

I took the 'Machine Learning' course during my second semester in the program, where I met my other supervisor, Professor Parag Singla. His pursuit of perfection significantly motivated me and other students. Professor Singla's meticulous teaching, deep expertise, and commitment to excellence profoundly impacted my academic journey. He often stayed awake with me, refining research papers until the last moment and drafting rebuttal documents, ensuring every detail was perfect. These experiences taught me perseverance, dedication, and attention to detail. His guidance inspired me to push the boundaries of my research and personal life, embracing challenges head-on. His invaluable feedback and relentless support were instrumental not only in my growth as a researcher but also helped me become a more thoughtful, ambitious, and well-rounded individual.

I also sincerely appreciate Professor Himanshu Asnani, who guided me in my initial works. His early mentorship laid a strong foundation for my research, and his insights have been crucial in shaping the direction of my studies.

The roots of my research journey trace back to my undergraduate days. Professor Swagatam Das, my first research supervisor, and my friends Nasir and Soumyadip encouraged and supported me to set the stage for my continued academic pursuits.

Additionally, I extend my heartfelt gratitude to all my collaborators: Sankalan, Aravind, Ajay, Piyush, and Lakshya. Our numerous late-night sessions, filled with discussions, idea inception, theory formulation, and brainstorming, have been fundamental to the development of this work. Their dedication, insights, and collaborative spirit have significantly enriched this research, and their camaraderie has made this journey all the more rewarding.

I would also like to extend my heartfelt gratitude to my other collaborators: Arnab (yes, a namesake!), Ayush, Chandrabhushan, Rushil, Harman, Indu, Poorva, and Sanket. Working with them was a truly enriching experience, leading to the publication of various research papers. Although these works are not directly related to my thesis, the knowledge and insights gained from our collaborations have contributed significantly to my overall academic growth. On various occasions, such as interview preparation and seeking advice, Yatin was my go-to person. He was always just a phone call away, ready to help and offer invaluable assistance.

I am profoundly grateful to my colleagues (read friends) from Fujitsu Research India Private Limited (FRIPL), Jay, Vishal, Sumit, and Niraj, who kept me motivated and assisted me during the final phase of my Ph.D. Their encouragement and support helped me push through the last hurdles with confidence and determination.

I am deeply thankful for the unconditional support of my parents and in-laws. Their encouragement and understanding were crucial not only when I initially embarked on this journey but also throughout the entire tenure of my Ph.D. My parents have always wanted me to pursue higher studies, and they must be very pleased and proud that I am finally so close to achieving my Ph.D. Their relentless belief in my abilities and constant support have been a source of immense strength and motivation. My sister and sister-in-law also provided tremendous support and encouragement, which have been instrumental throughout this journey.

A special mention goes to my son, Aniruddha, born during my Ph.D. journey. The joy of playing with him and looking at his smiling and innocent face after returning home from college brought me immense energy and happiness, helping me stay motivated through the most challenging times.

This journey has been a tapestry woven with the support, encouragement, and love of all these wonderful people. Their contributions have made the completion of this thesis possible and enriched my experience in ways that words cannot fully capture. Thank you for believing in me and guiding me in this challenging yet rewarding endeavor.

Throughout my Ph.D. journey, I faced numerous challenges, including multiple rejections of my research works. Each rejection was a difficult pill to swallow, testing my resolve and determination. During these times, I found solace and strength in the

teachings of Bhagwan Shri Krishna, particularly the shloka from the Bhagavad Gita:

कर्मण्येवाधिकारस्ते मा फलेषु कदाचन।
मा कर्मफलहेतुर्भूर्मा ते सङ्गोऽस्त्वकर्मणि॥ २–४७॥

These words deeply resonated with me, reminding me to stay focused on my efforts and not be disheartened by the outcomes. Shri Krishna's teachings gave me the clarity and resilience needed to navigate the turbulent ocean of Ph.D. research. His divine presence has been a constant source of inspiration and solace, guiding me through the difficult moments and encouraging me to persevere, for which I am eternally thankful. I dedicate this work to Him.

Arnab Kumar Mondal

Abstract

Keywords: Deep Latent Variable Generative Models; Regularized Autoencoders; Latent Space; Few-shot Generation.

The rise of deep neural networks has significantly advanced unsupervised generative modeling. Numerous Deep Generative Models (DGMs) have been proposed, including Variational Autoencoders (VAE) [KW14], Generative Adversarial Networks (GAN) [GPAM+14], Wasserstein Autoencoders (WAE) [TBGS18], Adversarial Autoencoders (AAE) [MSJG16], Autoregressive Models [vdOKE+16, OKK16], Normalizing Flow-based Models [KD18], Energy-based Models [LCH+06, DM19], and Diffusion Models [HJA20]. These models may be categorized along various dimensions, such as their architectural variations, presence or absence of explicit latent representation, training methodology and stability, density estimation capability, and time taken for sampling. In this thesis, our focus is on *Deep Latent Variable Generative Models* (DLVGMs), which refer to generative autoencoders (VAE, WAE, AAE) and GANs. This class of models uses low-dimensional latent representations of high-dimensional data, allowing learning informative low-dimensional representations for various downstream tasks (clustering, classification, and disentangling generative factors) while facilitating novel data generation. Interestingly, GANs are particularly notable for their high-quality generation but suffer from unstable training, mode collapse, and hyperparameter sensitivity. In contrast, Autoencoders with regularized latent spaces (VAE, WAE, AAE) offer stable training, interpretable inference, and efficient sampling, though their generated images are visually less impressive than those from GANs. In this thesis, we intend to address some of the complementary strengths and limitations of these DLVGMs. The thesis is organized in several parts, as outlined below.

In Part I (Prologue) of this thesis, we introduce various deep generative frameworks, define DLVGMs, highlight challenges (such as poor generation quality of RAEs, representation learning issues, and the need for large-scale data) associated with DLVGMs, and review existing solutions.

In Part II, 'Optimizing the Latent Space of RAEs for Improved Generation,' we diagnose the reasons behind the poor generation quality of generative AE frameworks by exploring two questions: 1. What is the 'optimal' latent dimension [MCJ+20], and 2. What is the 'optimal' latent prior [MASP21a] for a good generation? We hypothesize natural data generation as a two-step process involving a true low-dimensional latent

space and a non-linear mapping to a high-dimensional data space. We show that under the assumption of our data generation hypothesis, the best generation quality is achieved when the dimensionality of the generative AE's bottleneck layer matches the true latent dimensionality [MCJ+20]. In [MASP21a], we relax the assumption of simplistic Gaussian prior and learn the prior flexibly due to the lack of knowledge of true latent dimensionality.

In Part III, 'Optimizing the Latent Space of RAEs for Task-Specific Representation Learning,' we focus on representation learning in an RAE framework. First, we study the impact of bias-variance trade-off due to fixed prior distribution versus learnable priors on representation quality and demonstrate that learning the prior flexibly helps the model discover the actual data structure, improving clustering performance [MASP21b]. While [MASP21b] uses uni-modal data, we address disentangled representation learning in a multi-modal setting in [MSSA23]. We decompose the joint latent space into continuous and discrete components, each with domain-specific and domain-invariant representations. We demonstrate the effectiveness of these disentangled joint representations in downstream tasks like classification and generation. In our subsequent work [MST+23], we combine the representation learning aspect with the generation ability of RAEs to develop a framework for class-imbalance mitigation to enhance discriminating performance. Precisely, we propose a minority oversampling method that is distance-metric-free and class-preserving by design.

In Part IV, 'Few-shot Generation Using DLVGMs,' we address the issue that while DLVGMs offer a plethora of applications, they are data-hungry, limiting their applicability in real-world scenarios with data scarcity. Specifically, we develop techniques to transfer a source DLVGM built with large-scale data to a 'close' target domain with limited data. In [MTSA23], we perform few-shot 'generative domain adaptation' via inference-time latent-code learning by prepending a latent adapter network. While [MTSA23] achieves high-quality generation, it requires considerable time to generate due to inference-time optimization. In [MTSA24], we address this by learning to sample the parameters of the latent adapter network using a hypernetwork.

Finally, in Part V, we summarize our contributions and propose directions for future work based on the techniques and methods introduced in this thesis.

सार

डीप न्यूरल नेटवर्क्स के उदय ने अनसुपरवाइज्ड जेनरेटिव मॉडलिंग में प्रभावशाली प्रगति की है। विभिन्न प्रकार के डीप जेनरेटिव मॉडल्स (DGMs) प्रस्तावित किए गए हैं, जैसे कि वैरिएशनल ऑटोएन्कोडर (VAE) [KW14], जेनरेटिव एडवरसैरियल नेटवर्क (GAN) [GPAM+14], वासरस्टीन ऑटोएन्कोडर (WAE) [TBGS18], एडवरसैरियल ऑटोएन्कोडर (AAE) [MSJG16], ऑटोरिग्रेसिव मॉडल्स [vdOKE+16, OKK16], नॉर्मलाइजिंग फ्लो-बेस्ड मॉडल्स [KD18], एनर्जी-बेस्ड मॉडल्स [LCH+06, DM19], और डिफ्यूज़न मॉडल्स [HJA20]। इन मॉडलों को विभिन्न आयामों के आधार पर वर्गीकृत किया जा सकता है, जैसे उनकी आर्किटेक्चरल विविधताएँ, लेटेंट रिप्रेजेंटेशन की उपस्थिति या अनुपस्थिति, प्रशिक्षण पद्धति और स्थिरता, घनत्व अनुमान क्षमता, और सैंपलिंग के लिए समय। इस शोध-प्रबंध में, हमारा ध्यान 'डीप लैटेंट वेरिएबल जनरेटिव मॉडल्स' (DLVGMs) पर केंद्रित है, जो जनरेटिव ऑटोएन्कोडर्स (VAE, WAE, AAE) और GANs को संदर्भित करता है। इस वर्ग के मॉडल उच्च-आयामी डेटा के निम्न-आयामी लैटेंट रिप्रेजेंटेशन का उपयोग करते हैं, जिससे विभिन्न डाउनस्ट्रीम कार्यों (क्लस्टरिंग, वर्गीकरण, और जनरेटिव फैक्टर्स को डीसेंटैंगलिंग) के लिए जानकारीपूर्ण निम्न-आयामी रिप्रेजेंटेशन सीखते है, जबकि नए डेटा उत्पादन को भी सुविधाजनक बनाते हैं। GANs उच्च गुणवत्ता वाले उत्पादन के लिए विशेष रूप से उल्लेखनीय हैं, लेकिन वे अस्थिर प्रशिक्षण, मोड कोलैप्स, और हाइपरपैरामीटर संवेदनशीलता से ग्रस्त होते हैं। इसके विपरीत, रेगुलराइज़्ड लैटेंट स्पेस वाले ऑटोएन्कोडर्स (VAE, WAE, AAE) स्थिर प्रशिक्षण, व्याख्यात्मक इनफरेंस, और कुशल सैंपलिंग की पेशकश करते हैं, हालांकि उनके द्वारा उत्पन्न छवियां GANs की तुलना में दृश्य रूप से कम प्रभावशाली होती हैं। इस शोध-प्रबंध में, हम इन DLVGMs की कुछ पूरक ताकतों और सीमाओं को संबोधित करने का इरादा रखते हैं। इस शोध-प्रबंध का संगठन निम्नलिखित भागों में किया गया है।

इस शोध-प्रबंध के Part I (Prologue) में, हम विभिन्न डीप जनरेटिव फ्रेमवर्क्स का परिचय देते हैं, DLVGMs को परिभाषित करते हैं, DLVGMs से संबंधित चुनौतियों (जैसे RAE की खराब जनरेशन क्वालिटी, रिप्रेजेंटेशन लर्निंग समस्याएं, और बड़े पैमाने के डेटा की आवश्यकता) को उजागर करते हैं, और मौजूदा समाधानों की समीक्षा करते हैं।

Part II (Optimizing the Latent Space of RAEs for Improved Generation) में, हम जनरेटिव AE फ्रेमवर्क्स की खराब जनरेशन क्वालिटी के कारणों का निदान करते हैं, दो प्रश्नों का अन्वेषण करते हुए: 1. जनरेशन के लिए 'ऑप्टिमल' लैटेंट डाइमेंशन क्या है [MCJ+20], और 2. एक अच्छी जनरेशन के लिए 'ऑप्टिमल' लैटेंट प्रायर क्या है [MASP21a]? हम मानते हैं कि प्राकृतिक डेटा जनरेशन एक दो-चरणीय प्रक्रिया है जिसमें एक वास्तविक निम्न-आयामी लैटेंट स्पेस और एक उच्च-आयामी डेटा स्पेस के लिए एक नॉन-लिनियर मैपिंग शामिल है। हम यह दिखाते हैं कि हमारे डेटा जनरेशन परिकल्पना के तहत, सर्वश्रेष्ठ जनरेशन क्वालिटी तब प्राप्त होती है जब जेनरेटिव AE के बॉटलनेक लेयर की डाइमेंशनलिटी वास्तविक लेटेंट डाइमेंशनलिटी से मेल खाती है [MCJ+20]। [MASP21a] में, हम साधारण गौसियन प्रायर की धारणा को शिथिल करते हैं और वास्तविक लेटेंट डाइमेंशनलिटी के ज्ञान की कमी के कारण प्रायर को लचीले तरीके से सीखते हैं।

Part III (Optimizing the Latent Space of RAEs for Task-Specific Representation

Learning) में, हम एक RAE फ्रेमवर्क में रिप्रेजेंटेशन लर्निंग पर ध्यान केंद्रित करते हैं। सबसे पहले, हम फिक्स्ड प्रायर डिस्ट्रीब्यूशन बनाम लर्नेबल प्रायर्स के कारण बायस–वैरिएंस ट्रेड–ऑफ के प्रभाव का अध्ययन करते हैं और दिखाते हैं कि लचीले ढंग से प्रायर सीखने से मॉडल को वास्तविक डेटा संरचना का पता लगाने में मदद मिलती है, जिससे क्लस्टरिंग प्रदर्शन में सुधार होता है [MASP21b]। जबकि [MASP21b] युनि–मॉडल डेटा का उपयोग करता है, हम [MSSA23] में एक मल्टी–मॉडल सेटिंग में डिसेंटैंगल्ड रिप्रेजेंटेशन लर्निंग को संबोधित करते हैं। हम संयुक्त लेटेंट स्पेस को कंटीन्यूअस और डिस्क्रीट घटकों में विभाजित करते हैं, जिनमें प्रत्येक में डोमेन–स्पेसिफिक और डोमेन–इनवैरिएंट रिप्रेजेंटेशन होते हैं। हम इन डिसेंटैंगल्ड संयुक्त रिप्रेजेंटेशनों की प्रभावशीलता को डाउनस्ट्रीम कार्यों जैसे वर्गीकरण और जनरेशन में प्रदर्शित करते हैं। अपने अगले काम [MST+23] में, हम RAE के रिप्रेजेंटेशन लर्निंग पहलू को जनरेशन क्षमता के साथ मिलाकर एक फ्रेमवर्क विकसित करते हैं जो वर्ग–असंतुलन शमन के लिए है ताकि डिस्क्रिमिनेटिव प्रदर्शन को बढ़ाया जा सके। विशेष रूप से, हम एक अल्पसंख्यक ओवरसैंपलिंग पद्धति प्रस्तावित करते हैं जो डिस्टेंस–मेट्रिक–फ्री और डिज़ाइन द्वारा वर्ग–संरक्षणीय है।

Part IV (Few-shot Generation Using DLVGMs) में, हम इस मुद्दे को संबोधित करते हैं कि जबकि DLVGMs कई अनुप्रयोग प्रदान करते हैं, इसे प्रशिक्षण के लिए बहुत अधिक डेटा की आवश्यकता होती है, जो वास्तविक विश्व परिदृश्यों में डेटा की कमी के साथ उनकी प्रयोज्यता को सीमित करता है। विशेष रूप से, हम ऐसी तकनीकों का विकास करते हैं जो बड़े पैमाने पर डेटा का उपयोग करके बनाए गए सोर्स DLVGM को सीमित डेटा वाले 'संबंधित' टारगेट डोमेन के अनुकूल बनाती हैं। [MTSA23] में, हम एक लेटेंट अडैप्टर नेटवर्क को जोड़कर इन्फेरेंस–टाइम लेटेंट–कोड लर्निंग के माध्यम से 'फ्यू–शॉट जनरेटिव डोमेन अडैप्टेशन' करते हैं। जबकि [MTSA23] उच्च–गुणवत्ता की जनरेशन प्राप्त करता है, इसे इन्फेरेंस–टाइम ऑप्टिमाइजेशन के कारण जनरेट करने में काफी समय लगता है। [MTSA24] में, हम एक हाइपरनेटवर्क का उपयोग करके लेटेंट अडैप्टर नेटवर्क के पैरामीटर्स को सैंपल करना सीखकर इस मुद्दे को हल करते हैं।

अंत में, Part V में, हम अपने योगदानों को संक्षेप में प्रस्तुत करते हैं और इस शोध प्रबंध में प्रस्तुत तकनीकों और विधियों पर आधारित भविष्य के कार्यों के लिए दिशाएं प्रस्तावित करते हैं।

Contents

Acknowledgements i

Abstract iv

Abstract in Hindi vi

List of Tables xviii

List of Figures xxvii

I Prologue 1

1 Introduction 2

1.1 Generative Models . . . 3

1.1.1 Applications of Generative Models . . . 3

1.1.2 Taxonomy of Generative Models . . . 4

1.1.2.1 Regularized Autoencoders — Generative Autoencoders 5

1.1.2.2 Generative Adversarial Network . . . 7

1.1.2.3 Autoregressive Models . . . 7

1.1.2.4 Flow-Based Models . . . 8

1.1.2.5 Energy-based Models . . . 8

1.1.2.6 Diffusion Models . . . 9

1.1.3 Focus of the Thesis: Deep Latent Variable Generative Models . 9

1.1.4 Objective, Scope, and Contribution . . . 10

1.1.5 Thesis Overview . . . 14

2 Literature Review 15

2.1 Generative AE Frameworks . . . 15

2.2 Representation Learning . . . 17

2.2.1 Clustering . . . 18
2.2.2 Disentanglement . . . 18
2.2.3 Multimodal Representation Learning . . . 20
2.2.4 Class-imbalance Mitigation . . . 21
2.3 Few-shot Generation . . . 22

II Optimizing the Latent Space of RAEs for Improved Generation 24

3 Impact of Latent Dimensionality on RAE's Generation Quality 25
3.1 Introduction . . . 25
3.2 Related Work . . . 27
3.3 Effect of Latent Dimensionality . . . 29
3.3.1 Preliminaries . . . 29
3.3.2 Conditions for Good Generation . . . 30
3.4 MaskAAE (MAAE) . . . 33
3.4.1 Model Description . . . 33
3.4.2 Training MaskAAE . . . 35
3.5 Experiments and Results . . . 35
3.5.1 Synthetic Experiments . . . 36
3.5.2 Real Experiments . . . 37
3.6 Discussion and Conclusion . . . 40
3.7 Segue . . . 41

4 Impact of Prior Distribution on RAE's Generation Quality 42
4.1 Introduction . . . 42
4.2 Proposed Method . . . 43
4.2.1 Optimality of the Latent Space of Wasserstein AE . . . 43
4.2.2 Flexibly Learning Prior: FlexAE . . . 45
4.2.2.1 Smoothing the Latent Space . . . 46
4.2.2.2 Implementation . . . 47
4.2.3 Relation to Prior Works . . . 48

4.3 Experiments and Results . . . 49
4.3.1 Synthetic Experiments . . . 49
4.3.2 Real-world Datasets . . . 50
4.3.2.1 Baseline Experiments . . . 50
4.3.2.2 Effect of Latent Space Dimensionality . . . 52
4.3.2.3 Smoothness of the Latent Space . . . 53
4.3.2.4 Overfitting Vs Underfitting in FlexAE . . . 54
4.4 Conclusion . . . 55
4.5 Segue . . . 55

III Optimizing the Latent Space of RAEs for Task-Specific Representation Learning 56

5 RAE with Flexible Prior for Improved Clustering Performance 57
5.1 Introduction . . . 58
5.2 Related Work . . . 58
5.2.1 Classical Methods on Dimensionality reduction . . . 59
5.2.2 Deep-Learning based Dimensionality Reduction . . . 60
5.2.3 Bias-Variance Trade-off in RAEs . . . 61
5.3 Proposed Method . . . 61
5.3.1 Regularized Generative Autoencoders . . . 61
5.3.2 scRAE . . . 63
5.3.3 Realization of scRAE . . . 63
5.4 Experiments and Results . . . 65
5.4.1 Synthetic Experiment . . . 65
5.4.2 Real Experiment . . . 67
5.4.2.1 Datasets . . . 67
5.4.2.2 Preprocessing and Gene Selection . . . 68
5.4.2.3 Clustering . . . 69
5.4.2.4 Data Visualization . . . 72
5.4.2.5 Impact of Feature Dimension on Clustering . . . 74
5.4.2.6 Impact of Gene Selection Method . . . 74

5.4.2.7 Complexity Analysis 75
5.5 Conclusion 75
5.6 Segue 76

6 Variational Disentangled Representation Learning for Multi-modal Data 77

6.1 Introduction 77
6.2 Related Work 79
6.2.1 Variational Autoencoders 79
6.2.2 Joint Models 80
6.2.3 Disentangled Representation Learning 80
6.3 Proposed Method 81
6.3.1 Preliminaries 81
6.3.2 Variational Lower Bound 83
6.3.3 Joint Shared Continuous Posterior 84
6.3.4 Disentangling the Generative Factors 85
6.3.5 Implementation 85
6.4 Experimental Results 87
6.4.1 Image-Text 87
6.4.2 Image-Image 93
6.5 Conclusion 95
6.6 Segue 97

7 Minority Oversampling for Imbalanced Data via Class-Preserving Regularized Auto-Encoders 98

7.1 Introduction 99
7.1.1 Background and Motivation 99
7.1.2 Contributions 101
7.2 Related Work 101
7.3 Proposed Method 103
7.3.1 Class Preserving Oversampling 103
7.3.2 Regularized Autoencoders with Class Preserving Latent Space . 105
7.3.3 Implementation Details 105

7.4 Theoretical Analysis . . . 107
7.5 Experimental Results . . . 109
7.5.1 Dataset Description and Baselines . . . 109
7.5.2 Performance Metrics . . . 111
7.5.3 Model Architecture and Compute Resource . . . 112
7.5.4 Results . . . 112
7.5.5 Ablation Studies . . . 114
7.6 Conclusion, Limitations, Risks, and Broader Impact . . . 114
7.7 Segue . . . 115

IV Few-shot Generation Using DLVGMs 116

8 Few-shot Cross-domain Image Generation 117
8.1 Introduction . . . 118
8.1.1 Few Shot Image Generation . . . 118
8.1.2 Motivation and Contributions . . . 119
8.2 Related Work . . . 120
8.3 Proposed Method . . . 123
8.3.1 Overview . . . 123
8.3.2 Latent Adapter Network . . . 123
8.3.2.1 Reformulation of GAN Objective . . . 125
8.3.2.2 LAN Implementation Details . . . 127
8.3.3 Sampler over Latent Adapter Networks . . . 128
8.3.3.1 SoLAD Implementation Details . . . 130
8.4 Experiments . . . 131
8.4.1 Datasets . . . 132
8.4.2 Baselines for Benchmarking . . . 132
8.4.3 Metrics and Methodology . . . 133
8.4.4 Results . . . 134
8.4.5 Application: Image Translation . . . 135
8.5 Conclusion . . . 137

V Epilogue **138**

9 Conclusion and Future Work **139**

9.1 Conclusion . . . 139

9.2 Future Direction . . . 141

9.2.1 Identifiability of AE Based Generative Models with Adversarial Prior Matching . . . 141

9.2.2 Structured Latent Space for AE Based Generative Models with Adversarial Prior Learning . . . 141

9.2.3 Dataset Condensation using RAE . . . 142

9.2.4 Few-shot Generative Model . . . 142

Bibliography **143**

VI Appendices **181**

A Appendix to Chapter 3 **182**

A.1 Theory . . . 182

A.1.1 Justification for A2 . . . 182

A.1.2 Derivations for R1 and R2 . . . 182

A.1.3 Discussion . . . 184

A.1.4 Intuition for Lemma 1 . . . 184

A.1.5 Comparison to 2-Stage VAE . . . 184

A.2 Naive Alternatives to MAAE . . . 186

A.3 Objective, Training, and Architecture of MAAE . . . 186

A.3.1 Objective Functions of MAAE . . . 187

A.3.2 Training Algorithm . . . 189

A.3.3 Network Architectures . . . 189

A.4 Experimental Results . . . 191

A.4.1 Details of the Synthetic Dataset . . . 191

A.4.2 Log-Likelihood Score on Synthetic Dataset . . . 192

A.4.3 Analysis of Training of MAAE on MNIST . . . 192

A.4.4 Normalised Absolute Covariance Matrix: WAE vs MAAE . . . 193

A.4.5 Computing Resources and Average Runtime 195

B Appendix to Chapter 4 **196**

B.1 Theory 196

B.2 Details of Datasets 196

B.2.1 Synthetic Datasets 197

B.2.2 Real Datasets 199

B.3 Network Architectures 200

B.3.1 Architectures for Synthetic Experiments 200

B.3.2 Architectures for Experiments on Real-world Datasets 200

B.3.3 Architectures for Overfitting/Underfitting Experiment 201

B.4 Training Algorithm, Hyper-parameters, Computing Resources, and Runtime 202

B.5 Experimental Results 203

C Appendix to Chapter 5 **216**

C.1 Training Algorithm 216

C.2 Details of Synthetic Data Generation 216

D Appendix to Chapter 6 **218**

D.1 Derivation of PoE Formulation 218

E Appendix to Chapter 7 **219**

E.1 Theoretical Results 219

E.1.1 Dataset Description 223

E.1.2 Experimental Results 225

E.1.2.1 Performance of Deep Learning Methods 225

E.1.2.2 Performance of Classical Approaches 226

E.1.3 Ablation on t 226

E.1.4 Qualitative Samples 227

E.1.5 Model Architecture 228

E.1.6 Training Details 229

F Appendix to Chapter 8 **233**

F.1 Editing Using the Learnt Latent Codes 233
F.2 Generalization Across Complex Dataset 233
F.3 Details of Image Translation 233

List of Publications **238**

Biography **239**

List of Tables

1.1 Characteristics of the popular deep genera tive models. This table is adapted from [Mur23]. 5

1.2 Contrasting RAEs and GANs, the two prominent frameworks of DLVGMs. 11

3.1 FID scores for generated images from different AE-based generative models (Lower is better). 39

3.2 Average off-diagonal covariance NAC for both WAE and MAAE. m_A represents the number of unmasked latent dimensions in the trained model. It is seen that MAAE has lower NAC values, indicating lesser deviation of $Q_\phi(\boldsymbol{z})$ from $P_Z(\boldsymbol{z})$ as compared to a WAE. 40

4.1 Comparison of FID scores [HRU+17a] on real datasets. Lower is better. 51

4.2 Comparison of Precision/Recall scores [SBL+18] on real datasets. Higher is better. 51

4.3 Variation of FID w.r.t. bottleneck dimension, m. $m_b = 32$ for MNIST and Fashion and for CIFAR-10 and CELEBA $m_b = 64$. Unlike WAE [TBGS18], the generation quality of FlexAE-SR remains unaltered even when m increases. 52

4.4 Pixel/Inception Memorization Score 54

5.1 Summarized description of datasets used. 66

5.2 Comparison of the average normalized mutual information across 10 real datasets achieved by different algorithms. 68

5.3 Impact of gene selection method on clustering performance as measured by Normalized Mutual Information (NMI) score averaged over two real datasets Klein-2k [KMA+15] and Baron-8k [BVW+16]. In most cases, the clustering performance of an algorithm has improved when advanced gene selection techniques such as SCMarker [WLK+19] or M3Drop [AH18] are used. Our proposed method scRAE, outperforms current SOTA baselines for a fixed gene selection method and achieves the best performance for M3Drop [AH18]. 73

5.4 Comparison of average run time (in seconds rounded off to closest integer) of different algorithms on the smallest and the largest out of ten real datasets considered in this work. 74

5.5 Impact of feature dimension on clustering performance of scRAE as measured by NMI score for $n_z = 2$. 74

6.1 Different models with FactorVAE scores. Notations for model type are as follows. US: Unsupervised, WS: Weakly-supervised, SS: Semi-supervised, UM: Uni-modal, MM: Multi-modal 90

6.2 Different multi-modal models with Coherence scores (higher is better) for MNIST and MADBase. 91

6.3 Generation FID scores (lower is better) of different models for MNIST and MADBase. 92

6.4 Digit classification accuracy of a linear classifier using encoded representations for MNIST and MADBase. 93

7.1 Performance (Mean ± Std. Dev.) of our method on Balanced test dataset 110

7.2 Statistical significance: p-values obtained from Shaffer post-hoc tests and Bayesian Wilcoxon signed-rank tests for pairwise comparison of the proposed method with the baseline oversampling-based approaches for the ACSA metric. We have combined the results from imbalanced and long-tailed recognition test cases. 110

7.3 Density/Coverage performance comparison of the baselines with our method 110

7.4 Ablation Studies: Contribution of each component in classification performance. 111

7.5 ACSA of the proposed method at various values of bottleneck layer dimensionality, m. 111

8.1 Comparison of FID scores (↓). Lower is better. 132

8.2 Comparison of Density (↑) and Coverage (↑) scores. Higher Density and Coverage scores are better. 133

8.3 Comparison of intra-cluster pairwise LPIPS distance (↑). Higher *intra*-LPIPS is better. 133

8.4 Comparison of inference time of GenDA and SoLAD for generating 5000 images using a V100 GPU. m: minutes, h: hours, and d: days. 134

8.5 Effect of k in k-shot adaptation on FID score (↓) measuring generation quality. 134

8.6 Impact of a higher number of shots on the generation quality. 134

A.1 Encoder, Decoder, and Discriminator Architectures for Synthetic Datasets 190

A.2 Encoder, Decoder, and Discriminator Architectures for MNIST 190

A.3 Encoder, Decoder, and Discriminator Architectures for Fashion MNIST 190

A.4 Encoder, Decoder, and Discriminator Architectures for CIFAR-10 . . . 190
A.5 Encoder, Decoder, and Discriminator Architectures for CELEBA . . . 191
A.6 Effect of model capacity, m on generation LL scores of WAE and MAAE (for Synthetic$_{16}$ dataset). 192

B.1 Details of Real Datasets 199
B.2 Network Architectures for Synthetic Experiment 200
B.3 Encoder and Decoder Architectures for Real Datasets 201
B.4 Generator, Critic and Smoothing Regularization Network Architectures for Real Datasets 201
B.5 Encoder and Decoder Architectures for Overfitting/Underfitting Experiment 202
B.6 Critic Architecture for Overfitting/Underfitting Experiment 202
B.7 Generator Architectures for Overfitting/Underfitting Experiment . . . 202

E.1 Summarized Description of Benchmark Datasets 223
E.2 Class distributions of the datasets used in this work for benchmarking. 224
E.3 Class distribution for CIFAR-100 224
E.4 Class distribution for Imagenet-100 224
E.5 Performance (Mean ± Std. Dev.) of our method on Imbalanced test dataset. 226
E.6 FID of samples obtained from oversampling 226
E.7 Comparison of mAURPC-OVA 226
E.8 Performance of our method on Tabular datasets. 227
E.9 Performance of Classical Oversampling Methods on Imbalanced Test Dataset 227
E.10 Performance of Classical Oversampling Methods on Balanced Test Dataset 228
E.11 Encoder and Decoder Architectures for Image Datasets 231
E.12 Mixer and Critic Network Architectures for Image Datasets 232
E.13 Encoder and Decoder Architectures for Tabular Datasets 232
E.14 Mixer and Critic Network Architectures for Tabular Datasets 232

List of Figures

1.1 Visual summary of various deep generative frameworks. x denote observed sample, z is the latent code, and $\hat{x}$ is the model's output. For auto-regressive models, x_i denotes the generated data's i^{th} element. Auto-regressive models do not have a latent representation of data. Energy-based models may or may not have a latent representation. For diffusion and flow-based models, the latent code dimensionality is the same as the data dimension. GANs and RAEs (such as VAE, AAE, and WAE) provide a low-dimensional latent representation of visible data samples. This is why GANs and RAEs fall into the Deep Latent Variable Generative Models (DLVGMs) category. This diagram is adapted from [Mur23]. . 6

1.2 Visual summary of the contributions of this thesis. 13

3.1 FID score for a Wasserstein Auto-Encoder with varying latent dimensionality m for two synthetic datasets of 'true' latent dimensions, $n = 8$ and $n = 16$ and MNIST. It is seen that the generation quality gets worse on both sides of a certain latent dimensionality. FID scores have been scaled appropriately to bring them in the same range. 26

3.2 Depiction of the assumed data generation process. Samples drawn from a 'true' latent distribution $\widetilde{\Psi}(\widetilde{\mathbf{z}})$ are passed through a function f to obtain $\boldsymbol{x}$. 29

3.3 Block Diagram of MaskAAE. It consists of an encoder, E_ϕ, a decoder, D_θ, and a discriminator, C_κ as in AAE. A new layer called mask, μ is introduced at the end of the encoder to suppress spurious latent dimensions. The prior also gets multiplied with the same mask vector before going into the Discriminator to ensure prior matching (R2). 34

3.4 (a) and (b) shows FID score for WAE and MAAE and active dimension in a trained MAAE model with varying model capacity, m for the synthetic dataset of true latent dimensions, $n = 8$ and $n = 16$, m_A represents the number of unmasked latent dimensions in the trained model and (c) shows the same plots for MNIST dataset. 36

3.5 Behaviour of mask in MAAE with different model capacities, m for MNIST dataset. m, in figures (a), (b), and (c) are 32, 64, and 110, respectively. Dimensions active after training are m_A are 11, 13, and 11 respectively. 38

3.6 Randomly generated images of (a) MNIST, (b) Fashion MNIST, (c) CelebA, and (d) CIFAR-10 datasets. 38

4.1 Proposed Model, FlexAE: Nature first samples an n-dimensional latent code from the true latent space, $\widetilde{\mathcal{Z}}$. Next, the latent code is mapped to an n-dimensional manifold, $\mathcal{X}$ in $\mathbb{R}^d$. The observed variables are encoded using a deterministic Encoder, E_ϕ. The m-dimensional encoded representations lie in an n-dimensional manifold $\mathcal{Z}$. The decoder network, D_θ, learns an inverse projection from the learned latent space, $\mathcal{Z}$ to the dataspace, $\mathcal{X}$. The generator network, G_ψ parameterizes the learnable prior distribution that maps an isotropic Gaussian distribution in $\mathbb{R}^{m'}$ to any arbitrary prior $P_\psi(\boldsymbol{z})$ in $\mathbb{R}^m$. The critic network, C_κ, measures the distributional divergence between Q_ϕ and P_ψ. The entire network is trained in an end-to-end fashion. The forward path is shown using solid yellow arrows, and the flow of gradients due to different terms in the loss objective is also shown as color-coded dashed arrows. 45

4.2 Proposed Model, FlexAE-SR: We employ an adversarial smoothing regularization scheme on top of the proposed model FlexAE. The smoothness regulator, C_ζ measures the distributional divergence between $P^c_\theta(\boldsymbol{x}_{inp})$ and $P_d(\boldsymbol{x})$. The entire network is trained in an end-to-end fashion. The forward path is shown using solid yellow arrows, and the flow of gradients due to different terms in the optimization objective is also shown as color-coded dashed arrows. 46

4.3 Comparison of WAE (fixed prior) and FlexAE (learnable prior) on a synthetic dataset. Wasserstein distance between P_z and Q_ϕ reduces faster in the case of FlexAE compared to a fixed prior WAE, leading to a better FD. 50

4.4 Visualization of (a) true latent space; (b) latent space learned by the VAE [KW14]; (c) latent space learned by the WAE [TBGS18] with Normal prior; (d) latent space learned by the WAE [TBGS18] with GMM prior; and (e): latent space learned by the proposed FlexAE model, along with generation Fréchet Distance (FD) in each case. For multimodal data, models with multimodal prior (WAE-GMM) and FlexAE perform better. 50

4.5 (a) Visualization of reconstruction quality of FlexAE-SR model on randomly selected data from the test split of MNIST (first and second rows), Fashion-MNIST (third and fourth rows), CIFAR-10 (fifth and sixth rows), and CELEBA (seventh and eights rows). The odd rows represent the real data, and the even rows represent the reconstructed data. 64 Randomly generated samples (*i.e.*, no cherry-picking) of (b) MNIST, (c) Fashion MNIST, (d) CIFAR-10, and (e) CELEBA datasets using FlexAE model shows the quality of generation of the proposed model FlexAE-SR. . . 52

4.6 (a), (b) Each row represents linear interpolation in the latent space between two randomly selected test samples in the first and the last entry. (c) Each row presents manipulation of a particular face attribute (Big Nose, Heavy Makeup, Black Hair, Smiling, Male). The central image of each row is a true image from the train split with the attribute. (d) The first image in each row shows randomly generated samples using FlexAE-SR, and the next four entries are the nearest neighbors from training data. (e) The first entry in each row shows the same randomly generated samples as in (d) and the four nearest neighbors based on features extracted by a pre-trained Inception net. 53

4.7 Variation of reconstruction and generation FID scores on limited training datasets with varying P-GEN capacity, demonstrating overfitting and underfitting in FlexAE. Models (1-6) are presented in increasing order of capacity. 54

5.1 The novel architecture of scRAE consists of a reconstruction pipeline (the autoencoder) and a P-GEN network. The high-dimensional sparse single-cell RNA sequence is mapped to a dense low-dimensional latent representation using the encoder network, E_ϕ, and the original sequence is reconstructed using a decoder network, D_θ. The P-GEN network, which consists of a latent generator network, G_ψ, and a critic, C_κ, constrains the encoded latent space. G_ψ learns the prior flexibly based on feedback from C_κ. The learned latent representations are clustered to correspond to different cell types in the dataset. The entire network is trained in an end-to-end fashion. Solid yellow arrows in the left-right direction illustrate the forward path, and the color-coded dashed arrows in the right-left direction illustrate the backward flow of gradients due to different terms in the optimization objective. 62

5.2 Visualization of (a) true latent space; (b) latent space learned by the VAE [KW14]; (c) latent space learned by the WAE [TBGS18] with Normal prior; (d) latent space learned by the WAE [TBGS18] with Mixture of Gaussian (MoG) prior; and (e): latent space learned by the proposed scRAE model, along with NMI and ARI scores in each case. Models with multimodal prior (WAE-MoG) and learnable prior (scRAE) perform significantly better. 66

5.3 Clustering performance of scRAE is compared against 6 baseline methods. As can be seen from the figure, scRAE achieves the highest NMI score for all of the ten datasets, irrespective of the latent dimension. A higher NMI score indicates better cluster purity and performance w.r.t. ground truth labels. 69

5.4 Clustering performance of scRAE is compared against 6 deep learning-based baseline methods. As can be seen from the figure, scRAE achieves the highest AMI score for all of the ten datasets. A higher AMI score indicates better cluster purity and performance w.r.t. ground truth labels. 71

5.5 Clustering performance of scRAE is compared against 6 deep learning-based baseline methods. As can be seen from the figure, scRAE achieves the highest Homogeneity Score (HS) for all of the ten datasets. A higher score is better. 71

5.6 Clustering performance of scRAE is compared against 6 deep learning-based baseline methods. As can be seen from the figure, scRAE achieves the highest Completeness Score (CS) for all of the ten datasets. A higher score is better. 72

5.7 Two dimensional Visualization for the dataset Zheng-73k [ZTB+17] using deep models such as (a) scVI [LRC+18]; (b) SAUCIE [AVDS+19]; (c) DR-A [LMK20]; (d) scDeepCluster [TWSW19]; (e) scVAE [GVT+20] with Gaussian prior; (f) scVAE [GVT+20] with mixture of Gaussian prior; (g) scziDesk [CWZD20] and (h) scRAE (proposed). 72

5.8 Visualization of the latent space of the proposed method, scRAE when the latent dimension is 10 for the dataset Zheng-73k [ZTB+17] using (a) PCA [Jol11]; (b) t-SNE [vdMH08]; (c) UMAP [MHM18]. 73

5.9 Impact of gene selection method on clustering performance of different methods for two real-world datasets. The highest NMI score is achieved by scRAE when M3Drop [AH18] is used for gene selection for both datasets. 73

6.1 SSDMM-VAE Architecture: A block level diagram of SSDMM-VAE is presented in this figure for a pair of data modalities. We divide the latent representation into two parts - the domain-specific (private) latent representation ($\boldsymbol{z}_{p_1}$, $\boldsymbol{z}_{p_2}$) and the domain invariant (shared) latent representation, which is further divided into a discrete ($\boldsymbol{y}$) and a continuous part ($\boldsymbol{z}_s$). The discrete part encodes the label information extracted from the input samples. We approximate the continuous domain invariant latent space and the domain-specific latent space as Gaussian distribution. For each modality, the encoder network (E_{ϕ_1}, E_{ϕ_2}) outputs sufficient statistics of the private and shared continuous posterior distributions, along with an estimation of the target label. The product-of-experts technique is employed to estimate the sufficient statistics of the joint shared continuous latent space, and for the joint shared discrete latent space, we employ the average ensemble. The decoder networks (D_{θ_1}, D_{θ_2}) learn an inverse mapping from the latent space to the original data space of the respective modality. 82

6.2 Latent traversals of the SSDMM-VAE model trained on dSprites-Text with ten continuous private latent variables for image, one continuous private latent variable for text, and one 3-dimensional shared discrete latent variable. Each row corresponds to a fixed setting of the discrete latent variable, and the columns correspond to varying a single fixed dimension of the continuous latent variables. Each of the varied latent units corresponds to an interpretable generative factor, such as sprite shape (discrete factor - across rows of each sub-figure), (a) x-position, (b) y-position, (c) rotation, (d) scale, (e) White space padding (continuous factor - across columns). 88

6.3 Latent traversals of SSDMM-VAE trained on Shapes3D-Text with ten continuous private latent variables for image, one continuous private latent variable for text, and one 4-dimensional shared discrete latent variable. Each row corresponds to a fixed setting of the discrete latent variable, and the columns correspond to varying a single fixed dimension of the continuous latent variables. Each of the varied latent units corresponds to an interpretable generative factor, such as object type (discrete factor - across rows of each sub-figure), (a) floor color, (b) wall color, (c) object color, (d) object size, (e) White space padding (continuous factor - across columns). 89

6.4 t-SNE visualization of the learned latent spaces in the Shapes3D-Text experiment. For the image domain, the embeddings corresponding to four different classes are separated in the latent space, while the data points from the same class are closely clustered. For the text domain, the private latent space has only one degree of freedom (the number of white spaces appended or prepended); therefore, the encoded data sticks together. This explains the almost linear curves in the t-SNE plot for the text domain. 89

6.5 Latent traversals of the SSDMM-VAE model trained on MNIST-MADBase with ten continuous shared latent variables and one 10-dimensional shared discrete latent variable. Each row corresponds to a fixed setting of the discrete latent variable, and the columns correspond to varying a single fixed dimension of the continuous latent variables. Each of the varied latent units corresponds to an interpretable generative factor, such as digit type (discrete factor - across rows of each sub-figure), tilt (a, b), and width/size (c, d) (continuous factor - across columns). 92

6.6 (a, b) presents the reconstructed images (second row) of MNIST and MADBase, respectively, when real test images (first row) are fed to the model. (c, d) presents conditional cross generation (second row) for MNIST and MADBase, respectively, when real test images (first row) of MADBase and MNIST are fed to the model. (e) presents reconstructed MNIST (third row) and reconstructed MADBase (fourth row) when real test images from MNIST (first row) and MADBase (second row) are fed to the model. (f, g) presents randomly generated images of MNIST and MADBase, respectively, when random prior samples are fed to the trained decoders. Interestingly, the model exhibits coherent generation, *i.e.* the model generates the same digit from both datasets for a given prior. 93

6.7 Visualization of the joint latent space using t-SNE in the MNIST-MADBase experiment. As can be seen from the plot, the data points corresponding to different digits are separated because of disentanglement, while the digits of the same class are closely clustered. 94

7.1 Architecture of the proposed methodology. It is a regularized autoencoder, where the latent space is regularized using a linear classifier to facilitate distance metric-free class preserving oversampling of the minority classes. The decoder network maximizes the conditional data likelihood to avoid degeneracy in the latent space. 104

7.2 t-SNE visualization of the latent space. 113

8.1 (a) Emoji [HRZ17], (b) Amedeo Modigliani's Art [YNS19], (c) Fernand Leger's Art [YNS19], (d) Moise Kisling's Art [YNS19], (e) Sketches [WT09]. The left image in each pair is the original image, and the right image is reconstructed using its embedding in the extended intermediate latent space of StyleGAN2 [KLA+20] trained on the FFHQ dataset. The latent space accommodates a wide array of data. 119

8.2 Illustration of the proposed methods: (a) GenDA and (b) SoLAD. . . . 124

8.3 Source domain: FFHQ. Each row corresponds to one of the following target domains — Babies, Sunglasses, Sketches, Emoji, Paintings of Amedeo Modigliani. The first column shows 10-shot training examples from the target domains. The second, third, fourth, and fifth columns display ten images generated by CDC [OLL+21], RSSA [XLW+22], GenDA [MTSA23], and SoLAD, respectively. 130

8.4 Source domain: LSUN Church [YZS+15]. Target domains: Haunted house [OLL+21] and Van Gogh's house paintings [OLL+21]. The first column displays 10-shot training examples from various target domains, while the second and third columns present ten images generated using CDC [OLL+21] and RSSA [XLW+22], respectively. The fourth and fifth columns showcase images generated using our proposed methods, GenDA and SoLAD. . . 131

8.5 10-shot adaptation from cars to wrecked cars. 131

8.6 FFHQ → Babies: 1-shot adaptation. 136

8.7 FFHQ → Babies: 5-shot adaptation. 136

8.8 Smooth interpolation in the latent space depicts the smoothness in the transformation. 136

8.9 Transferring source domain (FFHQ) image (leftmost image in each subfigure) to target domains Babies (center image) and Sketches (rightmost image). 136

A.1 For $n = 2$, $\epsilon = 4$ radius of each ball, $r = \frac{\sqrt{(2)}}{8}$. 185

A.2 Architecture for Synthetic Dataset generation. 191

A.3 (a) Reconstruction loss, (b) Wasserstein distance, and (c) FID plot w.r.t. training iterations for MAAE model on MNIST for model capacity $m = 110$. 192

A.4 Co-variance Matrix of (a) WAE (b) MAAE latent representation for MNIST dataset. 193

A.5 Co-variance Matrix of (a) WAE (b) MAAE latent representation for Fashion MNIST dataset. 193

A.6 Co-variance Matrix of (a) WAE (b) MAAE latent representation for CelebA dataset. 194

A.7 Co-variance Matrix of (a) WAE (b) MAAE latent representation for CIFAR-10 dataset. 194

B.1 225 randomly generated MNIST samples using FlexAE-SR. 205

B.2 225 randomly generated Fashion MNIST samples using FlexAE-SR. . . 206

B.3 225 randomly generated CIFAR-10 samples using FlexAE-SR. 207

B.4 225 randomly generated CELEBA samples using FlexAE-SR. 208

B.5 Interpolations in the latent space of FlexAE-SR on (a) MNIST, (b) Fashion MNIST, (c) CIFAR10, and (d) CelebA. Each row's first and last entry is the reconstructed image corresponding to two real samples. The remaining images are obtained by decoding different convex combinations of the latent codes corresponding to the two real images. The central image in each row is decoded from the average latent code of the two real image latent codes. The interpolated images are mostly realistic, indicating the smoothness of the FlexAE-SR's latent space. 209

B.6 Interpolations in the latent space of FlexAE-SR on CelebA. Each row in (a) and (b) presents manipulation of the attribute "Big Nose". The central image of each grid in (a) and (b) is a true image without the attribute. The central image of each grid in (c) and (d) is a true image with the attribute. Interpolation direction: Top to Bottom, Left to right, like reading an English textbook. 209

B.7 Interpolations in the latent space of FlexAE-SR on CelebA. Each row in (a) and (b) presents manipulation of the attribute "Heavy Makeup". The central image of each grid in (a) and (b) is a true image without the attribute. The central image of each grid in (c) and (d) is a true image with the attribute. Interpolation direction: Top to Bottom, Left to right, like reading an English textbook. 210

B.8 Interpolations in the latent space of FlexAE-SR on CelebA. Each row in (a) and (b) presents manipulation of the attribute "Black Hair". The central image of each grid in (a) and (b) is a true image without the attribute. The central image of each grid in (c) and (d) is a true image with the attribute. Interpolation direction: Top to Bottom, Left to right, like reading an English textbook. 210

B.9 Interpolations in the latent space of FlexAE-SR on CelebA. Each row in (a) and (b) presents manipulation of the attribute "Smiling". The central image of each grid in (a) and (b) is a true image without the attribute. The central image of each grid in (c) and (d) is a true image with the attribute. Interpolation direction: Top to Bottom, Left to right, like reading an English textbook. 210

B.10 Interpolations in the latent space of FlexAE-SR on CelebA. Each row in (a) and (b) presents manipulation of the attribute "Male". The central image of each grid in (a) and (b) is a true image without the attribute. The central image of each grid in (c) and (d) is a true image with the attribute. Interpolation direction: Top to Bottom, Left to right, like reading an English textbook. 211

B.11 The first entry in each row represents a randomly generated CIFAR-10 object using FlexAE-SR. The remaining entries in each row represent 14 nearest neighbors (in terms of Euclidean distance) from the train split of the CIFAR-10 dataset based on pixel values. It can be seen that the images generated using FlexAE are very different from the training examples. This confirms that the state-of-the-art FID score and precision-recall score obtained using FlexAE are not due to mere overfitting on the training split. 212

B.12 The first entry in each row represents a randomly generated CIFAR-10 object using FlexAE-SR. The remaining entries in each row represent 14 nearest neighbors (in terms of Euclidean distance) from the train split of the CIFAR-10 dataset based on features extracted using a pre-trained Inception net. It can be seen that the images generated using FlexAE are very different from the training examples. This confirms that the state-of-the-art FID score and precision-recall score obtained using FlexAE are not due to mere overfitting on the training split. 213

B.13 The first entry in each row represents a randomly generated face using FlexAE-SR. The remaining entries in each row represent 14 nearest neighbors (in terms of Euclidean distance) from the train split of the CELEBA dataset based on pixel values. It can be seen that the images generated using FlexAE are very different from the training examples. This confirms that the state-of-the-art FID score and precision-recall score obtained using FlexAE are not due to mere overfitting on the training split. 214

B.14 The first entry in each row represents a randomly generated face using FlexAE-SR. The remaining entries in each row represent 14 nearest neighbors (in terms of Euclidean distance) from the train split of the CELEBA dataset based on features extracted using a pre-trained Inception net. It can be seen that the images generated using FlexAE are very different from the training examples. This confirms that the state-of-the-art FID score and precision-recall score obtained using FlexAE are not due to mere overfitting on the training split. 215

E.1 Ablation: Number of mixing components vs. Classification performance (ACSA) for different datasets. 229

E.2 MNIST minority class images, with rows corresponding to digit classes. All the images are randomly chosen without any cherry-picking. 229

E.3 FashionMNIST [XRV17] minority classes: trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot. Images are randomly chosen. 230

E.4 SVHN [NWC+11] minority class images, with rows corresponding to digit classes. All the images are randomly chosen without any cherry-picking. 230

E.5 CIFAR-10 [Kri09] minority class images: automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Images are randomly chosen without any cherry-picking. 231

E.6 CelebA [LLWT15a] minority class images: brown hair, blond hair, grey hair, bald. All the images are randomly chosen without any cherry-picking. 231

F.1 Editing generated baby images. The images in the middle row are generated using our method, GenDA. The top row and bottom row present edited images. 234

F.2 Editing generated sketches. The images in the middle row are generated using our method, GenDA. The top row and bottom row present edited images. 235

F.3 10 shot adaptation to floor plans from maps. The resolution of images is 1024×1024. 237