ANOMALY DETECTION VIA SCORE MATCHING

Ahsan Mahmood

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2024

Approved by:

Martin Styner

Junier Oliva

Heather Hazlett

Danielle Szafir

Guorong Wu

ABSTRACT

Ahsan Mahmood: Anomaly Detection in Medical Imaging via Score Matching
(Under the direction of Martin Styner)

Anomaly detection is a critical task across various domains, including healthcare, manufacturing, and finance. While deep learning has shown tremendous success in various supervised tasks, unsupervised anomaly detection remains an open challenge. In this dissertation, I introduce Multiscale Score Matching Analysis (MSMA), a novel methodology that enables unsupervised anomaly detection by analyzing score functions, which represent the gradients of the log-probability density with respect to the data.

MSMA estimates score functions at multiple noise scales, leveraging the intuition that anomalies exhibit distinct characteristics at different scales. The resulting multiscale score vectors are used to learn the distribution of inlier data, enabling the detection of out-of-distribution samples. To extend MSMA to mixed continuous and categorical data, I propose a novel score matching objective, Gumbel Noise Score Matching (GNSM), for learning scores of categorical variables. Furthermore, I introduce Spatial-MSMA, an extension that incorporates spatial information to localize anomalies within images. Spatial-MSMA is successfully applied to 3D brain MRI data, demonstrating its capability in detecting lesions. Finally, I present a case study, illustrating how MSMA can be employed as a hypothesis-generating tool. I will demonstrate how MSMA provides insights into the structural brain differences associated with Down Syndrome.

Overall, my research introduces a principled and flexible framework for unsupervised anomaly detection, localization, and data exploration. While this work focused on medical imaging, MSMA has potential applications in a variety of domains, and can easily be extended to other modalities such as audio and tabular data.

To my family, friends, 146

ACKNOWLEDGEMENTS

I extend my deepest gratitude to my advisor, Dr. Martin Styner, for his unwavering support and boundless optimism throughout my PhD journey. Martin provided me with the freedom to explore, learn, and grow – a gift for which I will always be grateful. My sincere appreciation goes to my committee members: Junier, for his guidance from the very beginning and for challenging me to pursue my ideas with rigor; Danielle and Heather, for their invaluable time, extensive feedback, and mentorship in navigating my research endeavors; and Guorong, for inspiring me to delve deeper into the field of medical imaging.

I am also greatly thankful for my family and friends, whose continuous support and patience during the challenges of research have been instrumental in my progress. Lastly, I would like to thank my partner in crime, my loaf. Her love, companionship, and cooking have been the bedrock of my success in this journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ASD	Autism Spectrum Disorder
DS	Down Syndrome
DSM	Denoising Score Matching
EBM	Energy Based Model
GAN	Generative Adversarial Network
GMM	Gaussian Mixture Model
GNSM	Gumbel Noise Score Matching
MSMA	Multiscale Score Matching Analysis
MRI	Magnetic Resonance Imaging
NCSN	Noise Conditioned Score Network
OOD	Out-of-Distribution Detection
SDE	Stochastic Differential Equation
SOM	Self-Organizing Map
VESDE	Variance Exploding Stochastic Differential Equation

CHAPTER 1: INTRODUCTION

1.1 Motivation

Anomaly detection is an active area of research with many potential benefits for healthcare, manufacturing, and financial industries. While the specific realization of an anomaly will vary across applications, the term usually refers to samples that are rare and markedly different from the average.

In healthcare, anomalies exhibit themselves as medical pathologies such as lesions and tumors. In manufacturing industries, anomalies can take the form of defects on the products being fabricated. In the financial sector, examples of common anomalies include fraudulent transactions, misuse of systems, and suspicious activity within computer networks. Figure 1.1 gives pictorial examples of the types of anomalies that may be observed across industries.

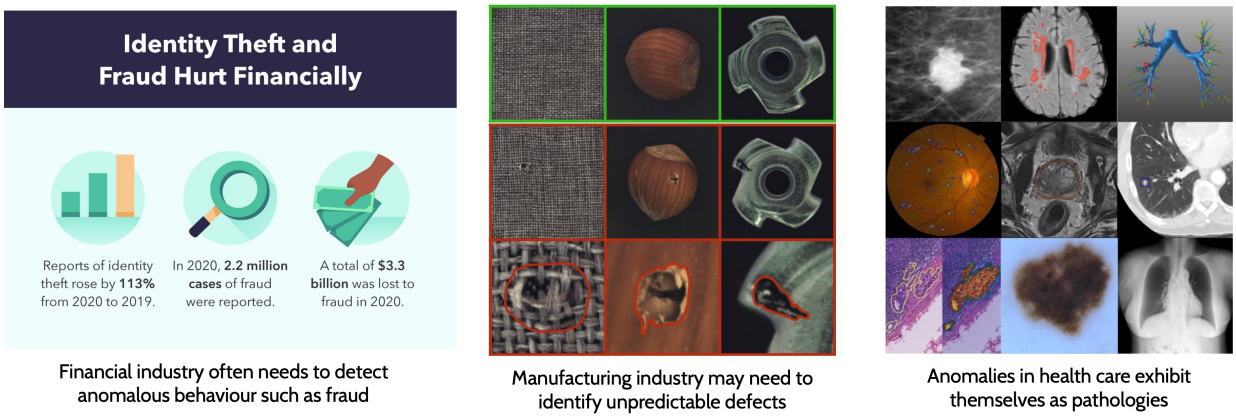


Figure 1.1: Examples of anomalies across different industries. In all cases, anomalies are unexpected and undefined deviations from the norm, making them difficult to predict a priori.

Defining the term Anomaly

As of yet, there is no standard definition of the term anomaly. For the purpose of my research, I will opt for a fairly loose definition of the term: any data point that does not belong to the typical set. That is, an anomaly is a sample with low probability *mass*, with respect to the distribution described by the training data.

Note that the diversity of my training samples will always be a limiting factor for the anomaly detection performance. It is very possible to encounter samples that human experts will categorize as typical but may seem out-of-distribution to a model trained on limited data. This is an inherent limitation of all learning techniques, and one should be mindful of such limitations.

Detecting Anomalies in Medical Imaging

This thesis will approach the anomaly detection problem from the lens of medical imaging. Identifying anomalies in images such as MRI scans can assist clinicians in detecting conditions earlier and initiating treatment more rapidly. Further, automating the detection of pathologies can help reduce error rates. For instance, it is estimated that radiologists can miss a relevant pathology in 5-10% of scans [Bruno, Walker and Abujudeh, 2015]. Of these medical errors, 60-80% could be explained by "perceptual errors" i.e., a finding is present on the image but is missed. Recent breakthroughs in deep learning have demonstrated tremendous abilities in detecting pathologies such as tumors and lesions [Kim et al., 2019; Lee et al., 2017].

There is much enthusiasm in productionizing these deep learning models in the clinical setting to augment the diagnostic abilities of medical practitioners and reducing error rates. Currently, some companies have deployed AI models for clinical use-cases [Chamberlin et al., 2023], albeit with limited success. Studies have shown that these models tend to trade-off false positives for higher sensitivity of detection [Niehoff et al., 2023]. As such, they are more reliable for their negative prediction properties i.e., clinicians may use the model to increase trust in their own assessments if the model also does not discover any pathologies. Furthermore, these models are trained in a supervised manner and can only detect the specific abnormalities included in training. For instance the 'AI-Rad' model of [Chamberlin et al., 2023] can only detect lung nodules and arterial abnormalities. This model would not be able to detect cancerous cells that appear in the lungs even if they are observable in the image, as it was never trained to do so. This leaves much room for research into developing general purpose anomaly detectors that may detect a wide range of anomalies, with low false-detection rates.

Note that while this thesis will ultimately build an anomaly detector to be used for medical imaging, the methodology presented in this document is general-purpose. One can easily adapt the proposed deep learning model to any application as well as most data modalities. [1]

[1]With the exception of text as it requires nontrivial modifications to the underlying training objective

The Case for Unsupervised Models

Most existing pathology detectors are trained using labeled data (supervised training). Collecting labels for anomalous data is time-consuming, cost-prohibitive, and requires multiple expert human annotators. Furthermore, anomalies are unpredictable by definition. In medical imaging, anomalies can exhibit themselves in various forms such as physical pathologies (e.g. tumors and lesions), image acquisition artifacts (e.g. noise caused by motion during a scan), or anatomical deviations caused by non-pathological sources such as age or neural divergence. This makes it impractical to predetermine the entire set of possible anomalies and collect data apriori.

In response to the dearth of labeled data, progress has been made towards developing unsupervised anomaly detectors [Bergmann et al., 2020; Baur et al., 2019; Ruff et al., 2021]. These models are trained on unlabeled images representative of a normal/typical population. However, meta-analyses have revealed that there is no clear winner in this space [Baur et al., 2021; Ruff et al., 2021]. Models behave inconsistently across different pathologies, and there is often no obvious correlation between model complexity and detection performance. Another major limitation of many existing unsupervised anomaly detection models is that their training objectives do not directly optimize for the task of identifying anomalies, but rather optimize for auxiliary objectives that are only tangentially related to anomaly detection. My work builds on the score matching objective, which approximates the gradient vector pointing in the direction of increasing likelihood for a datum. An anomalous sample (with respect to the inlying distribution) will have larger gradients as it is further away from the typical region. Thus, aided by score matching, my methodology directly captures the characteristics that make a sample atypical.

Autoencoders Do Not Make Good Anomaly Detectors

An overwhelming majority of anomaly models are based on some flavor of the autoencoding objective [Baur et al., 2019, 2021; Tschuchnig and Gadermayr, 2022; Kascenas et al., 2023] i.e., the model is tasked to reproduce the input image. The assumption is that, having only seen typical samples during training, the autoencoder will fail to reconstruct anomalies. Consequently, one can use reconstruction errors as anomaly scores. However, one can foresee a paradox underlying this assumption. As the performance of the autoencoder increases, its reconstruction capabilities improve, and it is able to generalize to unseen data. This implies that a performant autoencoder will be a poor anomaly detector, which is exactly what we observe

in practice. Well-optimized autoencoders learn to reconstruct anomalies with ease. For instance, consider Figure 1.2 taken from [Baur et al., 2019]. The model in the middle column is superior at reconstruction compared to the model in the right column. However, the performant model also reconstructs anomalies, making it worse as an anomaly detector. Researchers thus try to limit performance capabilities by including regularization schemes into the objective so that the autoencoder is performant *only* on inlier data and fails to generalize to other datasets. In my opinion, these constraints are ad-hoc solutions, and the objective itself is inappropriate for the task. We need the training objective to be more closely aligned to learning typicality.

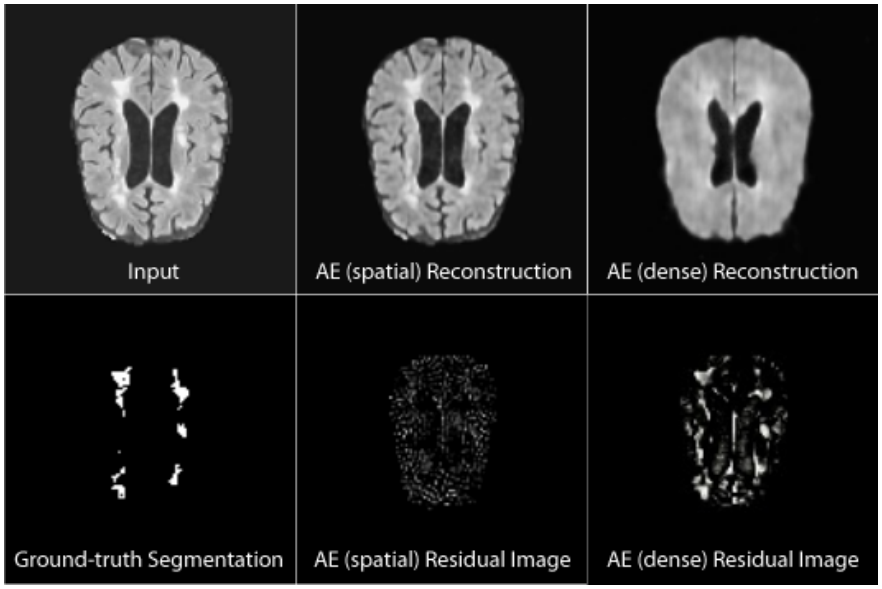


Figure 1.2: More performant autoencoders (middle column) will reconstruct anomalies with high accuracy, making them less appropriate for anomaly detection. However, less performant autoencoders (right column) will result in more false positives. How are practitioners supposed to determine the sweet spot? I argue that the autoencoding objective is inappropriate for anomaly detection from the start.

The Curious Failure of Likelihood Models

Likelihood models are a natural candidate for anomaly detection. In fact, most definitions of the term "anomaly" invoke concepts related to probability. In the past decade, there have been many exciting advances in deep likelihood models such as PixelCNN, Glow, and an explosion of an entire class of so-called normalizing flow models.

These methods will estimate the probability densities of the samples with respect to a training distribution. A high probability density is an indication that the data point belongs to a set of samples that are likely to occur. Conversely, out-of-distribution (OOD) samples will tend to reside in low-density probability regions.

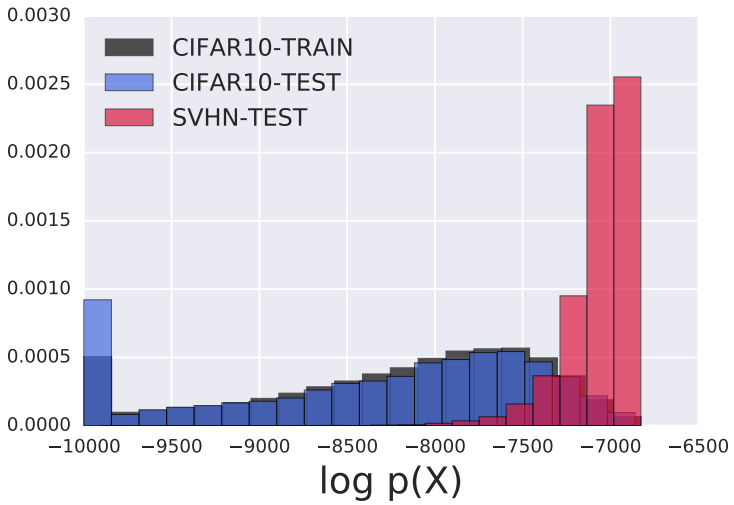


Figure 1.3: Log-likelihoods reported by a Glow model trained on CIFAR-10 (natural images). Note how it assigns *higher* likelihoods to the out-of-distribution SVHN (street view housing number) images. This behavior is paradoxical and yet to be fully understood.

So why not use deep likelihood models for anomaly detection? While these models often demonstrate exceptional generative capabilities, they frequently fail at detecting outlying samples. Paradoxically, they often assign *higher* likelihood to OOD samples. This curious failure of OOD detection has been extensively reported [Nalisnick et al., 2019; Kirichenko, Izmailov and Wilson, 2020*a*; Nalisnick et al., 2020] but has yet to be fully understood.

Some candidate explanations point to the overfitting of models to low-level features in the images. Of course, this argument only holds for vision models, whereas to my knowledge, the failure of deep likelihood models is agnostic to the data modality.

Another argument questions the typical set hypothesis. The typical set of a probability distribution is the set whose elements have an information content sufficiently close to that of the expected information [Shannon, 1948]. Researchers prescribing to this school-of-thought argue that high dimensional probability density functions may not correspond with our intuitions of probability. In high-dimensional Gaussians, the typical set resides in a thin shell at a specific distance from the center. Thus, a sample may belong to a typical set and yet need not lie in a high-density region. This is the so-called “Gaussian Soap Bubble” phenomenon and is rigorously proved in [Vershynin, 2018]

1.2 Score-Matching for Predicting Anomalies

In this thesis, I will develop a case for a particular methodology, called denoising score matching, as a promising solution to unsupervised anomaly detection. A *score* is defined as the gradient of the log-probability-density with respect to the data. Conceptually, a score is a vector field that points in the direction where the likelihood increases the most. Score matching is a technique that is utilized to estimate this vector. Intuitively, score matching estimates how far a given data point is from the data distribution. In the context of anomaly detection, score matching offers a powerful, yet under-explored, tool to discern pathological patterns from normal variations.

I posit that a *multiscale* analysis of score estimates can effectively identify anomalies stemming from multiple underlying factors. In this research, "multiscale" refers to the multiple levels of noise that are used to perturb the data. Intuitively, higher noise levels obscure local information, forcing the model to learn more global patterns. Therefore, by utilizing multiple noise levels during training, both global and local contextual features can be captured. This multiscale capacity is particularly beneficial for anomaly detection in medical images, where anomalies may manifest as localized textures or as large-scale structures.

Note that the idea of using gradients of the log density as a means of outlier detection has been done in the past, most notably by [Grathwohl et al., 2020] and [Zhai et al., 2016]. However, a bit surprisingly, neither work explicitly connected their ideas to score matching. [Grathwohl et al., 2020] described the norm of the gradient of the log-density as "Approximate Mass", obtained from an Energy-Based Model (EBM). The authors even noted how the Approximate Mass predictor greatly outperformed likelihoods for the OOD task but did not explore the idea further. Their work can be considered as special-case of my proposed method i.e., as a single-scale score estimator trained with sufficiently low noise would be estimating the gradient norms of the true log density. I will note that my technique significantly outperforms Approximate Mass, as reported in [Mahmood, Oliva and Styner, 2021]. [Zhai et al., 2016] chanced upon gradient norms while exploring energy-based models for outlier detection. They correctly identified how reconstruction errors obtained via an energy-based model would correspond to the gradient of the log density. However, they did not see an improvement over simply using the energy scores. I posit that they too would have seen improvements by adopting a multi-scale approach as it would capture multiple views of the energy function, and allow for an ensemble of gradient norms.

To my knowledge, my work has been the first to explore anomaly detection from the perspective of score matching. This thesis is also the first to empirically underscore the effectiveness of multiple noise scales for this task.

1.3 Thesis Statement

Score norms, computed by neural score estimators, produce a space in which out-of-distribution or anomalous samples may be separated from the inlying, typical samples.

CHAPTER 2: BACKGROUND AND PRIOR WORK

2.1 Score Matching

A score is defined as the gradient of the log probability density, with respect to the data. Conceptually, a score is a vector field that points in the direction where the log density grows the most. [Hyvärinen, 2005] introduced score matching as a means of computing the parameters of an unnormalized probability density model. The authors proved the remarkable property that learning the score involves the gradient of the score function itself as shown in Equation 2.1. Following the naming scheme used in [Vincent, 2011], this objective is called Implicit Score Matching.

$$J_{ISM}(\theta) = \mathbb{E}_{x\sim p(x)} \frac{1}{2} \left[||s_\theta(x) - \nabla_x \log p(x)||^2 \right] \tag{2.1}$$

$$= \mathbb{E}_{x\sim p(x)} \left[||s_\theta(x)||^2 + \sum_{i=1}^{d} \partial x_i s(x_i) \right] \tag{2.2}$$

Denoising Score Matching

[Vincent, 2011] formalized a connection between denoising autoencoders and score matching, and proposed the denoising score matching (DSM) objective. The authors noted how [Hyvärinen, 2005] had suggested the possibility of an alternate score matching objective; one that was based on regressing against the data gradients of a Parzen window density estimator. This so-called Explicit Score Matching objective is shown in Equation 2.3.

$$J_{ESM}(\theta) = \mathbb{E}_{x\sim q_\sigma(x)} \frac{1}{2} \left[||s_\theta(x) - \nabla_x \log q_\sigma(x)||^2 \right] \tag{2.3}$$

[Vincent, 2011] proved that under certain regularity conditions,[1] the Parzen window based objective is equivalent to the original objective proposed by [Hyvärinen, 2005] in Equation 2.1

Taking it one step further, assume the Parzen density estimate is chosen to estimate the joint density of clean and *corrupted* samples $(x, \tilde{x})$ i.e. $q_\sigma(x, \tilde{x}) = q_\sigma(x|\tilde{x})p(x)$. Thus, the DSM objective is simply:

$$J_{DSM}(\theta) = \mathbb{E}_{\tilde{x}\sim q_\sigma(x,\tilde{x})}\frac{1}{2}\left[||s_\theta(\tilde{x}) - \nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|x)||^2\right] \tag{2.4}$$

DSM mitigates the need for computing second order gradients as is the case for Equation 2.1. Furthermore, if q_σ is set as the Gaussian kernel $\mathcal{N}(\tilde{x}|x,\ \sigma^2 I)$, then $\nabla_x \log q_\sigma(\tilde{x}) = \frac{(x-\tilde{x})}{\sigma}$. One can now see the connection between score matching and the denoising autoencoder objective (when using a Gaussian kernel). The score model is effectively being trained to estimate the *noise* that was added to the image.

It should be emphasized that while [Vincent, 2011] and many subsequent works [Song and Ermon, 2019, 2020; Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020], use the Gaussian distribution in DSM, the proof for the validity of the objective by [Vincent, 2011] holds for *any* differentiable noise distribution. I will make use of this fact when deriving a score matching objective for categorical data in Chapter 4.

Noise Conditioned Score Matching

[Song and Ermon, 2019] extended the DSM objective in Equation 2.4 to incorporate multiple scales σ, and train a so-called Noise Conditioned Score Network (NCSN). The authors further outlined an iterative sampling algorithm, dubbed annealed Langevin dynamics, enabling the score network to be employed as a deep generative model. Let $\{\sigma_i\}_{i=1}^L$ be a positive geometric sequence that satisfies $\frac{\sigma_1}{\sigma_2} = ... = \frac{\sigma_{L-1}}{\sigma_L} > 1$. NCSN is a conditional network, $s_\theta(x, \sigma)$, trained to jointly estimate scores for various levels of noise σ_i such that $\forall\sigma \in \{\sigma_i\}_{i=1}^L : s_\theta(x, \sigma) \approx \nabla_x \log q_\sigma(x)$. In [Song and Ermon, 2019], the conditioning information is explicitly provided via a one-hot vector denoting the noise level used to perturb the data. The network is then trained via a modified denoising score matching objective as shown in Equation 2.5.

$$J_{NCSN}(\theta) = \frac{1}{L}\sum_{i=1}^{L}\lambda(\sigma_i)\left[\frac{1}{2}\,\mathbb{E}_{\tilde{x}\sim q_{\sigma_i}(\tilde{x}|x)p_{\text{data}}(x)}\left[\left|\left|s_\theta(\tilde{x}, \sigma_i) + (\frac{\tilde{x} - x}{\sigma_i^2})\right|\right|_2^2\right]\right] \tag{2.5}$$

[1]For any window size $\sigma > 0$, the kernel q_σ is differentiable, converges to 0 at infinity, and has a finite gradient norm

Connecting Score Matching to Diffusion Models

In a follow-up work, [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020] described a connection between noise-conditioned score matching and diffusion models [Sohl-Dickstein et al., 2015]. The connection is presented under the lens of generative modeling, and provides a framework which unifies Markov-based and continuous-time diffusion models. The key insight is that successive perturbation of a data point using a scale-dependent noise distribution (as done in NCSNs), follows a Stochastic Differential Equation (SDE). Thus, the 'forward' diffusion process can be modeled as an SDE. If one has access to the scores at each time point, it is possible to construct a reverse-time SDE as proved by [Anderson, 1982]. These reverse-SDEs can be numerically solved using any differential equation solver, only requiring access to the score function. The authors were able to use this formulation to generate images that surpassed the state-of-the-art generative models. This work was seminal in the development of future diffusion models.

Continuous-Time Score Matching

More relevant to this research, [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020] enabled a continuous relaxation to the discretized nature of noise conditioned score matching. Defining the noising process as a continuous forward SDE alleviates the need to predetermine the number of noise scales (as was the case for NCSN models). For generative models, this translates to faster sampling as one can control the number of gradient steps to take. For my research, it gives me the ability to observe different noise scales at test time while maintaining the advantages of using multiple noise scales during training. Namely, it forces the model to learn smoother transitions between noise scales, which helps test time generalizability.

2.2 Anomaly Detection

This thesis will specifically focus on *unsupervised* anomaly detection. A myriad of methods have been proposed to tackle this problem [Pang et al., 2021; Ruff et al., 2021], with varying success [Han et al., 2022]. The following sections provide a summary of some methodologies relevant to this research.

Probability-based

Perhaps the most common paradigm for detecting anomalous samples is to use the probability density estimate. The principal assumption here is that anomalies are located in low-density regions in the probability

space. The objective is to learn the density function representative of the typical (training) data. A trained model is used to assign probabilities to test samples, with low probabilities signifying anomalies.

Mixture models estimate the parameters for a mixture of probability distributions, which are then used in conjunction to estimate likelihood of the data. Gaussian Mixture Models (GMMs) [Reynolds et al., 2009] are one of the most popular examples of a mixture model, where the probability is estimated from the weighted sum of Normal distributions. [Zong et al., 2018] introduced Deep Autoencoding Gaussian Mixture Models (DAGMM), an unsupervised anomaly detection model based on GMMs. They combined the reconstruction objective of autoencoders with the likelihood objective of mixture models. The resulting network estimates the likelihood of the low-dimensional embedding of a data point, as obtained by the encoder network of the autoencoder.

Certain deep generative models also allow for the estimation of probability densities. *Autoregressive* models such as Glow and PixelCNN (and their earlier counterparts, NADE and MADE) allow direct access to density estimates. However, [Nalisnick et al., 2019] showed that deep likelihood models often fail to detect out-of-distribution samples.

Normalizing flow models are a flexible class of generative models. They utilize invertible transformations to project the data into the space of a predefined base distribution, such as a Standard Gaussian. Examples of deep normalizing flows are models such as RealNVP [Dinh, Sohl-Dickstein and Bengio, 2017] or Neural Spline Flows [Durkan et al., 2019], which are typically trained via the maximum likelihood objective. However, these models fail to detect outlying samples much like deep autoregressive models. The work by [Kirichenko, Izmailov and Wilson, 2020*b*] investigated the failure cases of flow models. The authors analyses suggest that flow models encode the visual appearance directly, without learning any semantic content. The anomaly detection performance of flow-based models can improve if they are trained on high-level semantic representations (e.g. from a pretrained neural network) rather than the raw images themselves.

Variational Autoencoders [Kingma and Welling, 2013] are another branch of generative models that utilize a latent variable factorization to approximate the density. While their behaviour on out-of-distribution samples has not been systematically studied, as with deep likelihood models, they empirically show middling performance as anomaly detectors [Baur et al., 2021; Kascenas, Sanchez, Schrempf, Wang, Clackett, Mikhael, Voisey, Goatman, Weir, Pugeault, Tsaftaris and O'Neil, 2023].

Lastly, many *non-deep learning* approaches are still being developed for anomaly detection. COPOD [Li et al., 2020] is a parameter-free, highly interpretable outlier detection algorithm based on empirical copula

models. DoSE [Morningstar et al., 2021] uses Kernel Density Estimates for learning the distribution of an ensemble of data statistics. The sum of the resulting log-probabilities is used as an outlier metric. ECOD [Li et al., 2022] uses a different notion of density and estimates the cumulative distribution function (CDF) for each feature in the data. It then uses the tail probabilities from each learned CDF to designate samples as anomalous.

Reconstruction based

Reconstruction models such as autoencoders train a model to reproduce the input image. These models first encode the data into a low-dimensional "bottleneck" embedding, followed by a decoding step. During inference, it is assumed that when given an anomalous sample, the model will output an anomaly-free reconstruction. One can then use the reconstruction error as the anomaly metric. Examples of such methods include vanilla autoencoders trained with a mean-squared objective [Luo et al., 2023], denoising autoencoders that are trained to reconstruct from a noisy input [Kascenas, Pugeault and O'Neil, 2022], and generative autoencoders to 'restore' the input [Graham et al., 2023; Wyatt et al., 2022]. The latter differ from the former in that they utilize generative models such as diffusion models or VAEs to iteratively remove the anomalies from the input image (hence restoring it to an anomaly-free reconstruction). A known drawback of all reconstruction models is the lack of specificity in their detection. As no reconstruction is pixel-perfect (especially in terms of image intensities), the output error maps have significant false-positives [Baur et al., 2021]. Another drawback of autoencoders is that as their reconstruction abilities improve, their anomaly detection capabilities decrease as the models are better at reconstructing the anomalies.

One-Class Objectives

Despite the name, these methods do not utilize labeled samples. Instead, they try to learn the boundary that best encapsulates the training data; treating all points outside the boundary as outliers. For example, One-Class Support Vector Machines (OC-SVMs) [Chen, Zhou and Huang, 2001] try to find the tightest hyperplane around the dataset, while Deep Support Vector Data Descriptors (DSVDD [Ruff et al., 2018] will compute the minimal hypersphere that encloses the data. Both methods assume that inliers will fall under the margins, and consequently use the distance to the margin boundaries as a score of outlierness.

2.2.1 Out-of-Distribution Detection

A considerable amount of effort has gone into detecting out-of-distribution (OOD) data samples, especially under the lens of classification models. Thus, there is a class of methods that augment existing classifiers to better detect OOD input samples, rather than training an unsupervised detector from scratch. Perhaps the seminal work in this area was done by Hendrycks and Gimpel [2017]. The authors were the first to significantly highlight the domain-shift discrepancy exhibited by a range of deep learning models, and established an experimental test-bed that has served as a template for subsequent OOD work. They posited that OOD samples are likely to be assigned low probabilities and purported the thresholding of softmax probabilities from well-trained classifiers to detect in-domain samples. [Liang, Li and Srikant, 2017] propose ODIN as a post-hoc method that utilizes a pretrained network to reliably separate OOD samples from the inlier distribution. They achieve this via a two-step procedure. First, the input image is perturbed in the gradient direction of the highest (inlier) softmax probability. Next, the softmax outputs of the classifier are scaled by a temperature, which is determined via a held out test set. While the authors report good performance, ODIN's effectiveness depends heavily on tuning the hyperparameters, namely the gradient step and the temperature. [DeVries and Taylor, 2018*a*] improved upon [Hendrycks and Gimpel, 2017] (and somewhat upon ODIN) by training networks to produce confidence estimates alongside softmax probabilities for outlier thresholding. Concurrently, [Lee et al., 2018] jointly trained a GAN alongside a classifier to generate realistic OOD examples. This required an additional OOD set during training, and the resulting classifiers were unable to generalize to unseen datasets.

2.3 Conclusion

This chapter has provided a comprehensive overview of the key concepts and prior work that form the foundation for the research presented in this thesis. The discussion of score matching and its connection to denoising autoencoders, noise-conditioned score networks, and diffusion models lays the groundwork for understanding the score-based methods developed in the subsequent chapters.

The survey of anomaly detection approaches - ranging from probability-based methods like mixture models and normalizing flows, to reconstruction-based techniques like autoencoders - highlights the diversity of existing methodologies. However, it also reveals limitations in the detection capabilities of these models, with models showing inconsistent performance across tasks or simply failing to detect OOD datasets. This

gap motivates the need for novel unsupervised anomaly detection frameworks that can effectively model structured, non-continuous data. The review of post-hoc out-of-distribution detection techniques provides further context for the need of an approach that is not label dependent and can generalize to models other than classifiers.

By establishing the theoretical underpinnings and contextualizing the challenges, this chapter equips the reader with the necessary background to fully appreciate the significance and novelty of the multiscale score matching approach introduced in the next chapter.

CHAPTER 3: MULTISCALE SCORE MATCHING

This chapter will introduce Multiscale Score Matching Analysis (MSMA); my method for outlier detection via score norms. The first sections will explore the intuition behind score-norms and the need for multiple scales. The latter sections empirically demonstrate MSMA's best-in-class performance on OOD benchmarks. Note that this work used Noise Conditioned Score Networks (NCSNs) [Song and Ermon, 2019], which have been superseded by diffusion models (employed in later chapters). However, NCSNs laid the foundation and proof of concept upon which the rest of the research was developed. Related code is available at `https://github.com/ahsanMah/msma`

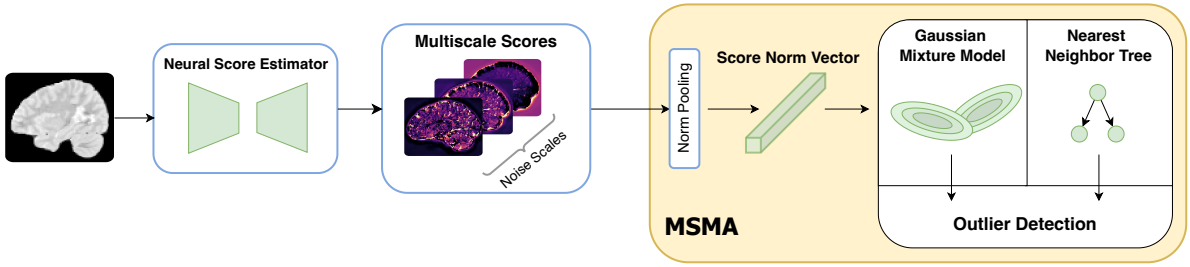


Figure 3.1: A schematic overview of MSMA. A neural score estimator produces score tensors at multiple noise scales for the inlier data. The score tensors are reduced using the L2 norm, producing the score-norm vectors. These vectors are fed into an auxiliary model such as a Gaussian Mixture Model, which learns the space of the score-norm vectors. The output of the auxiliary model will be used as the outlier score.

3.1 Multiscale Score Analysis

Recall that a score is the gradient of the log-likelihood with respect to the data. Intuitively, the score is a measure of how close a sample is to the local maximum in the log-probability-density space. Now consider taking the L2-norm of the score function: $||s(x)|| = ||\nabla_x \log p(x)|| = \left|\left|\frac{\nabla_x p(x)}{p(x)}\right|\right|$. Is the norm of a score function sufficient to identify samples with low density $p(x)$?

Since the data density term appears in the denominator, a high likelihood will correspond to a low norm. In contrast, out-of-distribution samples should have a low likelihood with respect to the in-distribution log density (i.e. $p(x)$ is small), and we can expect them to have high score norms. However, if these outlier

points reside in "flat" regions with very small gradients (or perhaps in a local mode), their score norms can be low despite the point belonging to a low density region. Figure 3.3 illustrates a possible scenario of outlying points. This is our first indication that a true score norm may not be sufficient for detecting outliers.

We can empirically validate this intuition by considering score estimates for a relatively simple dataset: FashionMNIST. Following the denoising score matching objective in Equation 2.5, one can obtain multiple estimates of the true score by using different noise distributions $q_\sigma(\tilde{x}|x)$. Like [Song and Ermon, 2019], I choose the noise distributions to be a zero-centered Gaussian scaled according to σ_i. Note that the scores for samples perturbed by the lowest σ noise should be closest to the 'true' score. My analysis shows that the true score alone was inadequate at separating inliers from OOD samples.

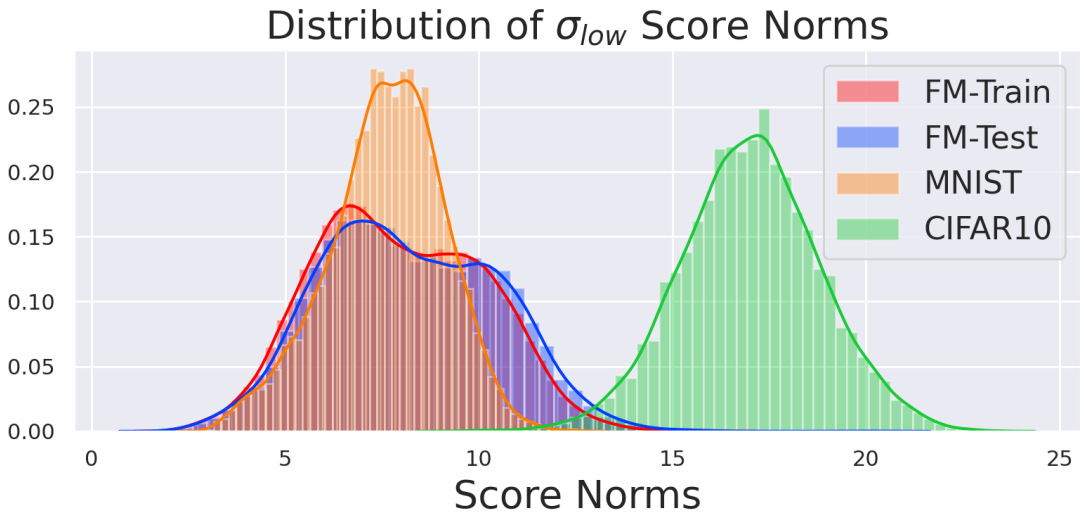
Why Do Multiscale Scores Capture Outliers?

In this section, I present an analysis in order to give an intuition for why multiple scales can be useful for OOD detection. Consider the toy distribution shown in Figure 3.3. We have three regions of interest: an inlier region with high density centered around $x = 10$, an outlier region with low density around $x = 30$, and a second outlier region with a local mode centered around $x = 50$. I will be adding noise of increasing scales to this distribution and observing the resulting score norms.

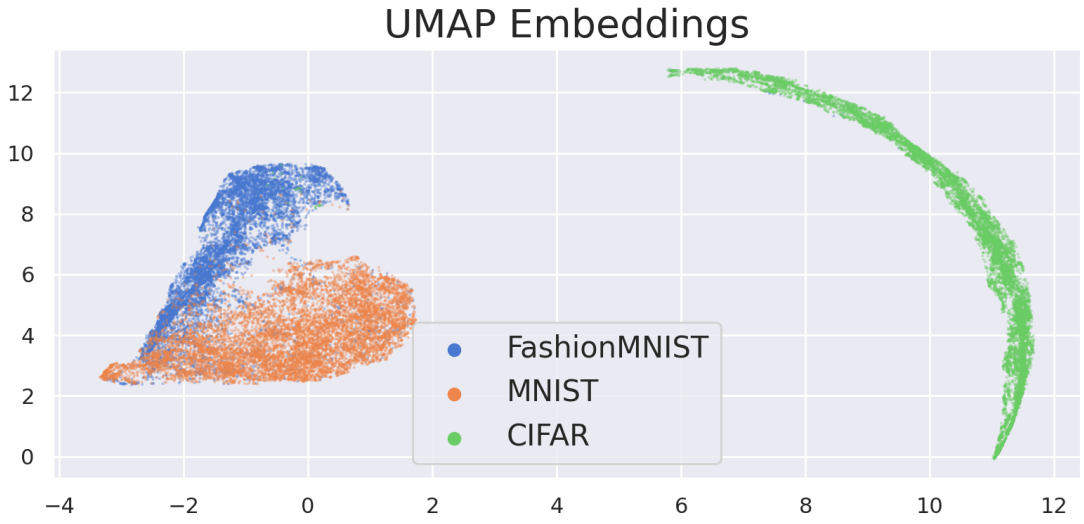
To begin, recall that sum of two probability distributions is equivalent to a convolution of their probability density functions, as per the law of total probability [Zwillinger and Kokoska, 1999]. Thus, I can analytically compute the perturbed distribution, as I know the parameters for both the base distribution (a Gaussian Mixture Model) and the noise $\mathcal{N}(0; \sigma_i)$. This not only enables us to visualize perturbations of our toy distribution, but also to analytically compute the score estimates given any σ_i.

On the left hand side, we have the initial density with no perturbation. Note how points within the low-density region and at the peak of the local-mode will have small gradients due to being inside a flat area. As we perturb the samples, we smooth the original density, causing it to widen. The relative change in density at each point is dependent on neighboring modes. A large scale perturbation will proportionally take a larger neighborhood into account at each point of the convolution. Therefore, at a sufficiently large scale, nearby outlier points gain context from in-distribution modes. This results in an increased gradient signal in the direction of inliers.

Figure 3.4 plots the score norms of samples generated from the original density along with markers indicating our key regions. Note how a small scale perturbation ($\sigma_L = 0.1$) stays very close to the true



(a) 1D score estimates from a model trained on FashionMNIST (FM)



(b) UMAP visualization of 10-dimensional score estimates from a model trained on FashionMNIST

Figure 3.2: Visualizing the need for a multiscale analysis. In (a), I plot the scores corresponding to the lowest sigma estimate. In (b), I plot the UMAP embedding of the $L = 10$ dimensional vectors of score norms. Here we see a better separation between FashionMNIST and MNIST when using estimates from *multiple* scales rather than the one that corresponds to the true score only.

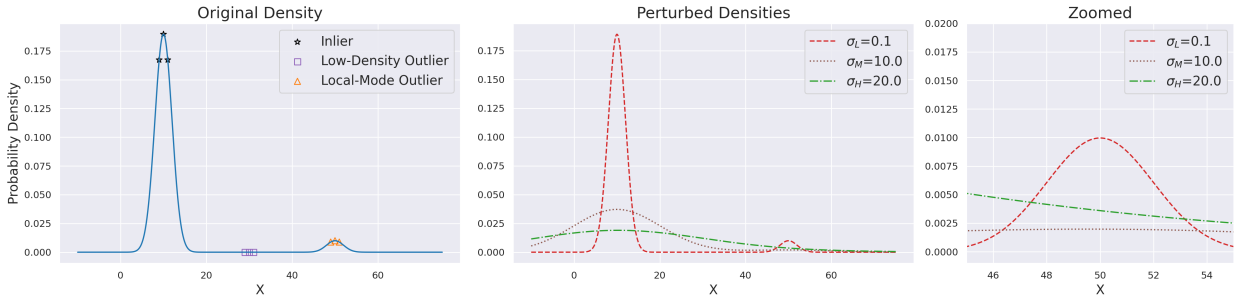


Figure 3.3: Left: A toy GMM to visualize my analysis with the three regions of interest. Right: Gaussian perturbed versions of the original distribution with (L)ow, (M)edium, and (H)igh noise levels, along with a plot zoomed into the local-mode outliers. Note the effect of different scales on this region: only the largest scale results in a gradient in the direction of the inliers.

Figure 3.4: In (a) observe that Low-Density outliers have comparatively high gradient norms for both σ_L and σ_M. However at this scale, Local-Mode points still have very small norms, causing them to be tightly packed around the in-distribution points. In (b) we see that Local-Mode outliers achieve a gradient signal only when a sufficiently high scale is used, $\sigma_H = 20$.

distribution. A medium scale ($\sigma_M = 10$) Gaussian perturbation is wide enough to "capture" the Low-Density outliers but still not wide enough to reach the inlier region from the Local-Mode outlier densities, causing them to simply fade into flat nothingness. It is only after we perform a large scale ($\sigma_H = 20$) perturbation that the in-distribution mode gets taken into account, resulting in a higher gradient norm. This analysis allows one to intuit that larger noise levels account for a larger neighborhood context. We surmise that given a sufficiently large scale, we can capture gradient signals from distant outliers.

Are Larger Scales Always Better?

However, it is imperative to note that no single scale is guaranteed to work for *all* outliers. Consider outliers close to inlier modes such as the samples between Low-Density outliers and Inliers in Figure 3.3. σ_H results in an overlap in the score distribution of inliers and Low-Density outliers. This makes it difficult to differentiate the aforementioned "in-between" outliers from the in-distribution samples. However, this large scale was necessary to get a big enough neighborhood context in order to capture the more distant Local-Mode outliers. Therefore, I postulate that a *range* of scales is necessary for separating outliers.

Admittedly, selecting this range according to the dataset is not a trivial problem. [Song and Ermon, 2020] outlined some techniques for selecting $\{\sigma_i\}_{i=1}^{L}$ for NCSNs from the perspective of generative modeling. Perhaps there is a similar analog to OOD detection. I did not perform such a statistical analysis for this work and used the default range for NCSN in all my experiments. However, I observed that my defaults are surprisingly generalizable, evident by the fact that all my experiments in Section 3.3 were performed with the same scale range. In Section 3.5, I further analyze how varying the scale range effects downstream accuracy and observe that the defaults were already near optimal.

3.2 Training an MSMA Model

I propose the inclusion of multiple noisy score estimates for the task of separating in- and out-of-distribution points, allowing for a Multiscale Score Matching Analysis (MSMA). Concretely, given L noise levels, we calculate the L2-norms of per-sample scores for each level, resulting in an L-dimensional vector for each input sample. Motivated by my analyses in Section 3.1, I posit that in-distribution data points occupy distinct and dense regions in the L-dimensional score space.

The *cluster assumption* states that decision boundaries should not pass high density regions, but instead lie in low density regions. This implies that any auxiliary method trained to learn in-distribution regions should be able to identify OOD data points that reside outside the learned space. Thus, I propose a two step unsupervised training scheme. First, one trains a NCSN model $s_{\text{IN}}(x, \sigma_L)$ to estimate scores for inlier samples, given $\{\sigma_i\}_{i=1}^{L}$ levels of noise. Next, we can calculate all L noisy score estimates for the N training samples and take the L2-norms across the input dimensions: $[||s_{\text{IN}}(X, \sigma_1)||_2^2, ..., ||s_{\text{IN}}(X, \sigma_L)||_2^2]$. This results in an NxL matrix. Finally, we can train an auxiliary model (such as a Gaussian Mixture Model) on this matrix to learn the spatial regions of in-distribution samples in the L-dimensional space.

Learning Concentration in the Score Space

We posit that learning the "density" of the inlier data in the L-dimensional score (norm) space is sufficient for detecting out-of-distribution samples. The term "density" can be interpreted in a myriad of ways. I primarily focus on models that fall under three related but distinct notions of denseness: spatial clustering, probability density, and nearest (inlier) neighbor graphs. All three allow us to threshold the associated metric to best separate OOD samples.

Spatial clustering is conceptually the closest to our canonical understanding of denseness: points are tightly packed under some metric (usually Euclidean distance). Ideally OOD data should not occupy the same cluster as the inliers. I train Gaussian Mixture Models (GMMs) to learn clusters in the inlier data. GMMs work under the assumption that the data is composed of k-components whose shapes can be described by a (multivariate) Gaussian distribution. Thus for a given datum, one can calculate the joint probability of it belonging to any of the k-components.

Probability density estimation techniques aim to learn the underlying probability density function $p_{data}(x)$ which describes the population. Normalizing flows are a family of flexible methods that can learn tractable density functions ([Papamakarios et al., 2021]). They transform complex distributions into a simpler one (such as a Gaussian) through a series of invertible, differential mappings. The simpler base distribution is then used to infer the density of a given sample. For this work, I use Masked Autoregressive Flows introduced by [Papamakarios, Pavlakou and Murray, 2017], which utilizes neural networks as the transformation functions. The inlier likelihoods are used to determine a threshold after which samples are considered outliers.

Finally, I consider k-nearest neighbor (k-NN) graphs to allow for yet another thresholding metric. Conceptually, the idea is to sort all samples according to distances to their k-closest (inlier) neighbor.

Presumably, samples from the same distribution as the inliers will have very short distances to training data points. Despite its simplicity, this method is surprisingly accurate. It is also computationally efficient as k-NN distances can be computed via efficient data structures (such as KD Trees).

3.3 Experiments on Benchmark Datasets

In this section I demonstrate MSMA's potential as an effective OOD detector. I first train a NCSN model as my score estimator, and then an auxiliary model on the score estimates of the training set. Following [Liang, Li and Srikant, 2017] and [DeVries and Taylor, 2018*b*], I use CIFAR-10 and SVHN as my "inlier" datasets alongside a collection of natural images as "outlier" datasets. I retrieve the natural image from ODIN's publicly available GitHub repo[1]. This helps maintain a fair comparison (e.g. it ensures I test the same random crops as ODIN). I will denote [Liang, Li and Srikant, 2017] as ODIN and [DeVries and Taylor, 2018*b*] as Confidence in my tables. In addition to experiments performed by [Hendrycks and Gimpel, 2017], [Liang, Li and Srikant, 2017] and [DeVries and Taylor, 2018*b*], I also distinguish *between* CIFAR and SVHN and compare my results to baselines.

Datasets

I consider CIFAR-10 ([Krizhevsky, Hinton et al., 2009]) and SVHN ([Netzer et al., 2011]) as my inlier datasets. For out-of-distribution datasets, I choose the same as [Liang, Li and Srikant, 2017]: **TinyImageNet**[2], **LSUN** ([Yu et al., 2015]), **iSUN** ([Xu et al., 2015]). Similar to [DeVries and Taylor, 2018*b*], in my main experiments I report only *resized* versions of *LSUN* and *TinyImageNet*. I also leave out the synthetic **Uniform** and **Gaussian** noise samples from my main experiments as I performed extremely well in all of those experiments.

Lastly following [DeVries and Taylor, 2018*b*], I consider **All Images**: a combination of all non-synthetic OOD datasets outlined above (including *cropped* versions). Note that this collection effectively requires a single threshold for all datasets, thus arguably reflecting a real world out-of-distribution setting.

[1] `https://github.com/facebookresearch/odin`

[2] `https://tiny-imagenet.herokuapp.com/`

Evaluation Metrics

To measure detection performance it is common to use threshold-free metrics established by previous baselines ([Hendrycks and Gimpel, 2017], [Liang, Li and Srikant, 2017]). These include:

FPR at 95% TPR: This is the False Positive Rate (FPR) when the True Positive Rate (TPR) is 95%. This metric can be interpreted as the probability of misclassifying an outlier sample to be in-distribution when the TPR is as high as 95%. Let TP, FP, TN, and FN represent true positives, false positives, true negatives and false negatives respectively. FPR=FP/(FP+TN), TPR=TP/(FN+TP).

Detection Error: This measures the minimum possible misclassification probability over all thresholds. Practically this can be calculated as $\min_\delta 0.5(1 - \text{TPR}) + 0.5\text{FPR}$, where it is assumed that we have an equal probability of seeing both positive and negative examples in the test set.

AUROC: This measures area under (AU) the Receiver Operating Curve (ROC) which plots the relationship between FPR and TPR. It is commonly interpreted as the probability of a positive sample (in-distribution) having a higher score than a negative sample (out-of-distribution). It is a threshold independent, summary metric.

AUPR: Area Under the Precision Recall Curve (AUPR) is another threshold independent metric that considers the PR curve, which plots Precision(= TP/(TP+FP)) versus Recall(= TP/(TP+FN)). AUPR-In and AUPR-Out consider the in-distribution samples and out-of-distribution samples as the positive class, respectively. This helps take mismatch in sample sizes into account.

These metrics are useful for evaluating the performance of outlier detection models and understanding their strengths and weaknesses. The FPR at 95% TPR metric provides a direct measure of the false probability rate, which is crucial to know for applications where a high true positive rate is required. The Detection Error metric gives an overall measure of the misclassification probability, helping to understand the model's average accuracy at detcting both inliers and outliers. The AUROC metric evaluates the model's ability to rank outliers higher than inliers, with better ranking leading to a better trade-off between true-positves and false-positives. Similarly, the AUPR metric considers the precision and recall trade-off, accounting for imbalances in the sample sizes of the two classes.

Together, these metrics provide a comprehensive evaluation of the model's performance, capturing different types of errors (false positives and false negatives) and allowing for comparisons across different operating points and scenarios. By considering multiple metrics, researchers and practitioners can gain

insights into the model's behavior and make informed decisions based on the specific requirements and constraints of their applications.

Comparison Against Previous OOD Methods

I compare my work against Confidence Thresholding ([DeVries and Taylor, 2018*b*]) and ODIN ([Liang, Li and Srikant, 2017]). For all experiments I report the results for the in-distribution *testset* vs the out-of-distribution datasets. Note that *All Images** is a version of *All Images* where both ODIN and Confidence Thresholding perform input preprocessing. Particularly, they perturb the samples in the direction of the softmax gradient of the classifier: $\tilde{x} = x - \epsilon \operatorname{sign}(-\nabla_x log S_y(x;T))$. They then perform a grid search over ϵ ranges, selecting the value that achieves best separation on 1000 samples randomly held for *each* out-of-distribution set. ODIN performs an additional search over T ranges, while Confidence Thresholding uses a default of $T = 1000$. I did not perform any such input modification. Note that ODIN uses input thresholding for individual OOD datsets as well, while Confidence Thresholding does not.

The tables below report accuracy metrics for three different auxiliary models: Gaussian Mixture Model (GMM), Normalizing Flow density estimator (Flow), and a nearest-neighbour-based model (KD Tree). The purpose of reporting results from different models is to showcase the flexibility of MSMA. Each auxiliary model provides a different notion of "distance" from the inliers. The results show that MSMA clearly outperforms baselines. Performance metrics for ODIN and Confidence Thresholding were taken from the original papers. Therefore results for some experiments are missing.

In-distribution Dataset	OOD Dataset	GMM	Flow	KD Tree	ODIN (DenseNet)	Confidence (VGG13)
SVHN	TinyImageNet	0.0	0.0	0.0	-	1.8
	LSUN	0.0	0.0	0.0	-	0.8
	iSUN	0.0	0.0	0.0	-	1.0
	All Images	0.0	0.0	0.0	-	4.3
	All Images*	-	-	-	8.6	4.1
CIFAR-10	TinyImageNet	0.0	0.0	0.3	7.5	18.4
	LSUN	0.0	0.0	0.6	3.8	16.4
	iSUN	0.0	0.0	0.4	6.3	16.3
	All Images	0.0	0.0	0.4	-	19.2
	All Images*	-	-	-	7.8	11.2

Table 3.1: Results for **FPR (95% TPR)**. *Lower* values are better.

In-distribution Dataset	OOD Dataset	GMM	Flow	KD Tree	ODIN (DenseNet)	Confidence (VGG13)
SVHN	TinyImageNet	100.0	100.0	100.0	-	99.6
	LSUN	100.0	100.0	100.0	-	99.8
	iSUN	100.0	100.0	100.0	-	99.8
	All Images	100.0	100.0	100.0	-	99.2
	All Images*	-	-	-	97.2	99.2
CIFAR-10	TinyImageNet	100.0	100.0	99.9	98.5	97.0
	LSUN	100.0	100.0	99.9	99.2	97.5
	iSUN	100.0	100.0	99.9	98.8	97.5
	All Images	100.0	100.0	99.9	-	97.1
	All Images*	-	-	-	98.4	98.0

Table 3.2: Results for **AUROC**. *Higher* values are better. All three auxiliary methods perform better than baselines.

In-distribution Dataset	OOD Dataset	GMM	Flow	KD Tree	ODIN (DenseNet)	Confidence (VGG13)
SVHN	TinyImageNet	0.0	0.0	0.1	-	3.1
	LSUN	0.0	0.0	0.1	-	2.0
	iSUN	0.0	0.0	0.1	-	2.2
	All Images	0.0	0.0	0.1	-	4.6
	All Images*	-	-	-	6.8	4.5
CIFAR-10	TinyImageNet	0.0	0.0	1.0	6.3	9.4
	LSUN	0.0	0.1	1.5	4.4	8.3
	iSUN	0.0	0.0	1.2	6.7	8.5
	All Images	0.0	0.0	1.2	-	9.1
	All Images*	-	-	-	6.0	6.9

Table 3.3: Results for **Detection Error**. *Lower* values are better.

In-distribution Dataset	OOD Dataset	GMM	Flow	KD Tree	ODIN (DenseNet)	Confidence (VGG13)
SVHN	TinyImageNet	100.0	100.0	100.0	-	99.8
	LSUN	100.0	100.0	100.0	-	99.9
	iSUN	100.0	100.0	100.0	-	99.9
	All Images	100.0	100.0	100.0	-	98.5
	All Images*	-	-	-	92.5	98.6
CIFAR-10	TinyImageNet	100.0	100.0	99.9	98.6	97.3
	LSUN	100.0	100.0	99.8	99.3	97.8
	iSUN	100.0	100.0	99.9	98.9	98.0
	All Images	100.0	100.0	99.9	-	92.0
	All Images*	-	-	-	95.3	94.5

Table 3.4: Results for **AUPR-In** with In-distribution as positive class. *Higher* values are better

In-distribution Dataset	OOD Dataset	GMM	Flow	KD Tree	ODIN (DenseNet)	Confidence (VGG13)
SVHN	TinyImageNet	100.0	100.0	99.9	-	99.1
	LSUN	100.0	100.0	99.9	-	99.6
	iSUN	100.0	100.0	99.9	-	99.5
	All Images	100.0	100.0	99.9	-	99.6
	All Images*	-	-	-	98.6	99.5
CIFAR-10	TinyImageNet	100.0	100.0	99.9	98.5	96.9
	LSUN	100.0	100.0	99.9	99.2	97.2
	iSUN	100.0	100.0	99.9	98.8	96.9
	All Images	100.0	100.0	99.9	-	99.3
	All Images*	-	-	-	99.6	99.5

Table 3.5: Results for **AUPR-Out** with Out-of-Distribution as positive class. *Higher* values are better

Auxiliary Method	FPR (95% TPR) ↓	Detection Error ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
GMM	11.4	8.1	95.5	91.9	96.9
Flow	8.6	6.8	96.7	93.4	97.7
KD Tree	**4.1**	**4.5**	**99.1**	**99.1**	**99.2**

Table 3.6: Comparison of auxiliary models tasked to separate CIFAR-10 (in-distribution) and SVHN (out-of-distribution). ↓ indicates lower values are better and ↑ indicates higher values are better.

Detection Function	Model	FPR (80% TPR) ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
$\|\|s_\theta(x, \sigma_i^L)\|\|$	KD Tree MSMA	**0.7**	**99.1**	**99.1**	**99.2**
$\log \frac{p_\theta(x)}{p_{\theta_0}(x)}$	Likelihood Ratios	6.6	93.0	88.1	-
$\log p(x)$	JEM	-	67.0	-	-
$\left\|\left\|\frac{\partial \log p(x)}{\partial x}\right\|\right\|$	Approx. Mass JEM	-	83.0	-	-

Table 3.7: Comparison with a multitude of likelihood-based models at separating CIFAR-10 (in-distribution) from SVHN (out-of-distribution). - represent metrics that were not reported by the work. All values are shown in percentages. ↓ indicates lower values are better and ↑ indicates higher values are better. Note that since Likelihood Ratios report FPR at **80%** TPR, we report the same.

MSMA Succeeds where Likelihoods Fail

[Nalisnick et al., 2019] showed how deep generative models are particularly inept at separating high dimensional complex datasets such as these two. One exemplary experiment kept CIFAR-10 as in-distribution and SVHN as out-of-distribution. It is common for researchers to use this experiment to explore the OOD failure of likelihood models [Kirichenko, Izmailov and Wilson, 2020*a*; Ren et al., 2019; Grathwohl et al., 2020]. Since this setting is not considered in the testbed used by ODIN or Confidence Thresholding, I did not report their results.

Table 3.6 shows MSMAs performance for this task. Note that *all* three auxiliary models definitively outperform the baselines (see Table 3.7), with KD-Trees preforming the best. Likelihood Ratios [Ren et al., 2019], and JEM [Grathwohl et al., 2020] are two unsupervised methods that have tackled this problem and have reported current state-of-the-art results. Table 3.7 summarizes the results that were reported by these papers. Both report AUROCs, with [Ren et al., 2019] additionally reporting AUPR(In) and FPR at 80% TPR. Since each method proposes a different detection function, we also provide them for reference.

3.4 Case Study: Age based OOD Detection in Brain MRIs

In this section I report MSMA's performance on a real-world case study. Here the task is to distinguish brain Magnetic Resonance Image (MRI) scans, based on age. I consider pediatric subjects at school-age (9 - 11 years of age) as the inlier distribution and a younger age group (1 - 6 years) as out-of-distribution. We expect visible differences in image contrast and local brain morphometry between the brains of toddlers and school-age children. As a child grows, their brain matures and the corresponding scans appear more like the prototypical adult brain. This provides an interesting gradation of samples being considered out-of-distribution with respect to age.

I employ 3500 high resolution T1-weighted MR images obtained through the NIH large scale ABCD study [Casey et al., 2018], which represent data from the general adolescent population (9-11 years of age). This implies that my in-distribution dataset will have high variation. After standard preprocessing, we extracted for each dataset three mid-axial slices and resized them to be 90x110 pixels, resulting in $\sim$11k axial images (10k training, 1k testing). For my outlier data, I sourced MRI datasets of children aged 1, 2, 4 and 6 years (500 each) from the UNC EBDS database [Stephens et al., 2020; Gilmore et al., 2020].

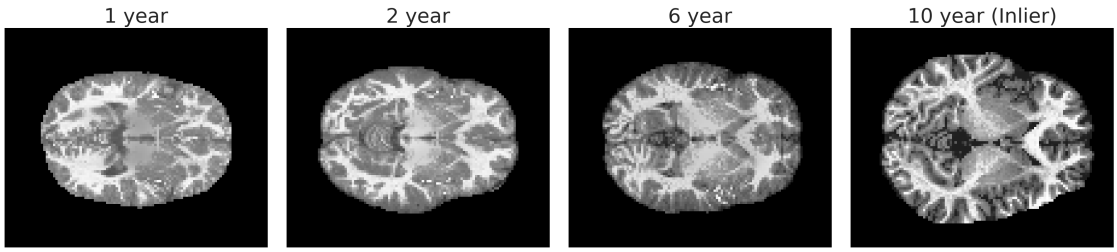


Figure 3.5: Note the change in image contrast, brain size and brain matter structure as the child grows. Each age is increasingly difficult to distinguish from our inliers.

MSMA was effectively able to identify younger age groups as out-of-distribution. Table 3.8 reports the results for different MSMA auxiliary models trained for this task. As expected, the separation performance decreases as age increases. Note that I kept the same hyperparameters for our auxiliary methods as in the previous experiments despite this being a higher resolution scenario.

Auxiliary Method	OOD Age (Years)	FPR (95% TPR) ↓	Detection Error ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
GMM	1	0.2	0.4	99.9	99.9	99.9
	2	0.6	1.0	99.7	99.5	99.9
	4	23.7	9.2	96.1	93.8	97.9
	6	30.5	9.7	95.0	92.2	96.8
Flow	1	0.2	0.3	99.9	99.9	99.9
	2	0.6	1.3	99.7	99.4	99.9
	4	12.2	8.4	97.3	94.6	98.8
	6	28.9	12.5	94.3	88.7	97.5
KD Tree	1	2.5	2.6	99.3	98.2	99.7
	2	3.6	3.1	98.9	96.2	99.6
	4	18.6	10.7	95.7	91.0	98.0
	6	39.2	14.9	91.6	84.2	95.8

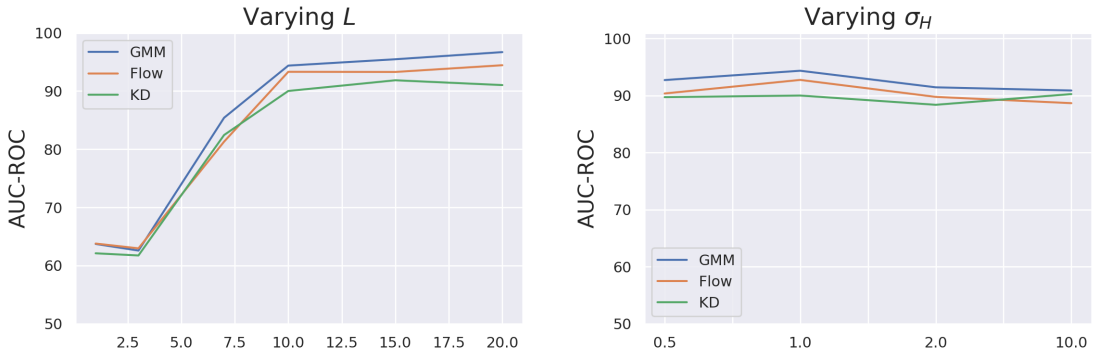
Table 3.8: Comparison of all auxiliary models tasked to separate the brain scans of different age groups. In-distribution samples are 9-11 years of age. All values are shown in percentages. ↓ indicates lower values are better and ↑ indicates higher values are better.

Further, I compared MSMA to f-AnoGAN [Schlegl et al., 2019] due to its promising results as an anomaly detector and its popularity in the medical community. I use the hyperparameters suggested in the original paper and trained them till convergence. Table 3.9 compares the performance of f-AnoGAN with MSMA-GMM. Note that MSMA outperforms f-AnoGAN in *all* experiments. However, f-AnoGAN is a considerably faster method, both in terms of training and inference, and allows for pixel-wise anomalies out of the box. Although, it should be noted that Spatial-MSMA introduced in Chapter 5 will overcome the latter limitation by adding localization capabilities to MSMA.

OOD Age (Years)	FPR (95% TPR) ↓	Detection Error ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
	f-AnoGAN / MSMA				
1	94.0 / **0.2**	24.1.0 / **0.4**	72.4 / **99.9**	80.2 / **99.9**	62.2 / **99.9**
2	62.7 / **0.6**	13.8 / **1.0**	90.6 / **99.7**	92.5 / **99.5**	87.0 / **99.9**
4	60.9 / **23.7**	19.0 / **9.2**	87.9 / **96.1**	88.9 / **93.8**	85.8 / **97.9**
6	78.1 / **30.5**	31.1 / **9.7**	74.6 / **95.0**	76.4 / **92.2**	72.4 / **96.8**

Table 3.9: MSMA-GMM trained on multiscale score estimates tasked to separate the brain scans of different age groups. In-distribution samples are 9-11 years of age. All values are shown in percentages. ↓ indicates lower values are better and ↑ indicates higher values are better. The results show that f-AnoGAN is unable to match the performance of MSMA for this task. In fact, under some metrics such as FPR at 95% TPR, it exhibits very poor performance as an anomaly detector.

3.5 Hyperparameter Analysis



(a) Varying L from 1 to 20

(b) Varying σ_H from 0.5 to 10

Figure 3.6: Analysis of the effect of hyperparameters σ_H and L on MSMA's out-of-distribution detection performance. We observe that the defaults $\sigma = 1.0$ and $L = 10$ perform the best, with a slight variance in performance when we deviate from them.

This section explores MSMA's sensitivity to its two main hyperparameters utilized in NCSN: the number of noise levels (L) and the largest scale used (σ_H). The smallest noise scale was fixed to $\sigma_L = 0.01$ through all experiments as it is low enough to learn the true score values without running into issues regarding numerical precision. The rationale being that noise perturbations to images at such a small scale are imperceptible to the human eye and are adequate to give an estimate of the true score. For all experiments, I train on CIFAR-10 and evaluate on the *All Images* dataset described in Section 3.3 and plot AUROC in Figure 3.6. In order to reduce GPU memory usage and computation time, I reduced the batch size from 128 to 64, which is why

we see a performance dip from the main experiment in Section 3.3. The hyperparameters for the auxiliary models are kept the same as the main experiment.

For the number of scales L, I tested the values 1, 3, 10, 15, and 20, with σ_H fixed at the default value ($\sigma_H = 1$). Recall that I follow the original NCSN schema by [Song and Ermon, 2019], and utilize a geometric sequence of sigma scales from σ_H to σ_L with L steps (endpoint inclusive). Thus, changing L changes the intermediate noise scales. The results in Figure 3.6a show that MSMA is optimal near the default ($L = 10$). Increasing L does not significantly vary the performance, while small values such as $L = 3$ are not adequate at providing enough intermediate scales. Note that $L = 1$ is the degenerate case where only the largest noise scale is used. This highlights the need for a range of scales as argued in Section 3.1 and empirically shows that simply using one large scale is not enough. Figure 3.6b plots the affect of varying the largest noise scale σ_H. I test the values 0.5, 1.0, 2.0, and 10, with the default number of scales $L = 10$. Again, we observe that our default $\sigma_H = 1$ performs the best and there is no noticeable improvement from varying it. Considering how images are rescaled to [0,1] before they are passed to the network, we posit that $\sigma_H = 1.0$ already introduces large noise, and increasing it further seems to degrade results to varying degrees.

Lastly, I would like to emphasize that *all* of my main out-of-distribution experiments in Section 3.3 were performed with the *same* default hyperparameters, without any tuning. Despite this disadvantage, MSMA still outperforms its competitors ODIN [Liang, Li and Srikant, 2017], Confidence Thresholding ([DeVries and Taylor, 2018*b*]), and Likelihood Ratios [Ren et al., 2019], all of which need some fine-tuning of hyperparameters. Recognize that tuning requires apriori knowledge of the type of out-of-distribution samples the user expects. From the analysis in this section and my main experiment, *I can confidently advocate the use of MSMA's defaults as they generalize across datasets and do not require such apriori knowledge.*

3.6 Conclusion

This chapter introduced MSMA: a method based on multiscale score matching. MSMA is easy to implement, trains completely unsupervised, requires minimal hyperparameter tuning, and generalizes to many OOD tasks. Section 3.1 outlines how scaling the noise perturbation is analogous to increasing context from local neighborhoods in the data space. Section 3.3 demonstrates how MSMA outperforms baseline methods in *every* metric for *all* benchmark experiments.

The excellent results reported in this chapter empirically validate MSMA as a fast, general purpose anomaly detector. The following chapters will focus on extending MSMA and applying it to 3D MRI data for detecting neurodevelopmental disorders.

CHAPTER 4: SCORE MATCHING FOR CATEGORICAL DATA VIA GUMBEL NOISE

It is common to represent information as discrete categories such as gender, shapes or product quality. One may wish to detect anomalies within this data type, in addition to continuous variables. However, MSMA, as it was introduced in the previous chapter is unable to model categorical data. This is because a score is ill-defined for categorical variables. There is no notion of a gradient for probability density functions [1] of discrete data. If we can overcome this limitation, we will be able to incorporate more information into MSMA. Ultimately, my goal is to allow MSMA to model any available metadata that is paired with the continuous features, e.g. demographic data paired with brain MRIs. I posit that this mixed data type modeling would enable more robust anomaly detection decisions.

This chapter introduces Gumbel Noise Score Matching (GNSM), a novel method for learning scores of categorical data for the purpose of anomaly detection. The chapter first develops the theoretical foundations of GNSM, showing how continuous relaxations of categorical distributions through the Gumbel-Softmax trick can be combined with denoising score matching to estimate scores for categorical variables. The resulting objective allows the model to directly output score estimates for one-hot encoded categorical features. Experiments on a suite of tabular anomaly detection datasets show GNSM-MSMA achieving competitive or state-of-the-art performance compared to baseline methods, with significant improvements on certain datasets. A case study on detecting anomalous image segmentations further illustrates the versatility of the approach beyond just tabular data.

The key contributions of this chapter are: 1) Developing a principled score matching framework for categorical data, and 2) Demonstrating the capability of score matching for anomaly detection on categorical types in both tabular and image datasets. The chapter highlights GNSM as a powerful and flexible approach for handling structured, non-continuous data types in anomaly detection. Related code is available at `https://github.com/ahsanMah/categorical-dsm`

[1] In this case, it is more correct to refer to them as probability *mass* function

4.1 A Recipe for Categorical Score Matching

In this section I will develop the ideas behind the GNSM objective. I will start by introducing a technique for continuous approximations (i.e. relaxations) for categorical distributions. I will then formulate an objective (based on denoising score matching) to learn the scores of the relaxed categorical variables.

Continuous Relaxation to Categorical Data

Gradients of log likelihoods are ill-defined for categorical inputs. In order to compute the score of categorical data, I propose to adopt a continuous relaxation for discrete random variables co-discovered by [Jang, Gu and Poole, 2017; Maddison, Mnih and Teh, 2017]. These relaxations build on the Gumbel-Max trick to sample from a categorical distribution [Maddison, Tarlow and Minka, 2014]. The procedure (often referred to as the Gumbel-Softmax) works by adding Gumbel noise [Gumbel, 1954] to the (log) probabilities and then passing the resulting vector through a softmax to retrieve a sharpened probability distribution over the categorical outcomes. Of particular interest to us, Gumbel-Softmax incorporates a temperature parameter (λ in Equation (4.1)) to control the sharpening of the resulting probabilities. I argue that this temperature can also be interpreted as a noise parameter, by virtue of it increasing the entropy of the post-softmax probabilities. I will make use of this intuition to combine continuous relaxations with denoising score matching.

Note that for the rest of the chapter, I will be utilizing the formulation of [Maddison, Mnih and Teh, 2017] i.e. concrete random variables. In particular, I will be using a variant of the Concrete Distribution called ExpConcrete introduced by the same authors (shown in Equation 4.1). Given unnormalized probabilities $\alpha \in (0, \infty)^K$, Gumbel i.i.d samples G_k, and a smoothing factor $\lambda \in (0, \infty)$, we can construct an ExpConcrete random variable $Y \in \mathbb{R}^n$ such that $\exp(Y) \sim \text{Concrete}(\alpha, \lambda)$:

$$Y_k = \frac{\log \alpha_k + G_k}{\lambda} - \log \sum_{i=1}^{K} \exp\left\{\frac{\log \alpha_i + G_i}{\lambda}\right\} \tag{4.1}$$

As $\lambda \to 0$, the computation approaches an argmax, while large values of λ will push the random variable towards a uniform distribution. The main purpose of preferring the ExpConcrete Distribution over the Concrete Distribution is numerical stability, as the former is defined in the log domain.

Score Matching with Categorical Variables

We now have the basic ingredients to start learning scores for our relaxed-categorical data. Firstly, note that the proof of the denoising score matching objective by [Vincent, 2011] (Equation 2.4) holds true for any q_σ, provided that $\log q_\sigma(\tilde{x}|x)$ is differentiable. Recall that q_σ plays the role of a noise distribution. While most denoising score matching models incorporate Gaussian perturbation [Song and Ermon, 2019; Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020; Vincent, 2011], I emphasize that *any* noise distribution may be used during training.

ExpConcrete(α, λ) as a Noise Distribution

Following the reasoning above and the temperature parameter (λ) available in Equation 4.1, I propose to repurpose the Concrete distribution to add 'noise' to our continuous relaxations of the categorical variables. Increasing λ will allow us to corrupt the input x by scaling the logits and smoothing out the categorical probabilities. Therefore, in GNSM, the (Exp)Concrete Distribution acts both as the relaxation mechanism *and* the noise distribution:

$$\log q_\sigma(\tilde{\mathbf{x}}|\mathbf{x}) = \log p_\lambda(\tilde{\mathbf{x}}; \alpha = \mathbf{x})$$

Here, the location parameter is that of the unperturbed input (similar to how one would use a Gaussian kernel), and λ is a known hyperparameter. I set α to be the logits of $\mathbf{x}$. As $\mathbf{x} \in \{0, 1\}^K$ will be a one-hot encoding for K outcomes, it does not strictly satisfy the requirement $\alpha \in (0, \infty)^K$. This can be circumvented by adding a small delta to the vectors to avoid zero values i.e. $\alpha = \mathbf{x} + \delta$. While it is possible to any transformation to convert $\mathbf{x}$ to unnormalized probabilities, I opted to use the clamped one-hot encodings for simplicity.

Score of ExpConcrete Distribution

To plug ExpConcrete into the DSM objective (Equation 2.4), I first need to derive the the score for the ExpConcrete distribution i.e. take the gradient of the log-density with respect to the data. Conveniently, the

authors of [Maddison, Mnih and Teh, 2017] derived the log-density of an ExpConcrete random variable, which I will be using going forward:

$$\log p_{\alpha,\lambda}(x) = \log((K-1)!) + (K-1)\log\lambda \; + \left(\sum_{k=1}^{K} \log\alpha_k - \lambda x_k\right) - K\log\sum_{k=1}^{K} e^{(\log\alpha_k - \lambda x_k)} \tag{4.2}$$

Here $x \in \mathbb{N}$ such that $\log\sum_{k=1}^{K}\exp(x) = 0$. Since the first two terms for $\log p_{\alpha,\lambda}(x)$ in Equation 4.2 are independent of x, we can ignore them and focus on the latter:

$$\begin{aligned}
\nabla_{x_j}\log p_{\alpha,\lambda}(\mathbf{x}) &= \nabla_{x_j}\left(\sum_{k=1}^{K}\log\alpha_k - \lambda x_k\right) - \nabla_{x_j}\left(K\log\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}\right) && (4.3)\\
&= \nabla_{x_j}\left(-\sum_{k=1}^{K}\lambda x_k\right) - K\left(\nabla_{x_j}\log\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}\right) && (4.4)\\
&= -\lambda - K\frac{\nabla_{x_j}\left(\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}\right)}{\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}} && (4.5)\\
&= -\lambda - K\frac{\exp\{\log\alpha_j - \lambda x_j\}\nabla_{x_j}(\log\alpha_j - \lambda x_j)}{\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}} && (4.6)\\
&= -\lambda - K\frac{\exp\{\log\alpha_j - \lambda x_j\}(-\lambda)}{\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}} && (4.7)\\
&= -\lambda + \lambda K\,\frac{\exp\{\log\alpha_j - \lambda x_j\}}{\sum_{k=1}^{K}\exp\{\log\alpha_k - \lambda x_k\}} && (4.8)
\end{aligned}$$

Note how the last equation can be rewritten as:

$$\nabla_{x_j}\log p_{\alpha,\lambda}(\mathbf{x}) = -\lambda + \lambda K\;\sigma(\log\boldsymbol{\alpha} - \lambda\mathbf{x})_j \tag{4.9}$$

where $\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_k}}$ is the softmax function.

Gumbel-Noise Score Matching Objective

Equation 4.9 represents the score function of the ExpConcrete distribution i.e. the gradient of the log-density with respect to the data. We can now combine the ideas from Denoising Score Matching and Concrete random variables. Combining Equation 2.4 and Equation 4.9, one obtains

$$
\begin{aligned}
J(\theta) &= \mathbb{E}_{q_\sigma}\left[||s_\theta(x) - \nabla_{\tilde{x}} \log q(\tilde{x}|x)||^2\right] \\
&= \mathbb{E}_{p_\lambda}\left[||s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log p_\lambda(\tilde{\mathbf{x}}|\mathbf{x})||^2\right] \\
&= \mathbb{E}_{p_\lambda}\left[||s_\theta(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log p_\lambda(\tilde{\mathbf{x}}; \boldsymbol{\alpha} = \mathbf{x})||^2\right] \\
&= \mathbb{E}_{p_\lambda}\left[||s_\theta(\tilde{\mathbf{x}}) - (-\lambda + \lambda K\ \sigma(\log \mathbf{x} - \lambda\tilde{\mathbf{x}}))||^2\right] \\
&= \mathbb{E}_{p_\lambda}\left[||s_\theta(\tilde{\mathbf{x}}) - \lambda K\ \sigma(\log \mathbf{x} - \lambda\tilde{\mathbf{x}})) + \lambda||^2\right] \\
&= \mathbb{E}_{p_\lambda}\left[||s_\theta(\tilde{\mathbf{x}}) - \lambda K\ \sigma(\epsilon) + \lambda||^2\right]
\end{aligned}
$$

Here $\epsilon = \log \mathbf{x} - \lambda\tilde{\mathbf{x}}$ and can be loosely interpreted as the "logit noise" as it is the difference between the original logit probabilities and the perturbed vector. This formulation is analogous to the simplification utilized by [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020; Ho, Jain and Abbeel, 2020]. It allows me to train the model to estimate the noise directly as the other variables are known constants. Assume a network ϵ_θ, that takes the input $\tilde{\mathbf{x}}$. Following Equation 4.9, I parameterize a score network as $s_\theta(\tilde{\mathbf{x}})_j = -\lambda + \lambda K\ \sigma(\epsilon_\theta(\tilde{\mathbf{x}}))_j$. I train the network ϵ_θ to estimate the noise values ϵ by the objective below.

$$
J(\theta) = \mathbb{E}_{p_\lambda}\left[\lambda^2 K^2 ||(\sigma(\epsilon_\theta(\tilde{\mathbf{x}})) - \sigma(\epsilon))||^2\right] \tag{4.10}
$$

Following [Song and Ermon, 2019], I can modify my loss to train a NCSN with L noise levels i.e. $\lambda \in \{\lambda_i\}_{i=1}^L$:

$$
J_{GNSM}(\theta) = \sum_{i=0}^{L} \lambda_i^2 K^2\ \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \mathbb{E}_{\tilde{\mathbf{x}}\sim p_{\lambda_i}}\left[||\sigma(\epsilon_\theta(\tilde{\mathbf{x}}, \lambda_i)) - \sigma(\epsilon)||^2\right]
$$

Note that our network is now additionally conditioned on the noise level λ. Finally, our loss objective can be extended to incorporate data with multiple categorical features. For D categories we have:

$$
\sum_{d=0}^{D}\sum_{i=0}^{L} \lambda_i^2 K_d^2\ \mathbb{E}_{\mathbf{x_d}\sim p_{\text{data}}} \mathbb{E}_{\tilde{\mathbf{x}}_d\sim p_{\lambda_i}}\left[||\sigma(\epsilon_\theta(\tilde{\mathbf{x}}_d, \lambda_i)) - \sigma(\epsilon)||^2\right] \tag{4.11}
$$

Here, K_d represents the number of outcomes per category, x_d represents the one-hot vector of length K_d, and $\tilde{\mathbf{x}}_d$ is the continuous, noisy representation of x_d obtained after a Concrete (Gumbel-Softmax) transform.

Thus, I have shown that a Concrete relaxation allows one to model the scores of categorical variables by acting as the noise distribution in the DSM objective. The network will output the scores of the logits representing the categorical feature. Intuitively, these scores are gradients on a simplex which are pointing in the direction of the category that maximizes the likelihood of the datapoint.

A Note on Optimizing the GNSM Objective in Practice

Observing the loss in Equation 4.11, we see that we are minimizing the difference between two distributions as both inner terms pass through a softmax function. This insight led me to postulate that that one could substitute the mean squared error loss (MSE) for a metric more apt for matching distributions. I therefore ran experiments using the KL divergence objective as shown in (4.12). This objective showed faster convergence than MSE. Admittedly, this result is only empirical. It may be possible to gain similar improvements in convergence for the MSE by properly tuning the optimization hyperparameters such as the learning rate.

$$J_{GNSM}(\theta) = \sum_{d=0}^{D} \sum_{i=0}^{L} \lambda_i^2 K_d^2 \, \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}} \mathbb{E}_{\tilde{\mathbf{x}} \sim p_{\lambda_i}} \left[D_{\text{KL}}(\sigma(\epsilon) \parallel \sigma(\epsilon_\theta(\tilde{\mathbf{x}}_d))) \right] \tag{4.12}$$

Anomaly Detection via GNSM-based MSMA

Once a network is trained with the denoising objective in Equation 4.11, we can plug the scores into MSMA to identify anomalies. For a given point x, I compute the score estimates for all noise perturbation levels. The resulting vector represents the L-dimensional multiscale embedding space:

$$\eta(x) = \left(||s_\theta(x, \lambda_1)||_2^2, ..., ||s_\theta(x, \lambda_L)||_2^2 \right) \tag{4.13}$$

where $s_\theta(x, \lambda_i)$ is the noise conditioned score network estimating $\nabla_x \log p_{\lambda_i}(x)$. Following the mechanism laid out in Chapter 3, our goal is to learn "areas of *concentration*" of the inlier data in the L-dimensional embedding space ($\eta(x)$, for $x \sim p$). Concretely, I train a GMM on $\eta(X_{\text{IN}})$, where X_{IN} represents the set of inliers. At inference time, I first use the score network to compute the score-embedding space $\eta(x)$ for the test samples and then compute the likelihoods of the scores via the trained GMM. The negative of this likelihood is then assumed as the anomaly score for the test samples.

4.2 What Makes GNSM Appropriate for Anomaly Detection?

This section will elaborate on the need for GNSM as an anomaly detector. Next, I will demonstrate the use of GNSM combined with MSMA as a promising methodology for anomaly detection in tabular data, which remains an unsolved problem [Pang et al., 2021; Ruff et al., 2021; Aggarwal, 2017].

Limitations of Current Models in Handling Categorical Data

The handling of categorical data in anomaly detection has been largely superficial in mainstream methodologies, where only a handful explicitly model categorical data types [Pang, Cao and Chen, 2021]. Recent comprehensive benchmarks, such as the one by [Han et al., 2022], reveal a notable gap: none of the tested methods utilize categorical data's intrinsic properties. Traditional approaches convert categorical variables into one-hot or binary encodings, subsequently treating them as if they were independent, continuous variables. This representation is fundamentally flawed as it ignores the exclusive nature of categorical variables, where the presence of one class implicitly denotes the absence of others within the same category.

This oversight in existing models highlights a significant opportunity for advancement. Drawing a parallel from the evolution in deep learning for computer vision, the success of seminal convolutional neural networks such as [Krizhevsky, Sutskever and Hinton, 2017; Simonyan and Zisserman, 2015] is attributed to their inductive bias – the assumption that neighboring pixels in an image are correlated. This understanding led to the development of convolutional kernels, which are well-suited for image data.

In a similar vein, for categorical data, a more fitting approach is to model each category as a probability distribution, acknowledging the inherent dependence among classes within a category. While classes within a category are dependent, it is reasonable to consider different categories as independent.

GNSM addresses this gap by treating each categorical variable distinctly. Internally, it employs a Gumbel-Softmax relaxation for each one-hot encoded vector. The model computes loss on a per-category basis, thereby respecting and leveraging the inter-class dependencies within each category. This design choice reflects a significant inductive bias of GNSM, aligning closely with the inherent structure of categorical data. Furthermore, GNSM yields a straightforward approach to model mixed continuous/discrete features via estimating scores through a denoising objective..

Most Anomaly Detection Methods Require Labels

There is a dearth of unsupervised deep learning anomaly detection methods that excel on tabular datasets. For example, the otherwise exhaustive benchmark of [Han et al., 2022] reports only two unsupervised deep learning models, DSVDD [Ruff et al., 2018] and DAGMM [Zong et al., 2018], in their analysis; with both models being outperformed by shallow unsupervised methods. Some reconstruction-based autoencoder approaches have been proposed [Hawkins et al., 2002] but they require optimization tricks such as adaptive sampling, pretraining, and ensembling to work effectively [Chen, Sathe, Aggarwal and Turaga, 2017]. GNSM, combined with MSMA, offers a straightforward approach to model the data characteristics (i.e. the scores) and detect anomalies. Therefore, I argue that GNSM is a noteworthy addition to the class of unsupervised anomaly detection methods.

GNSM Advances Categorical Score Matching: Denoising and Beyond

Finally, I emphasize that my method introduces a streamlined approach to estimate scores for categorical data using denoising score matching. Recently, [Sun et al., 2023] proposed a ratio matching objective, which may be viewed as a discrete analogue to score matching with continuous variables. However, this method mandates the parameterization of conditional densities, necessitating a crafted architecture to mask specific input segments. In contrast, my method sidesteps such complexities, and can fit into any established score matching framework. For example, my method is compatible with alternative (non-denoising) score matching objectives such as sliced-score matching [Song, Garg, Shi and Ermon, 2020], or the implicit score matching objective originally proposed by [Hyvärinen, 2005].

Further, there exists a link between score matching and diffusion models as established by [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020]. Indeed, recent works such as [Austin et al., 2021; Hoogeboom et al., 2021] model categorical distributions through a diffusion process. However, it is important to note that these generative models eschew the estimation of the score function $s(x) = \nabla_x \log p(x)$. Instead, they incorporate the Markov chain interpretation of diffusion models, and directly predict the parameters for transition kernels. As a consequence, these models are not directly suitable for a spectrum of score-based applications, such as out-of-distribution detection as explored in my research, or hypothesis testing as introduced by [Wu et al., 2022]. It is plausible that forthcoming research will unveil further applications of score functions, wherein our methodology stands ready to extend these findings to categorical data.

Dataset	# Samples	# Anomalies	# Features
Bank	36548	4640	53
Census	280717	18568	396 (+5 cont.)
Chess	28029	27	40
CMC	1444	29	25
Probe	60593	4166	67
Solar	1023	43	41
U2R	60593	228	40
Nursery	4648	328	26

Table 4.1: Statistics of public tabular datasets commonly used for evaluating anomaly detectors. All datasets other than Census are categorical only.

4.3 Experiments on Tabular Benchmark Datasets

This section will quantitatively assess the performance of GNSM compared to baselines. I created an experimental testbed featuring categorical anomaly detection datasets sourced from a publicly available curated repository[2]. Note that for my method, I need to know the number of outcomes for each category to appropriately compute the softmax over all classes. This requirement prevents me from using preprocessed datasets, such as those made available by [Han et al., 2022]. It is also the reason why I could not use all the datasets in the curated repository, as some had been pre-binarized and do not provide further details about the number of classes.

The collected datasets are first split into inliers and outliers. Next, I divided the inliers into an 80/10/10 split for train, validation, and test respectively. The validation set is used for early stopping and the checkpoint with the best validation loss is used for inference. The test set is combined with the outliers and used for assessing performance. The categorical features were first converted to one-hot vectors and then passed through a log transform to retrieve logits. I used standard normalization to normalize any continuous features (only relevant for Census). I report detection performance across five runs with different seeds.

I chose four methods to represent baseline performance in lieu of a comprehensive analysis with multiple methods. I was inspired to go this route due to the thorough results reported by ADBench [Han et al., 2022]. As the authors describe, no one method outperforms the rest. I picked two representatives for shallow unsupervised methods: Isolation Forests and ECOD. I picked these as they consistently give good performance across different datasets and require little to no hyper parameter tuning. There are much fewer options for unsupervised deep learning methods that have been shown to work on tabular datasets. I chose

[2] https://sites.google.com/site/gspangsite/sourcecode/categoricaldata

two models that are popular in this field: DAGMM and DSVDD. Note that these were the only unsupervised deep learning models reported by [Han et al., 2022].

For my score network, I used a ResNet-like architecture inspired by [Gorishniy et al., 2021]. I replaced `BatchNorm` layers with `LayerNorm` and set `Dropout` to zero. The dimensions of the `Linear` layers in each block were set to 1024. All activations were set to `GELU`([Hendrycks and Gimpel, 2016]) except for the final layer, which was set to `LeakyReLU`. The number of residual blocks was set to 20. To condition the model on the noise scales, I added a noise embedding layer similar to those used in diffusion models [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020]. I used the same architecture across all datasets.

My noise parameter λ is a geometric sequence from $\lambda = 2$ to $\lambda = 20$. Early testing showed that the models gave numerical issues for values lower than 2. For the upper-limit (i.e. the largest noise scale) we chose 20 as it works well to smooth out the probabilities to uniform across all datasets. We set the number of noise scales (L) to 20. We compute the score norms on the inliers (train+val) according to 4.13 and train a GMM on the resulting features. The negative likelihoods computed from the GMM are the final outputs of my method.

Results

Dataset	Ano Ratio	IForests	ECOD	DAGMM	DSVDD	GNSM (Ours)
Bank	0.56	63.24 ± 1.74	$\mathbf{66.52 \pm 0.57}$	57.62 ± 3.36	58.50 ± 5.30	65.58 ± 3.45
Census	0.40	40.64 ± 2.07	40.96 ± 0.15	32.90 ± 5.00	41.18 ± 3.44	$\mathbf{47.79 \pm 2.29}$
Chess	0.01	$\mathbf{2.31 \pm 1.36}$	1.43 ± 0.05	1.08 ± 0.44	1.47 ± 0.54	1.60 ± 0.68
CMC	0.17	22.72 ± 1.57	23.79 ± 1.75	24.99 ± 5.75	21.99 ± 6.15	$\mathbf{25.87 \pm 9.93}$
Probe	0.41	92.95 ± 2.28	95.39 ± 0.38	66.40 ± 9.43	89.16 ± 8.40	$\mathbf{97.48 \pm 0.62}$
Solar	0.30	67.99 ± 3.48	$\mathbf{72.23 \pm 0.91}$	50.84 ± 5.19	51.21 ± 3.94	69.28 ± 1.96
U2R	0.04	52.74 ± 12.88	67.84 ± 1.39	10.06 ± 6.47	71.17 ± 24.65	$\mathbf{82.35 \pm 5.45}$
Nursery	0.43	46.51 ± 6.52	$\mathbf{100.00 \pm 0.00}$	48.33 ± 8.64	$\mathbf{100.00 \pm 0.00}$	$\mathbf{100.00 \pm 0.01}$
Average	-	48.64 ± 3.99	58.52 ± 0.74	36.52 ± 5.53	54.33 ± 7.49	$\mathbf{61.24 \pm 3.05}$

Table 4.2: Average Precision across multiple datasets. Higher is better. Each experiment was repeated with 5 different seeds and we report the mean and standard deviations across seeds. IForest and ECOD represent shallow models, while DAGMM and DSVDD represent deep learning models. Ano ratio refers to the ratio of anomalies in the test set.

Table 4.2 reports the Average Precision error (AP) which can also be interpreted as the Area Under the Precision Recall curve (AUPR). Average precision computes the mean precision over all possible detection thresholds. I chose to highlight AP over AUROC as it is a more apt measure for detecting anomalies, where one often has unbalanced classes. Additionally, precision measures the positive predictive value of

a classification i.e. the true positive rate. This is a particularly informative measure for anomaly detection algorithms where one is preferentially interested in the performance over one class (outliers) than the other (inliers). I would also like to note that the anomaly ratios in the test set do not correspond with the true anomaly ratio in the original dataset. This is due to my data splitting scheme where my test set is effectively only 10% the size of inliers.

Table 4.2 shows that GNSM-based MSMA performs better or on par with baselines. GNSM achieves significant performance improvements over baselines for Census, Probe and U2R, respectively achieving a 6.61%, 2.09%, and 11.18% improvement over the next best method respectively.

Results for CMC and Bank are less straightforward to interpret as the differences in the models are not statistically significant, made apparent by the large overlap in the standard deviations. This is especially true for deep learning models which have to be optimized via gradient descent. On Solar, ECOD outperforms the rest by a significant margin. However, among the deep learning models, GNSM performs notably better. Note that Solar is the smallest dataset in this benchmark, with less than 800 training samples.

Lastly, every model struggled with Chess, quite possibly due to the exceptionally small anomaly ratio. While Isolation Forests achieves the highest mean, it is uncertain whether the win is statistically significant. One could easily opt in favor of the other methods for this dataset as they achieve more consistent results. Again, among deep learning models, GNSM performs better.

Overall, we observe that shallow models give more stable and consistent results, with ECOD having the smallest standard deviations on average. Additionally, note that all algorithms struggle on this benchmark. This behavior is prevalent in the field of unsupervised anomaly detection methods, where models exhibit a large variance in accuracy across datasets [Han et al., 2022]. As such, in my testing, no one method definitively outperforms the rest; an outcome that coincides with previous findings of [Han et al., 2022; Pang et al., 2021; Ruff et al., 2021].

In this context, I emphasize that GNSM consistently ranks high across all datasets tested. In contrast, each of the competing methods were the top performer in only one of the datasets in Table 4.2, and significantly underperformed in others. Averaged over all datasets, GNSM performs best. This is empirical confirmation that GNSM is a consistent contender in the suite of available algorithms for practitioners looking to detect anomalies in unlabeled data domains.

4.4 Case Study: Detecting Segmentation Failures

I designed this case study to demonstrate how GNSM may be applied to a real world use case of detecting anomalous segmentation masks. The task here is to learn the distribution of ground truth image-segmentation pairs. At inference time, GNSM-MSMA will score the outputs of a pretrained segmentation model. My hypothesis is that GNSM will correctly detect failure cases i.e. poor segmentations will be ranked higher on the anomaly scale.

While there are many ways to qualitatively define a failure, I will be using popular segmentation metrics (with respect to ground truth masks) as a proxy for performance. I posit that a useful anomaly score should correlate meaningfully with the ground truth segmentation accuracies.

I compare the anomaly scores against three common segmentation metrics: the Dice similarity coefficient (Dice), the mean surface distance (MSD), and the 95-th percentile Hausdorff distance (95-HD). I chose Dice as it is a popular segmentation metric that measures the overlap between the predicted masks and the ground truth. However, as Dice scores may overestimate performance, it is recommended to additionally report distance based metrics [Valentini et al., 2014; Taha and Hanbury, 2015]. These metrics compute the distance between the surfaces of the predictions and ground truth masks.

I train a convolutional noise conditioned score network on the train-set of the Pascal-VOC segmentation dataset [Everingham et al., 2010]. The model takes in a pairs of images and the one-hot segmentation masks. The model predicts the scores for the segmentation masks only. I chose to use paired data rather than segmentations alone as I want the model to learn whether a segmentation is appropriate for the *given* image.

As my test subject, I retrieved a pretrained `DeepLabV3 MobileNet (V3 Large)` segmentation model [Chen, Papandreou, Schroff and Adam, 2017] from the publicly available PyTorch implementation [3]. This model was trained on a subset of the COCO dataset [Lin et al., 2014], using only the 20 categories that are present in the Pascal VOC dataset. I used the validation set of Pascal VOC as the test set.

I compare the performance of GNSM to a convolutional DSVDD [Ruff et al., 2018]. While there may exist specialized segmentation uncertainty estimators, I argue that an unsupervised model provides a more apt comparison. It is reasonable to postulate that both GNSM and DSVDD could be improved by additionally incorporating segmentation-specific objectives into the training, but that remains outside the scope of this study.

[3] https://pytorch.org/vision/stable//models/deeplabv3.html

For my score matching network, I adopted the NCSN++ model used by [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020]. The only significant change was in the input/output layers as I am predicting scores over one-hot segmentation masks. For DSVDD, I used the implementation of the original authors [Ruff et al., 2018]. To keep a fair comparison, we modified the code to use a modern architecture as the backbone (specifically EfficentNetV2 [Tan and Le, 2021]) and kept the number of parameters similar to our model. Both models were trained to convergence and the best checkpoints (tested over a validation split of the train-set) were used for the analysis.

Detection Results

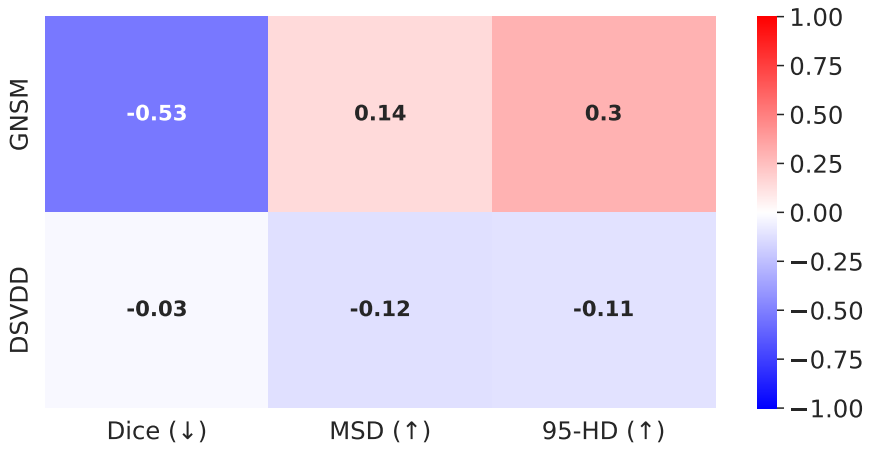


Figure 4.1: Correlations with segmentation metrics for Top-$K = 50$ anomaly scores retrieved from GNSM and Deep SVDD. The arrows next to the metric denote the expected correlation direction. The magnitude of the correlations reflects how well the anomaly scores capture segmentation errors.

I computed the anomaly scores from both GNSM and DSVDD and ranked the images from most to least anomalous. Next, I took the top $K = 50$ images (out of 1449) and computed the Pearson correlation coefficients between the ground truth segmentation metrics and the anomaly scores. I chose the worst ranked images for our analysis as we are interested in the efficacy of these scores for identifying segmentation failures as opposed to assessing the quality of successful segmentations.

Figure 4.1 shows the correlations between the ground truth segmentation metrics and the anomaly scores from GNSM and DSVDD. Recall that Dice is a similarity metric while MSD and 95-HD are both distance-based metrics. Therefore, I initially hypothesized that a good anomaly score should correlate negatively with Dice and positively with the distances. Our results show that GNSM correlates strongly in the direction

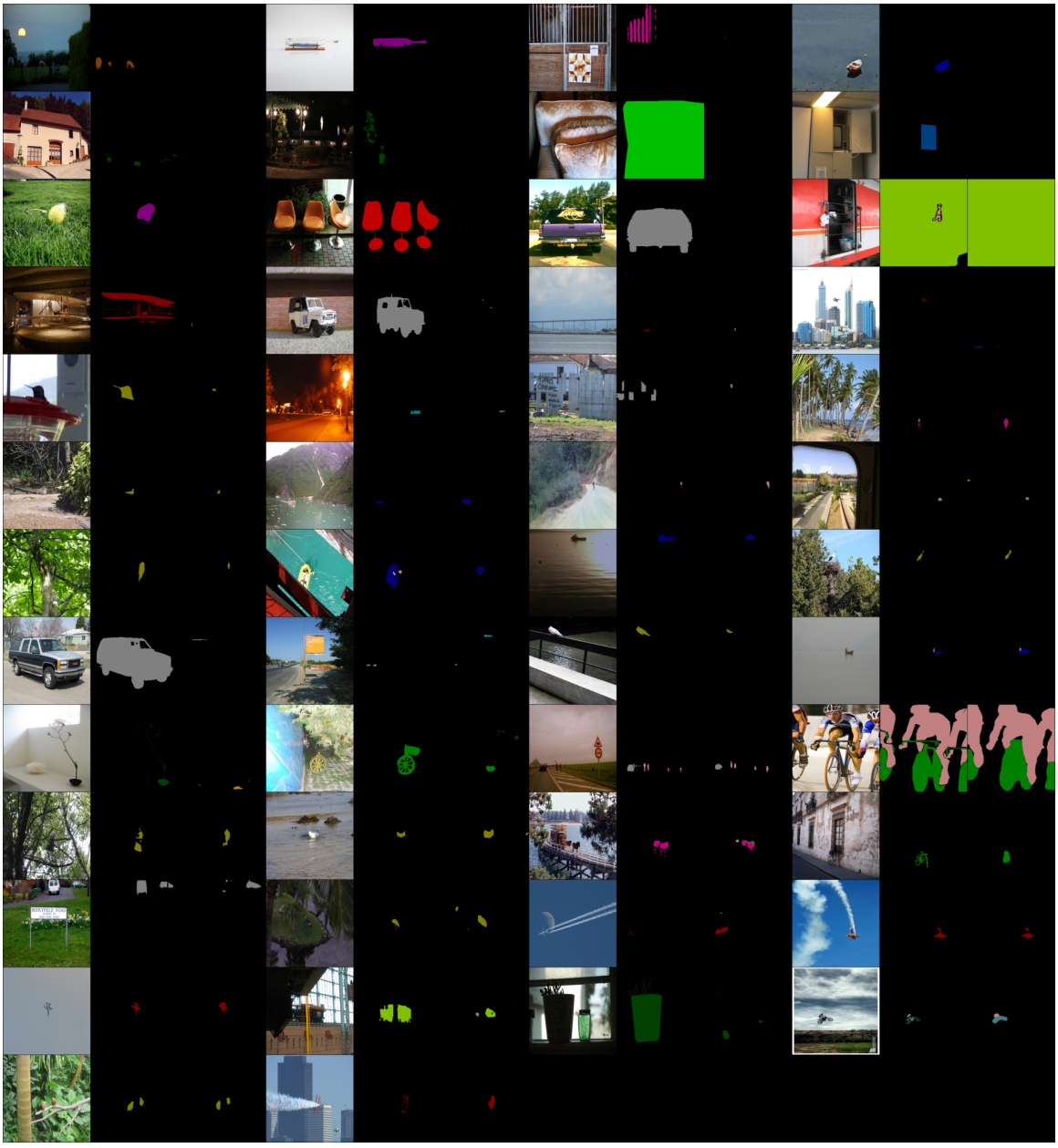


Figure 4.2: Samples from Top-K=50 GNSM rankings. The columns (repeated) show input image, ground truth segmentations, and model predictions respectively.

Figure 4.3: Random samples from Top-K=50 DSVDD rankings. The columns (repeated) show input image, ground truth segmentations, and model predictions respectively.

expected. DSVDD on the other hand achieved a poor correlation with Dice and inverse correlations with the distance based metrics.

To qualitatively assess the results of each model a subset of the worst ranked predictions are plotted alongside the groundtruth in Figures 4.2,4.3. The figures show predictions that were ranked to be anomalous by GNSM 4.2 and DSVDD 4.3. Images are displayed in order from highest ranking to lowest (displayed left to right).

Observe how the predictions ranked by GNSM in Figure 4.2 are either complete failures (most of the image is designated the background class) or severe under-segmentations. Predictions ranked by DSVDD in Figure 4.3 do not exhibit any obvious pattern of segmentation failures, with most being reasonable predictions.

I believe these results not only exemplify GNSM's generalization capabilities to non-tabular data, but also highlight a practical application. Quantifying segmentation uncertainties is useful when deploying off-the-shelf models. My experiments prove that GNSM-MSMA may be employed as a filtering mechanism to automatically detect poor segmentations, which could then be reviewed further downstream.

4.5 Limitations

Early testing showed that my score networks need to be deep and require more parameters than baselines. Although my proposed network size is performant, I observed a trend of increased performance as the models got deeper and wider. Due to time and resource constraints, I could not thoroughly explored the architecture space. Additionally, my models take a significant number of iterations to converge. For my experiments, I trained for 1 million iterations, which can take up to a day of training. This is admittedly in contrast to the baselines which may take a few seconds for shallow models and up to a few hours for the deep learning models.

I also acknowledge that GNSM explicitly needs to know the number of outcomes per category to appropriately add noise and compute the scores. While this is a strength of my approach, it does make for an overhead on the user's part. The baselines do not require this additional modeling complexity and are more straightforward to apply. Lastly, my method has hyperparameters pertaining to noise such as the number of scales used and the range of noise levels. While GNSM's hyperparameters have been proven to be stable, I posit that additional improvements may be obtained if these were also tuned per dataset.

4.6 Conclusion

This chapter introduced Gumbel Noise Score Matching (GNSM): a novel method for learning scores of categorical data types. Section 4.1 outlines how to compute scores of continuously relaxed categorical data and derive the appropriate training objective based on denoising score matching. Section 4.2 demonstrates how the estimated scores could be used for anomaly detection when plugged into MSMA.

GNSM achieves competitive performance with respect to baselines on a suite of tabular anomaly detection datasets, attaining significant improvements on certain datasets. Furthermore, GNSM can easily be extended to images and excels on the real-world task of detecting anomalous segmentations, as highlighted in Section 4.4. Lastly, I believe this categorical score matching formulation could be incorporated into generative models. I hope this direction may be explored in future work.

CHAPTER 5: LOCALIZING ANOMALIES VIA SPATIAL-MSMA

In the preceding chapters, we have discussed sample-wise anomaly detection. These methods operate on the full input and make binary predictions on whether a sample is anomalous. They do not, however, provide information about the input components that led to that assessment. It is often desirable to identify the specific regions within an image that are contributing to its atypicality. Such localization allows for increased model interpretability as well as directing future investigation.

For instance, in healthcare, the ability to interpret a model's prediction empowers medical practitioners to visually corroborate the identified regions of interest. Interpretation may also pave the way for novel insights

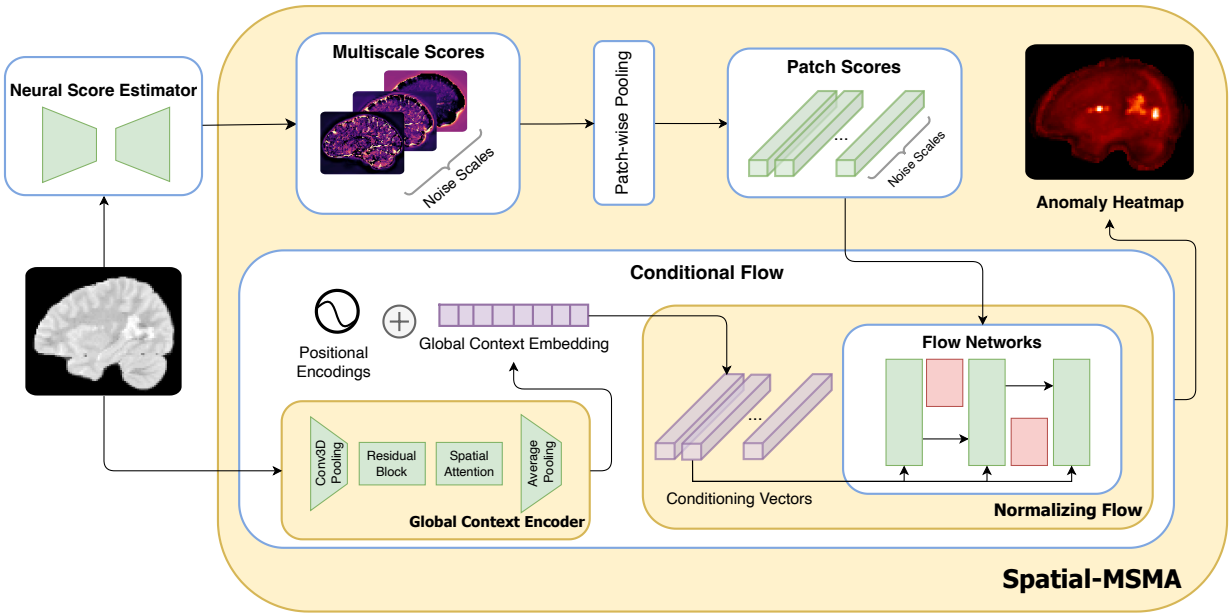


Figure 5.1: A schematic overview of Spatial-MSMA. A neural score estimator produces score tensors at multiple noise scales. The score tensors are divided into patches and processed by a conditional flow to estimate patch-wise anomaly scores. Global image features are extracted by a convolutional network and combined with positional encodings corresponding to each patch location, resulting in a conditioning vector per patch. The patch score norms and conditioning vectors are fed into a normalizing flow model with conditional coupling blocks. The result is a negative likelihood heatmap that highlights anomalous patches within the image. Spatial-MSMA thus enables precise localization of anomalies based on the patch scores and their spatial context.

into the disease. Furthermore, localizing anomalies enables targeted diagnosis and intervention planning based on the factors contributing to the detected outlier.

Another industry where anomaly detection has critical real-world value, is manufacturing [Bergmann et al., 2019]. Supply chains often need to automate defect identification to reduce operational costs and time requirements. This is typically accomplished through image-based anomaly detection. As the type of defects that can emerge in production are unknown, these systems utilize unsupervised models to enable broad detection. The models are tasked to output pixel-level segmentations to help manufacturers design solutions to remediate the defects. Although many unsupervised anomaly localization methods have been proposed in recent years, it still remains an active area of research [Liu et al., 2024].

This chapter will introduce Spatial-MSMA: an extended version of MSMA with localization capabilities. Note that the benefits of localization are twofold. First, it allows the possibility of pinpointing physical anomalies. For example, Spatial-MSMA is capable of highlighting brain lesions even though it was never trained with labeled data. Second, localization enables exploratory insights, gained through the interpretation of heatmaps. When taken together, these benefits significantly expand the use-case for MSMA. Related code is available at `https://github.com/ahsanMah/sade/`

5.1 Existing Techniques for Anomaly Localization

Reconstruction based Approaches

Reconstruction-based anomaly detectors are trained to produce typical counterparts (so-called reconstructions) of anomalous images. The methods may take some form of a deep autoencoder [Kascenas, Pugeault and O'Neil, 2022; Baur et al., 2021], trained with a reconstruction error objective such as mean squared error. At test time, the models are presumed to output an anomaly-free image, with the *reconstruction error* as the metric of atypicality. A known drawback of these models is the lack of specificity in their detection. As no reconstruction is pixel-perfect (especially in terms of image intensities), the output error maps have significant false-positives [Baur et al., 2021]. Another drawback of autoencoders is that as their reconstruction abilities improve, their anomaly detection capabilities decrease as the models are better at reconstructing the anomalies.

Generative Modeling based Approaches

While reconstruction-based methods directly learn to reproduce typical samples, generative modeling approaches learn the underlying data distribution. These models can then use iterative sampling techniques to gradually modify the input image into a in-distribution sample. There are two main methods proposed to employ generative models for removing anomalies: *imputation-based* such as [Liu et al., 2023] or *restoration-based* such as [You et al., 2019].

Unlike reconstruction-based approaches that attempt to recreate the entire input, imputation-based methods utilize masking strategies to selectively reconstruct certain regions of the image. Multiple passes are used to mask out different regions of the image. By focusing the network on only the unmasked regions, this approach can potentially lead to more precise anomaly localization. Restoration-based approaches, on the other hand, modify the entire image using the original image as the starting point in the sampling procedure, which can be more effective for global anomalies. The key advantage of generative modeling approaches over reconstruction-based methods is their ability to capture the probabilistic nature of the data distribution. This allows them to potentially handle a wider range of anomalies and provide more nuanced anomaly scores. However, they can be more computationally intensive and may require more complex training procedures compared to simpler reconstruction-based autoencoders.

Recently, owing to the success of score-based diffusion models, much of the research has focused on using diffusion models as the generative model (replacing GANs of yesteryear) [Wyatt et al., 2022; Pinaya et al., 2022; Liu et al., 2023; Behrendt et al., 2023]. These models provide slight modifications to the diffusion sampling process, starting the generation from an input image rather than random noise. The sampling process in these generative models typically involves adding noise to the input image and iteratively denoising the sample to generate an anomaly-free counterpart of the original image. Once the cleaned sample is generated, a voxel-wise difference between the input and its anomaly-free counterpart is used as the anomaly score. This process differs from reconstruction-based methods, which directly output the reconstructed image in a single forward pass, without adding any noise to the input. The main differentiating factors between these generative modeling methods are the hyperparameters used for training and the sampling strategies used during inference, which can significantly impact their performance and computational efficiency.

Feature Embedding based Approaches

Some methods aim to detect anomalies in a learned embedding space. The feature embeddings are computed by a neural network trained on the typical samples. At test time, it is assumed that the model will output feature embeddings that are *close* to the feature embeddings of the training population if the sample is an inlier and *away* otherwise. A popular method in this category is the Student-Teacher architecture anomaly detector by [Bergmann et al., 2020]. In this setting, we have two models: a high parameter-count Teacher network and low parameter-count Student network. The student model is considered to be a "weaker" version of the Teacher, and is trained using neural network distillation techniques. It is assumed that the Student model will fail to generalize to unseen datasets, producing a discrepancy between the teacher's features and that of the student. This discrepancy is used to produce an anomaly score.

Attribution based Approaches

Certain anomaly detection techniques draw on the insights of interpretability research. The task is to identify features of the data that contribute to the model's output. The identified features are often assigned a score relative to their importance, as determined by the rules of the interpretation technique. Examples of such methods include SHAP [Lundberg and Lee, 2017], Saliency Maps [Simonyan, Vedaldi and Zisserman, 2013], and GradCAM [Selvaraju et al., 2017].

SHAP (SHapley Additive exPlanations) is a game-theoretic approach to explain the output of any machine learning model. SHAP values attribute the prediction of an instance to the different features, highlighting the positive or negative impact of each feature. Saliency Maps are a visualization technique that highlight areas of an input image that most influence the output of a network. The saliency map is computed by taking the gradient of the output with respect to the input image. Areas with high gradient values correspond to regions in the input that have a significant impact on the model's prediction. GradCAM (Gradient-weighted Class Activation Mapping) is an extension of saliency maps that computes gradients with respect to feature vectors of an image rather than the image itself.

All of the mentioned attribution-based approaches aim to identify the features or input regions that contribute most to the model's predictions. This information can be used for anomaly detection, as it can localize the patches that lead a model to classify an instance as an anomaly.

Situating Spatial-MSMA within Existing Works

Spatial-MSMA addresses key limitations of existing anomaly localization techniques. Unlike reconstruction-based approaches, it avoids the pitfall of decreasing detection capabilities as reconstruction quality improves. In contrast to generative modeling methods, Spatial-MSMA doesn't require complex sampling procedures or modifications to pretrained models. It surpasses feature embedding techniques by considering both local and global context, potentially leading to more nuanced anomaly detection. Unlike attribution-based methods that often require labeled data or focus solely on model interpretability, Spatial-MSMA offers unsupervised learning with built-in localization capabilities. By leveraging conditional likelihoods and spatial information, Spatial-MSMA is expected to provide more accurate and interpretable anomaly localization, making it particularly valuable for applications in healthcare and manufacturing where precise identification of atypical regions is crucial.

5.2 Spatial-MSMA: Incorporating Spatial Information into MSMA

The basic assumption of MSMA is that inliers will occupy distinct regions in the score-norm space. At test time, we ask the question: Does the given sample belong to the inliers? MSMA consequently estimates the likelihood of a sample belonging to the inlier region in the score-norm space. Up until this chapter we have looked at the data samples holistically, i.e. we considered the entire set of features available to us (e.g. all the pixels in an image).

However, MSMA is also amenable for *subsets* of features. For instance, we may divide an image into patches and consider the score-norms of each patch independently. Now, we can ask the question: Does this *patch* belong to the inliers? As before, MSMA will output a likelihood estimate of a test patch belonging to the inliers, but this time *only* considering information present at the given patch location.

It is possible to naively extend MSMA to consider patches. One can decompose the image into a regular grid, and train an independent MSMA model for each grid location. One may even reduce computational costs by running training/inference in parallel for each patch location. However, while this approach is straightforward, it leaves much room for improvement.

Image Patches are Not Independent of Each Other

Namely, we can leverage *spatial locality*: the notion that neighbouring image patches are highly correlated, and thus they should be assessed together for anomalies. Furthermore, even patches which are spatially apart may depend on each other. Consider an image of a face. Observing patches of the left eye gives us rich information about what we may observe in the location of the *right* eye, even if the location of the right eye is distal to the left. One can incorporate this information into the decision making process to reason about the typicality of a queried patch. For instance, observing a brown-colored right eye is typical. However, observing a brown colored right eye *given* a black-colored left eye, is atypical.

Modeling Conditional Likelihoods

Following the motivation above, one can employ a conditional model where in addition to the contents of a patch, its position and surrounding context are also taken into account. As such, I posit to use a conditional likelihood model as the basis of my patch-based anomaly detector.

Concretely, the model will be conditioned on the patch position and the image features. Let $s_p = \{s(x_p)\}_{i=1}^{L}$ be the multi-scale score tensor for a given patch x_p at location p, belonging to the image x. Let $h(x)$ are the feature vectors of the image x computed by a convolutional network h. I propose to estimate the conditional likelihood model $p(s_p|p, h(x))$. Both the flow model and the context-encoding convolutional network are trained via the Maximum Likelihood Estimation (MLE) objective. As this model will output likelihoods of score-norms for each patch conditioned on the surrounding spatial information, the model is called Spatial-MSMA.

Spatial-MSMA uses a flexible class of likelihood estimators called normalizing flows introduced in Chapter 2. The patch locations are modeled via sinusoidal positional embeddings, commonly used in Transformer models [Vaswani et al., 2017]. In order to capture global image context, the original image is passed through a convolutional network with a large receptive field. The resulting feature embeddings are concatenated with the positional embeddings and fed into the flow model as contextual information. Recall that the feature embeddings are learned, as the convolutional network is trainable and will be updated in every backwards pass. The positional embeddings are fixed and not trained.

5.3 Prototyping on 2D Images

This section focuses on the task of segmenting anomalies in 2D images due to the availability of benchmark datasets and pretained models. A controlled experiment like this allowed me to refine the model architecture and compare Spatial-MSMA's performance to existing methodologies.

Most existing methods are trained and tested on the MvTec anomaly detection dataset [Bergmann et al., 2019]. This dataset focuses on industrial inspection and comprises of a set of defect-free training images and a test set of images with various kinds of manufacturing defects. The test set includes high resolution segmentation maps which allow us to validate the performance of the anomaly detection model. Note that the difficulty of this task stems from its unsupervised nature. At training time, the model will only see typical images and it cannot assume anything about the defects at the testing stage.

Similar to MSMA, Spatial-MSMA is trained in two stages. First, a score matching model is trained on the images and then the weights are frozen. Next, a likelihood model is trained on the score norms of the images. However, unlike MSMA, in Spatial-MSMA the likelihood model is trained on score norms of *patches* rather than on score norms of entire images.

Experiment Details and Results

To train the score-matching model, I finetuned a publicly available checkpoint [1] of a score-based diffusion model trained on the CIFAR-10 dataset. Next, a deep normalizing flow model is trained on the score norms for each patch, where the patch size is a fixed hyper parameter. The flow model receives the positional embeddings and the feature representations of the original image as conditioning vectors. Once the flow model is trained, we can evaluate it at every patch location in the image using a sliding-window. The resulting image is a likelihood heatmap, which is inverted to get the anomaly heatmap (negative log-likelihood).

For this experiment, the score model was trained on the *Cables* class. To evaluate the anomaly detection performance, the results were compared to that of the original authors of the MvTec dataset [Bergmann et al., 2020]. I compute the per-region overlap metric (PRO-AUC) introduced by the same authors, which measures the overlap between connected components in the ground truth and the anomaly map for every threshold.

[1] `https://drive.google.com/file/d/1JInV8bPGy18QiIzZcS1iECGHCuXL6_Nz/view?usp=drive_link`

Even without any hyperparameter tuning, Spatial-MSMA is able to outperform the baseline by 10% for the same patch size on the Cables class (.741 PRO-AUC vs .671 PRO-AUC).

5.4 Case Study: Lesion Detection in Volumetric Brain MRIs

Following the encouraging results on the MvTec dataset, this section will consider the more challenging task of detecting anomalies in medical images. The purpose of this case study is to reflect a real world usecase: automatic detection and segmentation of pathologies. As this is a feasibility experiment, one needs to minimize confounding factors that can be introduced due to a distributional shift between the training and testing populations. Thus, the anomalies will be simulated on a held out inlier test set, ensuring that the introduced anomalies are the *primary* factor differentiating the test set from the inlier population.

Constructing a Healthy Population

Our inlying, healthy population will comprise of typically developing school-age children. We chose this cohort due to the availability of public datasets within this demographic. Speicifically, we retrieved data from two studies: the Adolescent Brain Cognitive Development (ABCD) Study [Casey et al., 2018], and the Human Connectome Project Development Study (HCP-D). Samples from these studies were preprocessed to remove any outliers. To keep the inlier cohort as nominal as possible, we used the Child Behavior Checklist (CBCL) [Achenbach, N.d.] scores as our filtering mechanism. This checklist assesses the behavior and emotional competencies of children. Children with behavioral problems tend to score high on this test. The data was then split into an 80/10/10 train/validation/test split. Our processing resulted in 1320 training, 165 validation, and 165 testing samples.

I used both T1-weighted and T2-weighted images. As the images are high-resolution 3D MRIs, they require a lot of GPU memory during training. In order to fit a batch size of 4 per GPU, the images were downsampled to a pixel spacing of 2mm isotropic. They were further cropped by the largest brain mask, computed from the training data. After some padding to make the images multiples of 2, the resulting 3D volume was of size 96x112x80.

Simulating Lesions

The anomalies were simulated using a lesion simulator tool [da S Senra Filho et al., 2019], available as the MSLesionSimulator extension [2] of the Slicer3D software package [Fedorov et al., 2012]. The lesion load parameter was set to 20 and the rest of the hyperparameters were kept at their default values. A post processing step was performed to enhance the lesion intensity by a factor of 1.5. The lesions were generated on the test set.

Training Details

The score-norms were retrieved from a diffusion model, using a 3D convolutional UNet-like architecture. The VESDE formulation was used, with 2000 timesteps. The minimum sigma was set to 0.06, which is the average standard deviation of the image intensities. This is done so that, at minimum, the model is able to capture the intensity variation of within an image. Following the suggestion of [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020], the maximum sigma was set to 545.0 which is the 99-th percentile of the pairwise distance in the training set. This is done to allow the largest noise distribution to maximally cover the support of the data distribution i.e. $p_{\sigma_{\text{max}}}(x) \approx \mathcal{N}(x|0, \sigma^2_{\text{max}} I)$. The model was trained for 1.5 million iterations, by which point the validation loss had started to flatten out. The batch size was doubled at roughly the half way point during training. This is a simple yet effective method proposed by [Smith, Kindermans and Le, 2018], to effectively anneal the learning rate without having to use a decay schedule. The authors also found that increasing the batch size also reduced the number of parameter updates required to reach the same test accuracies as strategies for decaying the learning rate.

During inference, the voxel-wise anomaly scores are first brain masked followed by thresholding. The threshold is determined for each sample by searching for the threshold that gives the lowest symmetric mean surface distance between the ground truth and the post-threshold segmentation. Searching for a threshold like this is common practice in evaluating anomaly detectors [Baur et al., 2019]. The segmentations are post-processed by removing connected components of size less than 3 voxels (using a connectivity of 1). The remaining segmentation mask is dilated via a disk of radius 1 as the structuring element. Note that this inference procedure is performed for all methods tested in the experiment.

[2] https://www.slicer.org/wiki/Documentation/Nightly/Modules/MSLesionSimulator

Baseline Methodologies

Spatial-MSMA was compared to a selection of models that encompass a broad range of anomaly detection methodologies that have been successfully used in the medical imaging field. Namely, the baselines represent reconstruction-based, generative-based, and interpretation-based methods.

For the reconstruction-based baseline, I chose an autoencoder model by [Luo et al., 2023] owing to its success on volumetric brain MRIs. The model uses a ResNet-like architecture is trained using a reconstruction objective based on a Mean Squared Error (MSE). The authors also provide a publicly available implementation. This method is denoted as AE in Table 5.1.

Two generative-model based approaches were also included in the comparison. First is an imputation-based approach inspired by [Liu et al., 2023] which uses a checkerboard mask to in-paint different regions of the image (denoted as Inpaint in Table 5.1). This method performs multiple runs of imputation, alternating the checkerboard pattern each time and computing the average error across all runs. Second is a restoration-based approach (denoted as Restoration in Table 5.1), which first adds noise to the image and then invokes the sampling procedure of the diffusion model to iteratively generate the restored counterpart. Following [Wyatt et al., 2022], the sampling procedure was initiated from 1/4th of the original timesteps. However, unlike [Wyatt et al., 2022], I did not use Simplex noise during training/inference as recent research has shown that it may not be necessary (and sometimes detrimental) [Kascenas et al., 2023]. Note that these technique are agnostic to the diffusion framework, which allows me to use the same diffusion model that was used as the backbone of Spatial-MSMA. This keeps the comparison fair and limits confounding factors as the diffusion model was trained to convergence and is able to generate realistic looking samples.

Lastly, GradCAM was included as a representative of attribution-based approaches. Specifically, I used Guided-GradCAM [Selvaraju et al., 2016] which combines saliency maps (with some modifications) and GradCAM to give superior results to vanilla GradCAM. The gradients were computed using the outputs of a non-spatial MSMA. Concretely, a GMM was trained on the whole-image score norms and the GradCAM gradients were computed using the negative likelihood estimates. This corresponds to computing voxel-wise attribution maps for an MSMA anomaly score. Thus, the method is denoted as GradCAM-MSMA in Table 5.1.

Segmentation Metrics for Analysis

Similar to Section 4.4, I chose mean surface distance (MSD), and the Hausdorff distance as the segmentation metrics to compare the results. These metrics compute the distance between the surfaces of the predictions and ground truth and are less biased towards over-segmentations compared to the more popular Dice score. Both distances are computed in a directed manner i.e. the distance is computed from the ground truth to the prediction. For Hausdorff distance, the 99-th percentile is used. In addition to distance metrics, component-wise metrics were also computed. Connected components were computed from the voxel-wise segmentation masks by considering an 8-connectivity-neighborhood (diagonals were included as neighbours). We assign a *true positive* (TP) label to a component in the prediction which overlaps with any component in the ground-truth at any voxel location. Conversely, the absence of any overlap is used to keep a tally of the number of *false positives* (FP). Table 5.1 reports the True Positive Rate (TPR = TP/(TP+FN)) and the Positive Predictive Value (PPV = TP/(TP+FP)).

	99-HD ↓	MSD ↓	TPR ↑	PPV ↑
AE [Luo et al., 2023]	12.27± 0.51	3.63± 0.35	0.44± 0.02	0.19± 0.01
Inpaint [Liu et al., 2023]	13.26± 0.50	3.71± 0.27	0.63± 0.02	0.50± 0.02
Restoration [Wyatt et al., 2022]	8.67± 0.53	2.68± 0.36	0.68± 0.02	0.17± 0.01
GradCAM-MSMA	12.68± 0.54	3.75± 0.37	0.43± 0.02	0.16± 0.01
Spatial-MSMA	**7.05 ± 0.61**	**2.10 ± 0.43**	**0.83 ± 0.01**	**0.96 ± 0.01**

Table 5.1: Segmentation metrics for lesion detection. Each model was trained only on the (same) inlier samples. Right column shows distance based metrics: 99th-percentile of the Hausdorff Distance (99-HD) and Mean Surface Distance (MSD). Right column shows component-wise metrics: True Positive Rate (TPR) and Positive Predictive Value (PPV). Spatial-MSMA significantly outperforms the baseline methodologies, especially for component-wise metrics.

Results

Table 5.1 reports the segmentation performance of all the methods tested. Note that due to the size of the lesions, the segmentation task was difficult for all models. However, Spatial-MSMA significantly outperforms the competition. Compared to baselines, Spatial-MSMA shows the lowest distance metrics. These metrics reflect the specificity of the model's segmentation capabilities. Lower distances imply that the models segmentations have tighter boundaries around the anomalies, compared to baselines. Conversely, the component-wise metrics reflect the sensitivity of the model to anomalous regions in the image, regardless of size. Spatial-MSMA shows excellent detection capabilities, evident by the 0.83 TPR as well as the

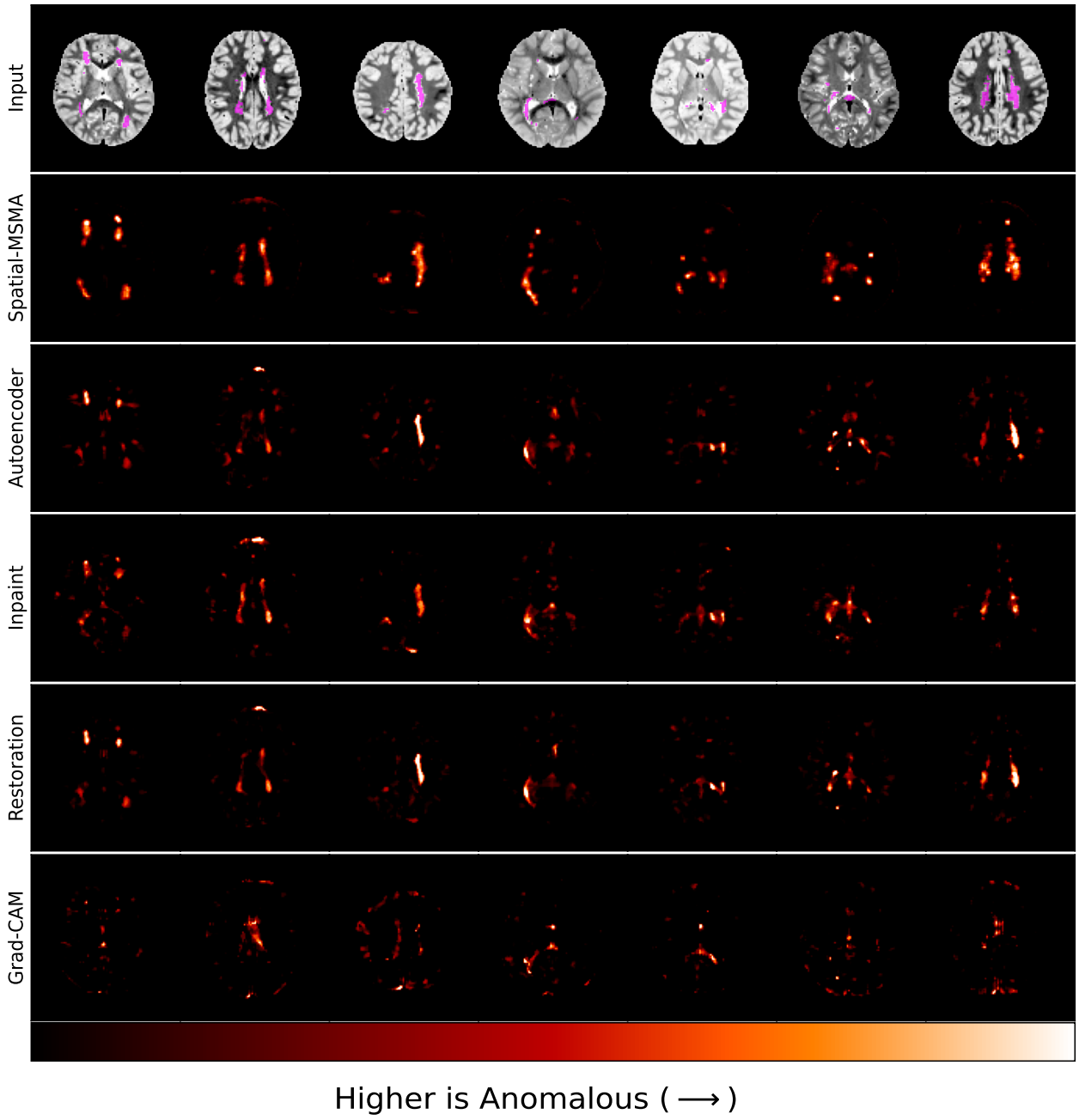


Figure 5.2: Qualitative comparison of anomaly heatmaps across different methods. The first row shows random axial slices of the volumetric input samples. The lesions are highlighted in magenta. Each column is a slice from random individuals. Note how Spatial-MSMA consistently detects all the lesions in the image, while other methods tend to miss smaller lesions.

exceptionally high PPV of 0.96 (recall that the maximum possible PPV is 1.0). This implies that Spatial-MSMA detects few false positives, an advantageous trait in anomaly detection. We can qualitatively observe the results in Figure 5.2. The plotted heatmaps are clipped at the 90th percentile for each sample i.e. the range represents the top-10% of the anomaly scores. Note that while Spatial-MSMA tends towards over-segmentation, it manages to detect most if not all the lesions. Other baselines such as Inpaint are overly biased towards larger anomalies, and often fail to detect smaller lesions.

5.5 Conclusion

This chapter introduced Spatial-MSMA: an extension to MSMA that enables localization of anomalies. The key insight is to use a conditional likelihood model to learn the distribution of patch score norms, conditioned on the patch location and surrounding context. Spatial-MSMA demonstrated superior performance in detecting and localizing simulated lesions in volumetric brain MRIs compared to several state-of-the-art baselines. The results show that Spatial-MSMA significantly outperformed existing methods across multiple metrics, including mean surface distance, Hausdorff distance, true positive rate, and positive predictive value. While Spatial-MSMA is not expected to outperform supervised networks trained for lesion segmentation, the model's ability to detect lesions of varying sizes while maintaining a low false positive rate highlights its potential as a powerful tool for unsupervised anomaly detection in medical imaging. This chapter focused on experimentally verifying the capabilities of Spatial-MSMA using ground-truth data. The next chapter will investigate the data exploration capabilities of this approach.

CHAPTER 6: DEMYSTIFYING NEURODIVERGENCE VIA SCORE ESTIMATORS

Given the success of MSMA for anomaly detection on benchmark experiments, we will now explore whether this methodology allows for any new *scientific* insights. Assuming the underlying model has learned rich information about the data, can it help us uncover useful knowledge about our dataset? I will try to answer that question by focusing on a narrow task: detecting neuroatypicalities from structural MRIs of pre-pubertal brains.

Neurodevelopmental disorders are disabilities associated primarily with the functioning of the brain. Certain disorders such as Down Syndrome, Fragile-X Syndrome and Angelman Syndrome can be detected via genetic testing. Others, such as Autism Spectrum Disorder (ASD) and Attention-Deficit/Hyperactivity Disorder (ADHD), are diagnosed through clinical assessments that measure behavioral response. While effective in diagnosis, neither genetic testing nor behavioral assessments provide insight into the differences in *brain morphometry* of the neurodivergent cohort. Furthermore, there is evidence for a heterogeneity in the underlying brain structures for certain brain disorders, such as ASD [Lenroot and Yeung, 2013]. This implies that the same behavior can originate from different atypical brain development.

Existing analyses primarily look at group wise differences of brain features (such as region volumes) between cohorts [Girault and Piven, 2020; Hamner et al., 2018], or correlations of said features with a behavioral assessment [Shen et al., 2022; Weerasekera et al., 2022]. These analyses forgo a broad study of all brain regions in favor of specific regions that are more relevant to the disorder. This is an understandable trade-off as reserarchers often have strong prior evidence that correlates brain regions to behavioral actions, and have limited time and resources. However, in making this trade-off, these analyses are less likely to uncover novel associations between brain regions and behavioral phenotypes (in the context of the neurodevelopmental disorder). Furthermore, the researchers introduce additional biases when they select regions of interest (ROIs) to be studied using their preconceived notions about the disorder.

Herein lies the opportunity for a data-driven methodology to uncover atypically developing ROIs that are reflected in the data but, may be underexplored in the existing literature. One can task a model to identify subgroups that deviate from the typical (via anomaly detection), and subsequently highlight the brain

morphometry shared across the subpopulations (via localization). MSMA, combined with its localization augmentation, naturally allows for such data exploration. This chapter outlines a process through which MSMA was utilized to derive insights about brain MRI data.

6.1 A Hypothesis Generating Tool

The previous chapters have demonstrated MSMA's anomaly detection capabilities. At the heart of MSMA is the score-norm vector. These vectors describe a space in which the outliers become separable from the inliers. How can one explore this space to gain further insights into the data?

We can assume that a user exploring the outputs of MSMA will be looking to answer a set of research questions. Examples of the questions may be:

- Do any atypical subpopulations cluster in the score-norm space?
- Can we identify atypical input features (such as brain ROIs) that are shared among samples within a cluster?
- Do atypical subpopulations in the score-norm space correlate with any non-morphometric measurements (such as behavioral assessments)?

To answer such questions, one needs to be able to link multiple sources of data: the score-norm space, the localization heatmaps, and existing metadata associated with each sample. I posit to use an interactive visualization to facilitate such a task. Using this visualization the user should be able to observe how samples are distributed/clustered in the score-norm space. Additional views can show the aggregated localization heatmaps for each cluster, as well as metadata such as behavioural scores.

6.2 Exploring a High-Dimensional Space

It is often desirable to visualize data for the purposes of exploratory research. This poses a problem if we want to visualize the space of score-norms (as a 2D image). While the score-norms are much lower in dimensionality compared to the high resolution MRIs they are derived from, each score-norm vector is still multidimensional. For reference, all of the experiments in this chapter use $L = 20$ scales, resulting in a 20-dimensional vector for each sample. One has to utilize a dimensionality reduction algorithm to reduce the space into two or three dimensions so that the points may be visualized on a screen.

This section will briefly go over existing dimensionality reduction algorithms and why they may not be appropriate for our analysis as they can introduce spurious clusters in the visualization. Instead, I will be advocating the use of Self-Organizing Maps (SOM) [Kohonen, 1990] to visualize score-norm data and will discuss its advantages.

Common Dimensionality Reduction Algorithms

Most commonly, researchers have used Principal Component Analysis (PCA) [Abdi and Williams, 2010]) to select the most important components in the data. PCA finds the eigenvectors of the correlation matrix of the data. Intuitively, these eigenvectors capture the largest variation in the data. While this technique is fast and well understood, it has the downside of only capturing *linear* subspaces as it will select orthogonal components of increasing variation. Therefore, it is most effective for visualization when the data itself is linearly separable. Recently, two other techniques have gained popularity as methods to visualize high dimensional non linear data: t-SNE and UMAP.

t-SNE (t-distributed Stochastic Neighbor Embeddings) defines the relative distance between two points as a conditional probability. It uses an optimization process to determine a low-dimensional embedding such that the relative distance between each point in the low-dimensional space matches the distance in the high-dimensional space (using the Kullback-Leibler Divergence between the pairwise conditional probability distribution).

UMAP (Uniform Manifold Approximation and Projection) computes the low-dimensional embedding in two stages. First, it constructs a weighted graph of the k-nearest neighbours (the number of neighbors being a hyperparameter). The weights of the graphs represent probability distributions. In the second stage, it uses an iterative procedure to pull the neighbors in the graph closer together in the low-dimensional space. Additionally, it applies a repulsive force between all non-neighbours in the graph, pushing them away in the low-dimensional space. Generally, UMAP is considered to have superior runtime performance compared to t-SNE [McInnes, Healy and Melville, 2020]

t-SNE and UMAP Do Not Reflect True Distances

Both t-SNE and UMAP aim to find a low-dimensional space that reflects the distances in the original space. However, the resulting low-dimensional space is highly dependant on the hyperparameters. These hyperparameters (associated with the size of the neighbourhood considered) produce a trade-off between

the preservation of local structure and preservation of global structure. Often, the algorithms are unable to preserve both [Wang et al., 2021]. If local structure is emphasized, the algorithms produce spurious clusters, making their results untrustworthy. Conversely, if global structure is emphasized, the algorithms tend to produce large homogenous blobs, and fail to distinguish any underlying patterns [Lei et al., 2023].

Self Organizing Maps as an Alternative

The Self-Organizing Map, also known as Kohonen Networks, [Kohonen, 1990] produces a two-dimensional grid-like map such that each cell on the map corresponds to a location in the data-space. Nearby cell locations on the grid are closer in the data-space compared to distal cell locations. A cell is commonly also referred to as a 'neuron'.

During training, the method randomly initializes a grid of neurons (cells). Note that the weight vector corresponding to each neuron resides in the data-space. For each data sample we find the closest neuron to the current point. This is referred to as the Best Matching Unit (BMU). The BMU's weights are updated to be closer to the data point. Additionally, the neighboring neurons (in the *grid-space*) are also updated. The latter step is crucial to preserve the topological structure of the data. This procedure is repeated for a pre-defined number of iterations. Most online packages implement the batch-learning algorithm described by [Kinouchi et al., 2002]. The authors improve convergence by initializing the neurons using the first two eigenvectors computed from the data, and by modifying learning objective to use the full dataset in each weight update rather than a minibatch as in the original algorithm. I will be using the latter training algorithm.

SOMs Circumvent Dimensionality Reduction

Note that SOMs operate directly in the high-dimensional data space, computing distances between the data points and the neuron prototypes in this original space. The topological constraint on the grid of neurons mainly facilitates the visual organization. This is in contrast to t-SNE and UMAP, which first project the data into a low-dimensional space, where distance computations may not accurately reflect the true distances in the high-dimensional space. As a result, SOMs can better preserve the structure and relationships within the data, reducing the risk of distortions introduced by dimensionality reduction.

SOMs Identify Prototypes

It is helpful to think of SOM as a vector quantization algorithm with a spatial constraint on the codebook of vectors. Thus, each neuron can be interpreted as a prototype in the data-space. The grid-like structure of SOMs provides a natural way to organize these data prototypes. The relative positions of the neurons on the grid reflect their similarities. This spatial organization can aid in identifying patterns, clusters, or relationships within the data that may not be immediately apparent from a scatter plot produced by techniques like t-SNE or UMAP.

To elucidate how one can use SOM to gain insights about the data, recall that each sample is assigned to a single neuron (the sample's BMU). If class labels are available, we can tally up the labels of all the samples that are assigned to a given neuron. The maximally matching label can then be computed for each neuron. This label is often overlayed on top of the map visualization to quickly associate the prototype with a population. Figure 6.1 shows an example of a SOM trained on the CIFAR10 score-norms. Overlayed are examples of prototypical BMUs for OOD datasets. The BMUs displayed are those that matched at least 1% of the samples belonging to the OOD dataset.

We can observe that ImageNet (orange stars) has the highest number of BMUs, which may be explained by the class overlap between ImageNet and CIFAR10. Both datasets comprise natural images, with ImageNet being a superset containing most of the classes present in CIFAR10. LSUN and iSUN, which consist of images of locations and scenes (such as bedrooms), have more overlap with each other than they do with CIFAR10. This similarity is reflected on the grid, where BMUs from LSUN and iSUN appear near the edge (as they are semantically dissimilar to CIFAR10) but neighboring one another (as they are semantically similar to each other). Finally, note the widespread distribution of SVHN BMUs. As this dataset consists of images of house numbers, it mostly comprises simple shapes like lines and circles. It is possible that these shapes are matching to prototypes that have accentuated shapes, rather than matching based on semantic similarities.

6.3 Case Study: Detecting Brain Regions involved in Down Syndrome

This case study will explore how MSMA can be used to derive scientific insights. We will focus our analysis on a particular neurodevelopmental disorder: Down Syndrome.

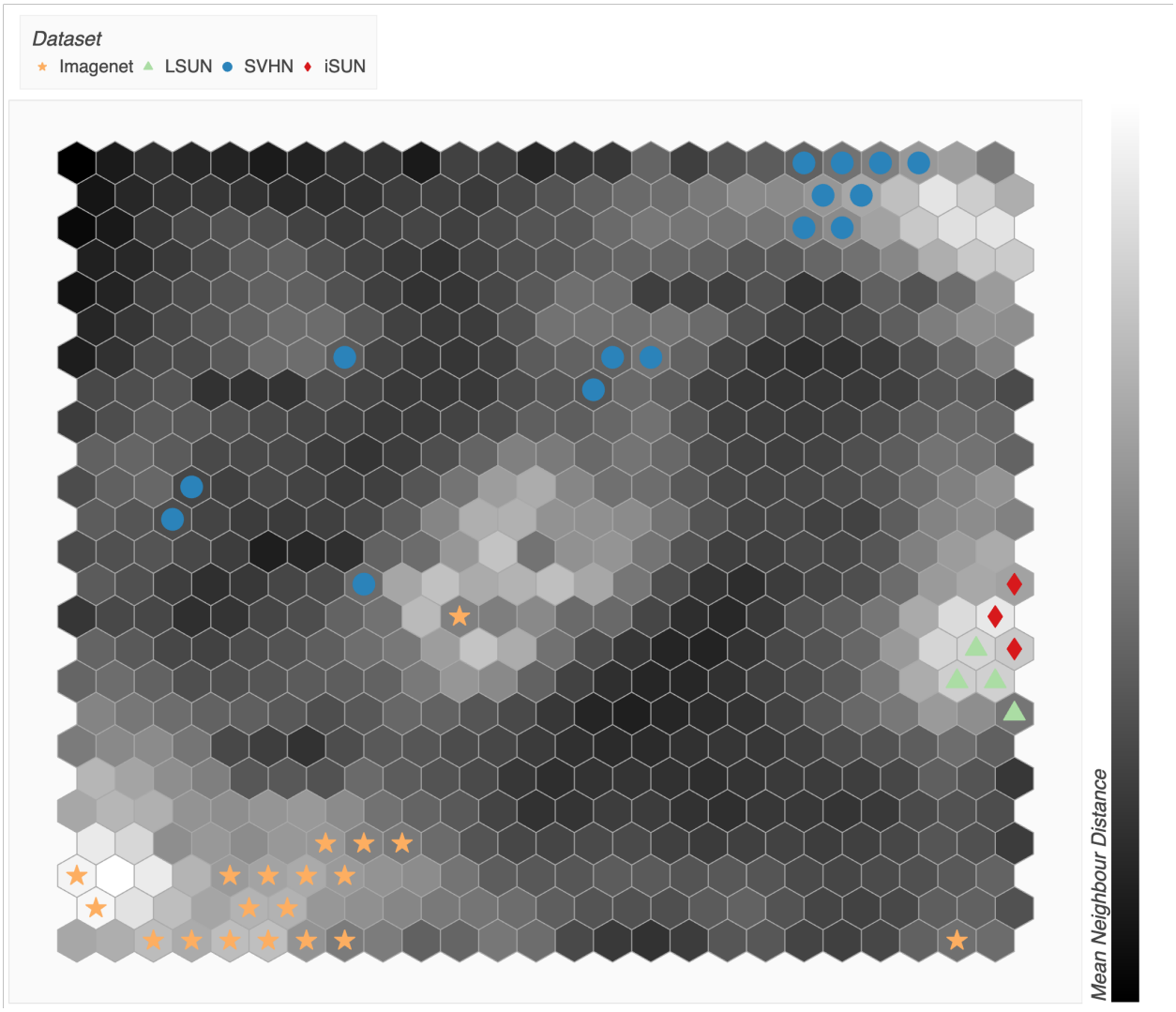


Figure 6.1: A 25x25 SOM trained on the score-norms of CIFAR10. The SOM grid maps the space of typicality, represented by prototypes in the inlying dataset. Overlayed are examples of BMUs that match with at least 1% of the datasets. Colored markers represent different datasets. Note how ImageNet (natural images) has the most BMUs. LSUN and iSUN appear together and at the edge of the map, as they are images of scenes (such as bedrooms) which have little overlap with CIFAR10. SVHN (images of numbers) has a wide spread of BMUs, which BMUs may be matching to accentuated shapes rather than semantic content.

Down Syndrome (DS) is one of the most common neurodevelopmental disorders, occurring at a rate of 1/700 live births [Parker et al., 2010]. Studies have revealed large deviations from typical development in brain structure of children with DS, with the most pronounced deviation in the total brain volume (TBV). Further, when TBV is accounted for, studies have revealed deviations in specific regions of the brain such as the temporal lobe, cerebellum, and hippocampus [Hamner et al., 2018]. However, there remains a push for developing focused research to elucidate the nature of DS neuroanatomic phenotypes [Hamner et al., 2018]. While individuals with DS present with a rather heterogeneous set of behavioral and cognitive challenges, existing neuroimaging studies investigate atypical brain morphometry in DS from the viewpoint of a single, homogeneous population.

This case study aims to uncover brain morphometric phenotypes that are associated with DS, using learned score-estimates. It will incorporate MSMA, SOMs, and Spatial-MSMA to provide a cohesive analysis of structural brain MRIs belonging to a DS cohort. We will be answering the following research questions:

- Does the score-norm space produced by MSMA *cluster* any atypical neuroanatomic phenotype(s) pertinent to DS?
- Can Spatial-MSMA *isolate* brain regions relevant to the phenotype?
- Do the regional anomaly scores *correlate* with any behavior assessment?

Data and Methodology

As in Chapter 5, we will be using samples from the ABCD and HCPD study as our reference inliers. I will denote this as ABCD-HCPD. This cohort was preprocessed to remove any outlying samples. To keep the inlier cohort as nominal as possible, we used the Child Behavior Checklist (CBCL) [Achenbach, N.d.] scores as our filtering mechanism. This checklist assesses the behavior and emotional competencies of children. Children with behavioral problems tend to score high on this test. For our analysis, all children that scored above a t-score of 66 (~ 95-th percentile) in the summary scores as well as *any* of the subscores were removed. Note that this is more conservative than only using the summary scales.

Our testing population is retrieved from the Infant Brain Imaging Study (IBIS) study. The goal of the study is to investigate brain development starting from infancy and following up at later ages. The study is primarily focused on investigating Autism Spectrum Disorder (ASD) but has also gathered data for Down's Syndrome. This section will be looking at school-age scans as they best overlap with the ages in the ABCD

study. IBIS uses a protocol to determine typically developing children, as a means of obtaining a control population. These control samples have to be non-ASD as well as low-likelihood (no older sibling with ASD). The control population was further filtered using Autism Diagnostic Observation Schedule (ADOS) [Lord et al., 2012] scores. Specifically, we removed samples that had an ADOS severity score above 2 measured at the age of 2 years.

In addition to the Down Syndrome population, I will consider three other cohorts as out-of-distribution datasets:, ASD, Atypical, and HR-Typical. ASD refers to individuals that have been diagnosed with the Autism Spectrum Disorder. Atypical refers to samples that scored above a threshold of 2 on the ADOS scores, and were thus labelled as outliers. The HR-Typical cohort includes individuals that are considered to have a high likelihood of developing autism, as they have an older sibling with diagnosed ASD. I include these cohorts to give us supplementary information about the phenotypic space that we will be observing.

A 3D diffusion model was trained on the ABCD-HCPD dataset. At inference time, the diffusion model was used to obtain score norms for the IBIS data as well as a held out test-set of ABCD-HCPD. A Spatial-MSMA model was also trained on the score-norms of the ABCD-HCPD samples. This allowed us to obtain the voxel-wise anomaly scores for each test sample. These anomaly scores were then averaged according to ROIs retrieved from the AAL atlas [Rolls et al., 2020]. Additionally, a SOM with a height of 7 nodes and width of 9 nodes was trained on the inlier score-norms from all three studies.

Does MSMA reveal a DS Prototype?

Intuitively, MSMA capture a feature space of atypicality. As such, different regions in the score-norm space may correspond to different manifestations of abnormalities. I argue that we can employ SOM to map the score-norm space and descretize it into phenotypes/prototypes of atypicality. Using SOM, it is possible to explore whether there exist sub-populations in the DS cohort that are differentially expressed in the score-norm space.

Recall that in SOMs, each sample is matched with one BMU (represented as a cell in the grid). Consider the BMU with the most matches for a particular cohort. I will denote this BMU as the *Maximal-Prototype* for that cohort. Figure 6.2 shows a SOM trained on the score-norms of the inliers. The heatmap reflects the neighbouring distances between each grid cell, and can be interpreted as the relative density of each region in the score-norm space. Cohorts are represented by different markers and colors. Each marker is sized according to the number of samples matching the prototype.

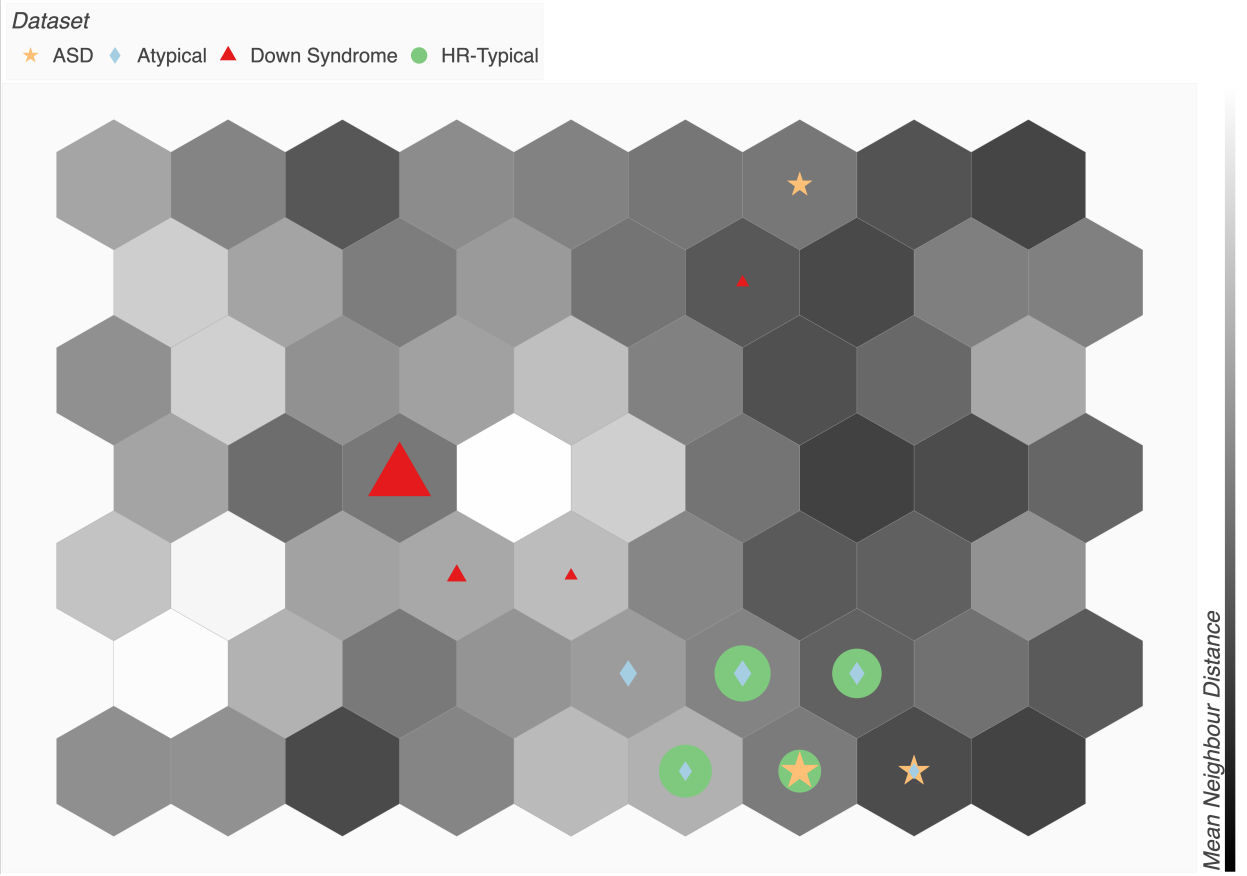


Figure 6.2: A Self-Organizing Map trained on score-norms of brain MRIs of typically developing children. The heatmap represents the distance between neighbouring grid cells Overlayed are the prototypes for different cohorts. The markers are scaled according to the number of samples matching the BMU at that location. Protoypes for Autism (ASD), high-risk inliers (HR-Typical), and outliers (Atypicals) are also displayed for reference. Markers are scaled by the number of matching samples. Note the Maximal-Prototype for DS is significantly larger than the rest.

We can observe a Maximal-Prototype for the DS cohort (largest red triangle in Figure 6.2). Out of 28 DS cases, 16 match this prototype. 12 out of the matching 16 are female, pointing to a female-dominant DS phenotype. Also plotted are prototypes for other cohorts, including autism (ASD), ADOS-atypicals (Atypical), and high-risk inliers (HR-Typical). We observe distinctive patterns in the aforementioned non-DS populations. There is a grouping of ASD, Atypicals, and HR-Typicals. Some of them share the same prototype, hence the overlapping of the markers such as ASD (star) and Atypical (diamond) on the bottom right. Note that these cohorts are determined by considering the likelihood of the individuals to fall on the autism spectrum. Thus, we can postulate that these prototypes reflect some underlying phenotypic similarities between the populations, and that these pheontypes may be relevant to ASD in some manner.

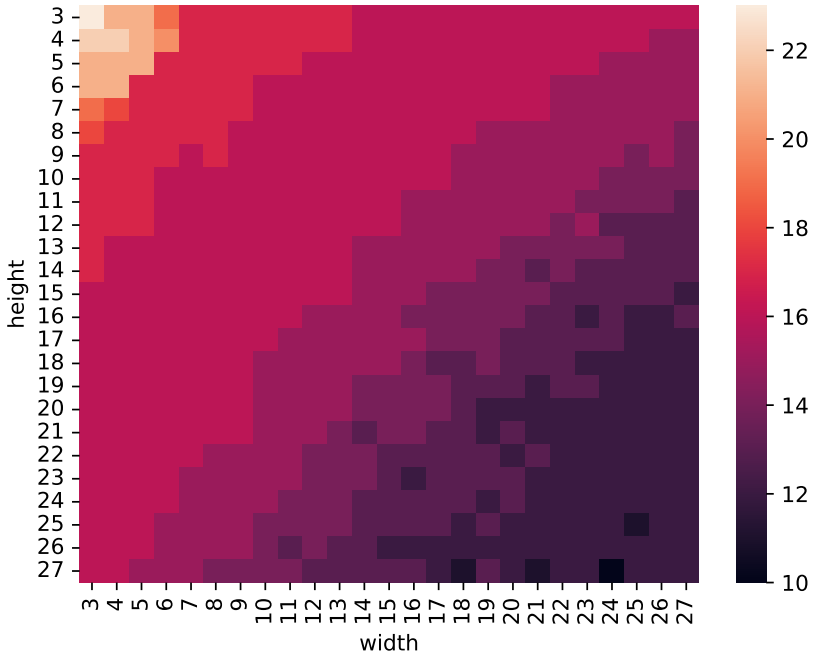


Figure 6.3: Hyperparameter analysis of SOM train on MSMA score-norms. The x-axis represent the width of the SOM grid, measured in number of neurons. The y-axis represents the height of the SOM grid. The heatmap shows the number of samples belonging to the Maximal-Prototype for each experiment. Note the diagonal 'stable' regions of the hyperparameter space.

Prototype Consistently Detected Across Experiments

When using a data exploration tool such as SOM for producing scientific insights, it is crucial to delineate true correlations that exist in the data and spurious correlations that may be caused by random variations in

the algorithm. Therefore, to increase our confidence in the findings, it is helpful to analyze the stability of the algorithm across runs with different hyperparameters. If the algorithm outputs consistent results, then one can have more confidence when interpreting the results. Note that as we are using PCA initialization, and thus each SOM experiment is deterministic and reproducible (given the same training set).

SOM has two main hyperparameters: the number of rows, and the number of columns in the grid. We can assess whether SOM consistently finds the same Maximal-Prototype for DS. Figure 6.3 shows a heatmap of the number of samples that matched the Maximal-Prototype in each experiment. We can observe that the number of matches remains consistent across different hyperparameters. Smaller grid sizes will have the most matches as there are only a few candidate prototypes. Large grid sizes will cause a more refined binning, resulting in multiple prototypes that are relatively close to one another. We can observe large "bands" of stability in the middle regions.

Further analysis revealed that each band captures the *same* samples as well as having a perfect overlap with the one before it. For instance, the set of experiments that produced a 16-sample prototype not only captured the same 16 samples within the set of experiments but also the same samples that were captured by the 15-sample prototype. This indicates that SOM is capturing a true sub-population of DS samples in the score-norm space rather than clustering at random.

Does Spatial-MSMA Capture Relevant Brain Regions?

Once a sub-population of the DS cohort is identified, we can begin to explore the brain regions that significantly differ from the rest. This analysis will use the anomaly scores produced by Spatial-MSMA to quantify the relevance of each region. The goal of the analysis is to determine whether there are regions that are significantly different between the prototype and non-prototype populations. To select anomalous ROIs, the anomaly scores (per region/ROI) were averaged across the DS cohort. Only those regions that had an average above the 90-th percentile with respect to the *inlier* population were retained for the significance analysis.

The filtered set of anomalous ROIs was used to determine whether there exists a distributional shift between the anomaly scores for the prototypical and the non-prototypical samples. Concretely, we will be using the Mann-Whitney U-test; a non-parametric test to compare the medians between two groups. This test is well-suited for our scenario as the sample distributions are long-tailed, and thus we cannot expect our populations to follow a Normal distribution. The null hypothesis of this test is that the two

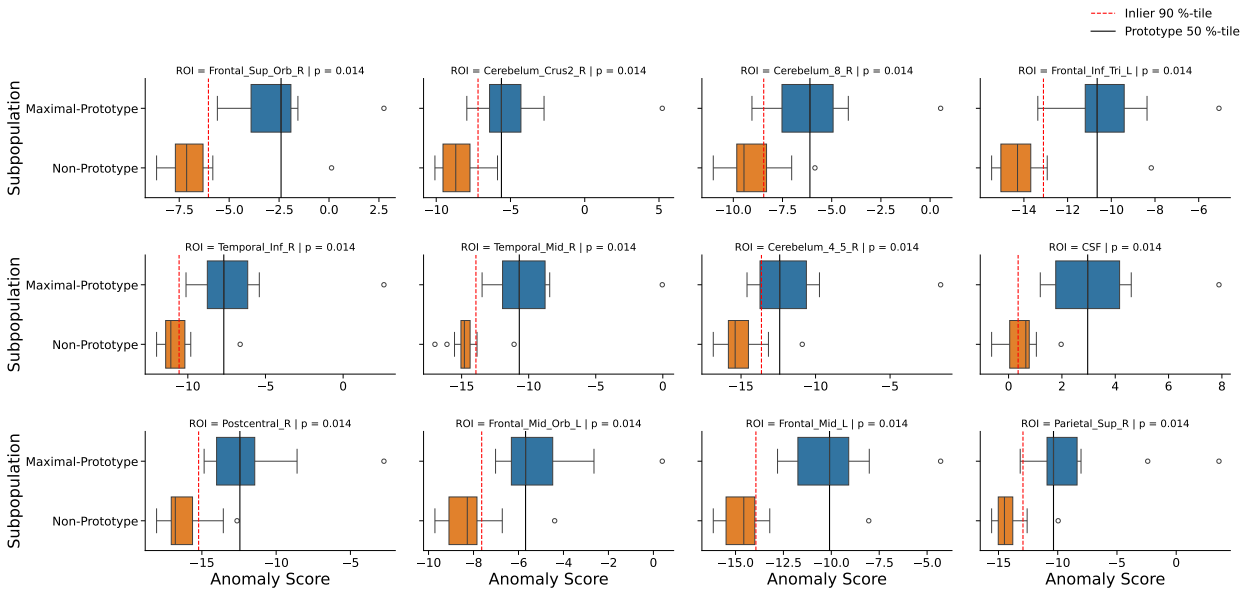


Figure 6.4: Box-and-Whisker plots of the most significant ROIs that significantly differ between the prototypical DS and non-prototypical DS population. The x-axis represents the anomaly score for each sample. *Higher* values are more anomalous. For reference, the 90-th percentile of the inlier anomaly scores for the given ROI is also plotted (red dashed line). All plotted ROI anomaly scores have statistically significant differences in their medians after Bonferroni correction.

groups have the same median, and thus belong to the same underlying distribution. The p-value from the Mann-Whitney U-test represents the probability of observing a test statistic as extreme (or more extreme) as the one calculated, given that the null hypothesis is true. As this analysis will be performing a statistical test for each ROI, it is important to control for the *multiple comparison problem*. One effective method is to correct for the False Discovery Rate (FDR). This correction will adjust the p-values belonging to a set of statistical tests such that a chosen significance level also corresponds to the expected proportion of false positives after thresholding. For instance, if we choose a p-value threshold of 0.01 after FDR correction, then we can expect 1% of the significant ROIs to be false positives. This analysis used the Benjamini-Hochberg procedure [Benjamini and Hochberg, 1995] for controlling FDR.

After the first stage filtering, 71 ROIs were identified. The medians of the filtered ROIs were compared across the prototypical and non-prototypical sub-populations. Figure 6.7 shows the 40 out of the 71 filtered ROIs that were significantly different between the prototypical samples and non-prototypical samples. The significance level was chosen to be a p-value of 0.05, *after* Bonferroni correction.

Figure 6.4 shows a subset of the most significant ROIs (top 12). For reference, the 90-percentile of the low-risk typical population is also plotted. Note how the median of the Maximal-Prototype are always higher

than the 90-th percentile, whereas the same does not hold for the non-prototypes. This tells us that the model finds this subpopulation particularly anomalous.

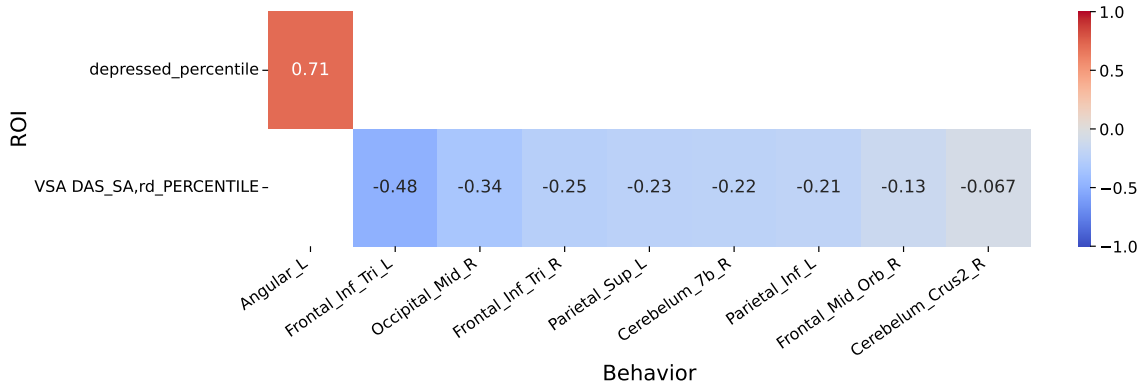


Figure 6.5: Heatmap of correlations between behavioral scores and ROI anomaly scores. The ROIs included in these experiments were those that were identified as significantly relevant for the prototypical class. All correlations shown are below a FDR-corrected p-value of 0.1. Note the high correlations between certain subscores and the ROI anomaly scores.

Do the regional anomaly scores correlate with behavioral assessments?

The previous section identified a set of anomalous brain regions that significantly deviated from the non-prototypical DS samples. This section will explore whether the anomaly scores of the identified ROIs have meaningful correlations with ground truth data. Individuals with neurodevelopmental disorders often exhibit atypical behaviors compared to a typically developing cohort. These behavioral atypicalities can be captured by behavioral assessments such as CBCL and DAS (Differential Ability Scales). We can test to see whether the anomaly scores produced by Spatial-MSMA correlate with scores of such assessments.

For each ROI, a Spearman-rank correlation was computed against a set of behavior scores. Specifically, the analysis covers CBCL, Vineland, and DAS scores. Both global, and subscores were included. A total of 35 behavior scores and 40 ROIs were tested, resulting in 1400 correlation experiments. p-values were computed for the pairs using a Permutation test with $10,000$ permutations. An FDR correction was applied to the resulting p-values using the Benjamini-Hochberg procedure. Figure 6.8 shows the correlation matrix for all ROI-Behavior pairs while Figure 6.5 shows correlations for only the significant pairs (corrected-$p < 0.1$) at an FDR of 10%. Specifically, the DAS subscore *Recall of design* has weak to moderate negative correlations with a large number of ROIs, while the CBCL subscore *depressed* has a strong correlation with the *Left Angular gyrus*.

Figure 6.6 shows the linear regression plot of the ROI anomaly scores and behavioral assessment subscores (percentiles). Note that the r values in the plots are the rank correlations. All pairs have *corrected* p-value below 0.1. Recall, that this implies an expected FDR of 10%. The plots show above moderate correlations across most ROI-Behavior pairs.

6.4 Conclusion

This chapter has demonstrated the potential for using the score-based representations learned by MSMA to gain valuable scientific insights, rather than just using them for anomaly detection. By leveraging the SOM algorithm to visualize the score-norm space, we were able to identify a distinct subpopulation within the Down Syndrome cohort that exhibited a characteristic neuroanatomical phenotype. The stability of this prototypical pattern across different SOM hyperparameter settings provided confidence that it represents a true underlying pattern in the data, rather than an artifact of the visualization technique.

Further analysis using Spatial-MSMA localized this prototypical pattern to specific brain regions that were significantly different from the non-prototypical Down Syndrome samples. Importantly, these identified brain regions were found to strongly correlate with behavioral measures like the CBCL, and Vineland assessments. This suggests that score-based representations are capturing meaningful neuroanatomic features that are linked to the functional and behavioral manifestations of Down Syndrome.

Overall, the case-study presented in this chapter showcases how the combination of MSMA, and Spatial-MSMA can be a powerful framework for not just detecting anomalies, but also generating substantive scientific hypotheses about the underlying drivers of atypical neurodevelopment. By connecting anomalous brain features to validated behavioral assessments, this methodology provides a data-driven approach to uncover the neuroanatomical basis of neurodevelopmental disorders. Future work can further expand on these insights to inform targeted neuroimaging biomarkers and intervention strategies.

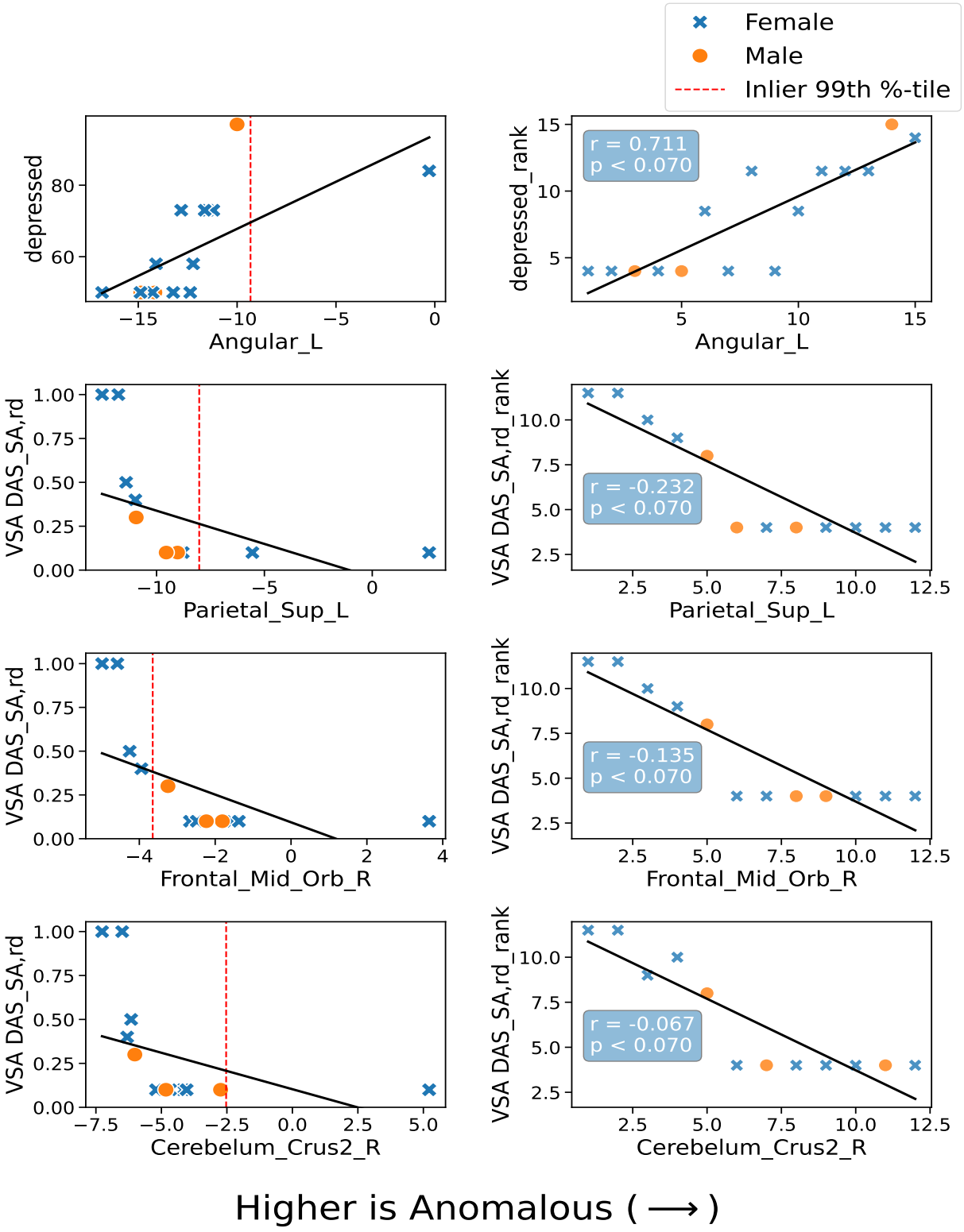


Figure 6.6: Spearman-rank correlations (corrected-$p < 0.1$) between ROIs belonging to prototypical Down Syndrome samples and behavioral scores. First column shows raw scores (percentiles for behavior scores and negative log-likelihoods for anomaly scores), while second column shows ranks. Note that all *percentiles* in the range 0-100. All p-values were corrected for FDR.

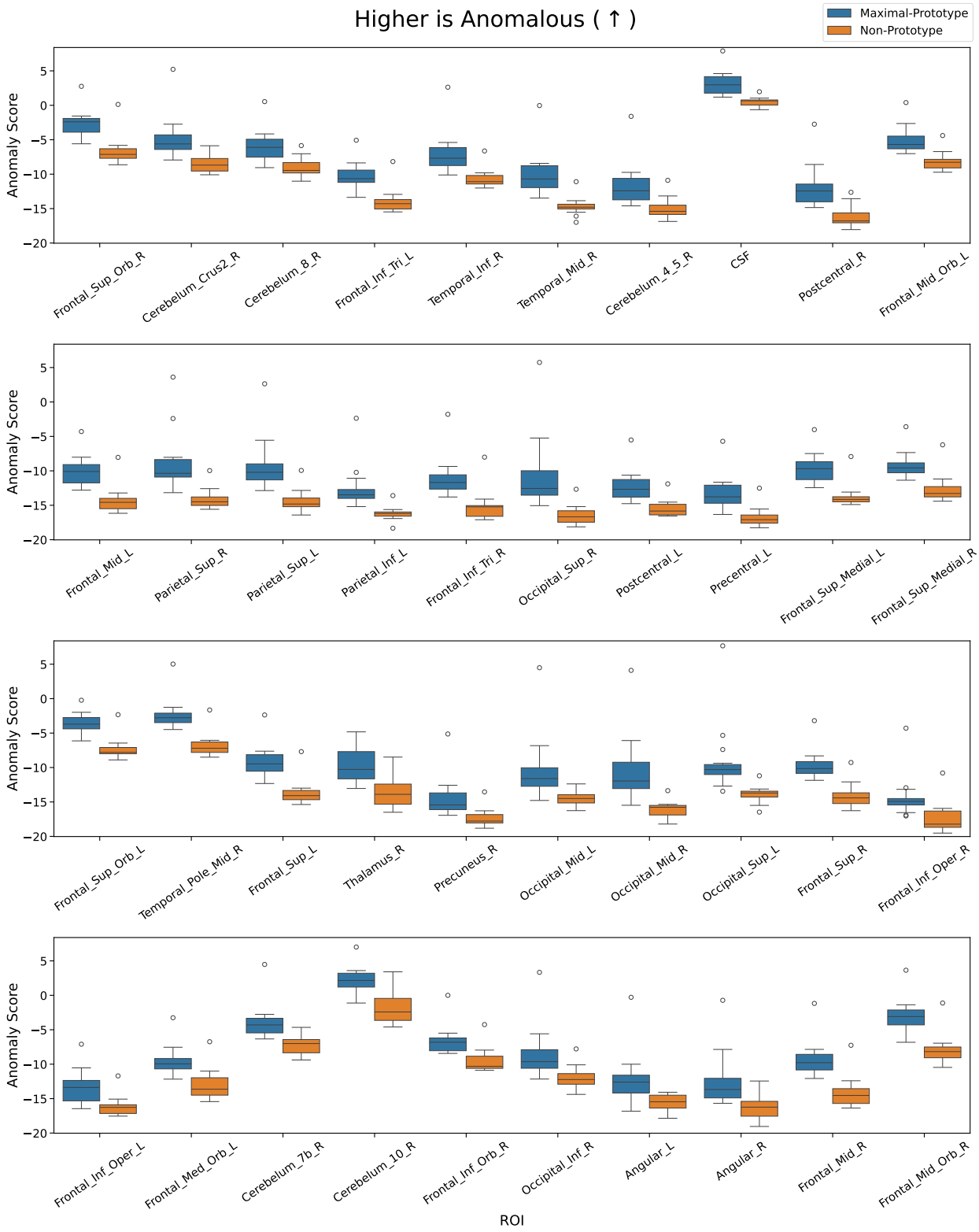


Figure 6.7: Box- and-Whisker plots of all ROIs that significantly differ between the prototypical DS and non-prototypical DS population. The ROIs are plotted in order of significance (from lower p-values to higher). They-axis represents the anomaly score per sample, higher values indicating the presence of higher anomalies. All plotted ROI anomaly scores have statistically significant differences in their medians *after* Bonferroni correction ($p < 0.05$).

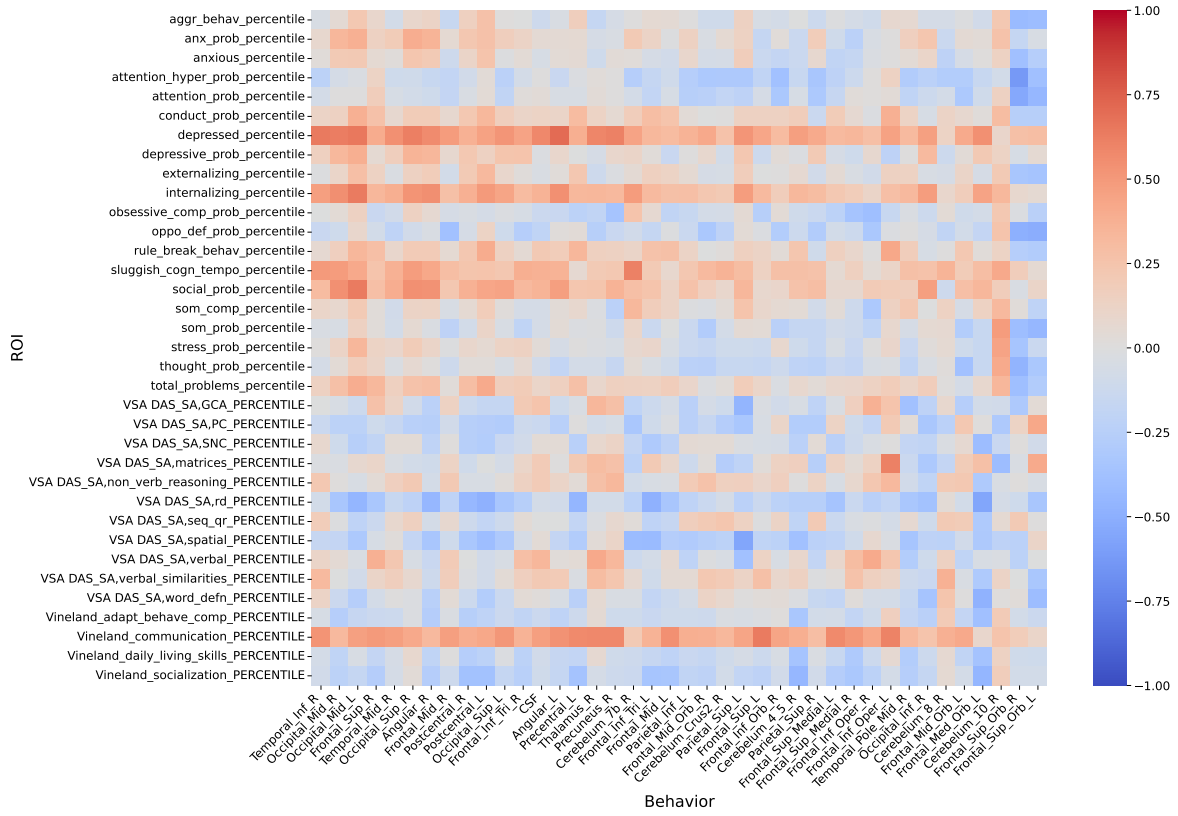


Figure 6.8: Heatmap of Spearman-rank correlations between behavioral scores and ROI anomaly scores. Most of these correlations are not significant after FDR correction due to the low sample size.

CHAPTER 7: CONCLUSION, LIMITATIONS, AND FUTURE WORK

7.1 Summary

This thesis introduces a novel anomaly detection methodology based on the analysis of gradients of the log-density (scores). The key contribution is the Multiscale Score Matching Analysis (MSMA), a method utilizing *multiple* scales to differentiate outliers from inliers. To incorporate categorical information, the Gumbel-Noise Score Matching (GNSM) objective was proposed, enabling the training of score matching networks on discrete data. This innovation not only extends MSMA to tabular data but also paves the way for future integration of demographic data into the models.

The methodology was further extended with Spatial-MSMA, which localizes anomalies to specific patches within an image. This technique was successfully applied to 3D MRI data, demonstrating significantly higher accuracy in detecting lesions compared to baseline methods. Additionally, a case study was presented to illustrate how score-norms can be employed for exploratory research, providing valuable scientific insights into the data.

7.2 Limitations

A fundamental limitation of MSMA, shared by many anomaly detection approaches, is its reliance on the training data to represent the typical population accurately. In other words, typicality is *solely* inferred from the training data. This places a significant burden on the user to ensure the training data is free from outliers and truly representative..

The computational requirements of MSMA present another challenge. The method requires deep learning score estimators with millions of parameters, necessitating multiple GPUs and extended training periods.[1] These resource demands may impede widespread adoption of the presented methods.

Score matching is also data-intensive, requiring thousands of samples for effective training. This can be particularly problematic in medical imaging, where data acquisition is costly and time-consuming. Limited

[1]For instance, the 3D score model used in chapters 5 and 6 was trained for 15 days, followed by ab additional two days for training Spatial-MSMA.

sample sizes may lead to overfitting, resulting in unreliable score estimates during testing. Moreover, when combining datasets from multiple sources, practitioners must address site-wide or scanner differences through preprocessing.

Lastly, while empirical evidence supports the anomaly detection capabilities of score-norms, the thesis lacks a theoretical analysis explaining why the score-norm space is expected to separate outliers effectively. Such an understanding could inform hyperparameter selection, optimization techniques, and data curation.

7.3 Future Work

There are several short-term directions for extending the work presented in this thesis. Firstly, the entire MSMA pipeline, including the score model, the likelihood model for MSMA, and the patch-based likelihood model for Spatial-MSMA, could be trained in an end-to-end fashion. This combined loss objective could focus the model's attention on input features that are best suited for anomaly detection.

Secondly, it may be worthwhile to automate the selection of noise scales used in MSMA, currently a user-defined hyperparameter. This could be achieved by employing a 'routing' network, similar to those used in mixture-of-experts models [Zoph et al., 2022]. Such a network would be responsible for selecting a subset of time points at inference time according to the input sample. Intuitively, the model would select only those time points that have the highest probability of detecting the input in the score-norm space.

Another avenue for future research would be to incorporate metadata paired with the images into the analysis. For natural images, one could include text captions or class labels, while for medical imaging, demographic data or diagnostic reports could be included alongside the brain MRIs. Furthermore, while the GNSM objective is well-suited for learning scores of categorical information, there are novel developments in categorical score matching, such as [Graves et al., 2024; Lou, Meng and Ermon, 2024]. It remains an open question whether these methods can be used in MSMA and how they compare to GNSM.

The aforementioned extensions are specific and can be considered 'low-hanging fruits' However, I am also interested in a broader, more long-term research endeavor for improving medical anomaly detectors. As mentioned in the previous section, the main limitation of anomaly detectors stems from defining typicality solely through the training data. If the training set of brain imaging data includes a significant number of lesioned brains, the model may assume them to be part of the typical set. However, any reasonably trained

medical practitioner would not make the same mistake. This is because medical practitioners accumulate a rich prior about healthy anatomy, including healthy brains, over the course of their studies.

The question arises: Can we build deep learning models that have also accumulated this prior? There is a concept of a ”world-model” in the field of robotics [Ha and Schmidhuber, 2018], which refers to the robot's understanding of physics and its best interpretation of the external world. Could we build deep learning models with a world-model of medicine and anatomy? Such a model would need to be multi-modal, taking both images and text as input. It could be trained on a combination of medical images (potentially paired with diagnostic reports), research papers, and medical textbooks. A model like this would have existing notions of pathologies, developmental disorders, and typical development, which it could use to inform its anomaly detection decisions.

Lastly, the utility of MSMA could be explored for the analysis of brain regions relevant to neurodevelopmental disorders other than Down Syndrome. Autism Spectrum Disorder (ASD) is an interesting candidate to investigate. The heterogeneity of ASD makes it difficult to isolate the underlying causes of the disorder, and one may hypothesize the presence of multiple underlying factors. If these factors manifest themselves in structural MRIs, it may be possible to observe subpopulations within an ASD cohort. An analysis similar to the one used in Chapter 6 could be applied to ASD, potentially discovering multiple prototypes. Such a finding would provide invaluable insight into the nature of this complex disorder.

APPENDIX A: EXPERIMENT DETAILS FOR CHAPTER 4

A.1 Hyperparameters

ECOD is hyperparamter free so no tuning was required. Early testing showed that Isolation Forests hyperparameters were stable. Note that I did not use labelled anomalies during hyperparamater tuning and the rest of the deep learning models were tuned on an inlier-only validation set.

GNSM Networks

Most of these details are easily identifiable in the open source code. However, I still provide basic information for posterity. I used the same ResNet-like architecture for all datasets:

$$\texttt{t} = \texttt{TimeEmbeddingLayer}(\lambda) \tag{A.1}$$

$$\texttt{Net(x, t)} = \texttt{Head(ResBlock(... ResBlock(x, t)))} \tag{A.2}$$

$$\texttt{ResBlock(x,t)} = \texttt{x + Linear(FiLM(x,t))} \tag{A.3}$$

$$\texttt{Head} = \texttt{Linear(LeakyReLU(LayerNorm(x))} \tag{A.4}$$

Note that the `FiLM` block is taken from [Perez et al., 2018] and the `TimeEmbeddingLayer` is the same as used in diffusion models [Song, Sohl-Dickstein, Kingma, Kumar, Ermon and Poole, 2020], using the `GaussianFourierProjection`. A simplified implementation of the ResBlock is shown below.

```
class TabResBlockpp(nn.Module):
    def __init__(self, d_in, d_out, time_emb_sz, act="gelu", dropout=0.0):

        self.norm = nn.LayerNorm(d_in)
        self.dense_1 = nn.Linear(d_in, d_out)
        self.act = get_act(act)
        self.film = FiLMBlock(time_emb_sz, d_out)
        self.dropout = nn.Dropout(dropout)
        self.dense_2 = nn.Linear(d_out, d_out)
```

```
def forward(self, x, t):

    h = self.act(self.norm(x))
    h = self.dense_1(h)
    h = self.film(h, t)
    h = self.dropout(h)
    h = self.dense_2(h)

    return x + h
```

For Bank I trained for 2MM iterations while for CMC and Solar, I trained for 600K iterations (as they were significantly smaller datasets). All the other models were trained for 1MM iters. I used the AdamW optimizer with default paramaters. The learning rate was set to $1e-3$ with a cosine decay to $1e-5$ spanning the number of iterations. I also use an Exponential Moving Average of the weights at a decay rate of 0.999.The base config is shown below.

```
def get_config():
    config = ml_collections.ConfigDict()
    # training
    config.training = training = ml_collections.ConfigDict()
    training.batch_size = 2048 # Except for CMC and Solar where it was 512
    training.n_steps = 1000001
    training.snapshot_freq = 10000 # Number of iterations for checkpointing

    # evaluation
    config.eval = evaluate = ml_collections.ConfigDict()
    evaluate.batch_size = 1024

    # data config holds information about the dataset such as number of categories
    config.data = data = ml_collections.ConfigDict()

    # default model parameters
    config.model = model = ml_collections.ConfigDict()
    model.name = "tab-resnet"
```

```
model.tau_min = 2.0
model.tau_max = 20
### Only relevant for Census
model.sigma_min = 1e-1
model.sigma_max = 1.0
#####
model.num_scales = 20
model.ndims = 1024
model.time_embedding_size = 128
model.layers = 20
model.dropout = 0.0
model.act = "gelu"
model.embedding_type = "fourier"
model.ema_rate = 0.999

# optimization
config.optim = optim = ml_collections.ConfigDict()
optim.weight_decay = 1e-4
optim.optimizer = "AdamW"
optim.lr = 1e-3
optim.beta1 = 0.9
optim.beta2 = 0.999
optim.grad_clip = 1.0
optim.scheduler = "cosine"
```

Lastly for MSMA, I trained a GMM on the combined train,val set. I ran a small grid search over number of components (3,5,7,9) and picked the one with the best likelihood.

DSVDD

For Deep SVDD I used the implementation available in the PyOD library [Zhao, Nasrullah and Li, 2019]. Initial testing showed that the autoencoder variant of this model usually performed better. This version adds a reconstruction loss to the one-class objective for increased regularization. The hidden neurons were set to

[1024, 512, 256], with the `swish` activation function. Training was done with the Adam optimizer at default hyperparamters, with learning rate set to 1e-3. I trained for 1000 epochs, with the batch size set to 512.

DAGMM

DAGMM proved to be very difficult to train as most implementations often unexpectedly result in NaNs. In fact the implementation used by [Han et al., 2022] never seemed to converge for any dataset. and the loss would not improve no matter how much I tweaked the hyperparameters. I believe the matrix inverse operation during the forward pass to be the culprit for this numerical instability.

We settled on modifiying a publicly available PyTorch implementation[1]. I added the following changes to improve stability and performance:

- Added Layer Normalization
- Added weight initialization
- Included checkpointing and early stopping using val set
- GMM parameters converted to double (float64)

Furthermore, I hand tuned hyperparameters for each dataset to find the most optimal (stable + performant) setting. Essentially, I tried to start from the same hyperparamters as DSVDD and tweaked until I got a stable model. I also early stopped on the checkpint that gave the best validation loss (tested every epoch). If a NaN was encountered before the first epoch was finished (i.e. before any checkpoint could be saved), I would restart training. The following hyperparameters were used for the final experiments:

```
hyp = {
    "input_dim": input_size,
    "hidden1_dim": 1024,
    "hidden2_dim": 512,
    "hidden3_dim": 256,
    "zc_dim": 2,
    "emb_dim": 128,
    "n_gmm": 2,
```

[1]https://github.com/lixiangwang/DAGMM-pytorch

```
    "dropout": 0.5,
    "lambda1": 0.1,
    "lambda2": 0.005,
    "lr": 1e-4,
    "batch_size": 256,
    "epochs": 1000,
    "patience_epochs": 10,
    "checkpoint": "best",
    "return_logits":False,
}

# Taken from KDDCUP-Rev config from original DAGMM paper
# Most other configs are unstable and frequently result in NaNs during training
if config.data.dataset in ["probe", "u2r"]:
    hyp["hidden1_dim"] = 120
    hyp["hidden2_dim"] = 60
    hyp["hidden3_dim"] = 30
    hyp["emb_dim"] = 10
    hyp["n_gmm"] = 4
    hyp["zc_dim"] = 1
    hyp["batch_size"] = 1024
    hyp["return_logits"] = True
    hyp["lr"] = 1e-5

if config.data.dataset == "bank":
    hyp["hidden1_dim"] = 64
    hyp["hidden2_dim"] = 32
    hyp["hidden3_dim"] = 16
    hyp["emb_dim"] = 10
    # hyp["zc_dim"] = 1
    hyp["batch_size"] = 4096
    hyp["lr"] = 1e-5
```

```
if config.data.dataset == "census":
    hyp["hidden1_dim"] = 256
    hyp["hidden2_dim"] = 128
    hyp["hidden3_dim"] = 64
    hyp["emb_dim"] = 10
    hyp["lr"] = 1e-5
```

APPENDIX B: ADDITIONAL FIGURES FOR CHAPTER 6

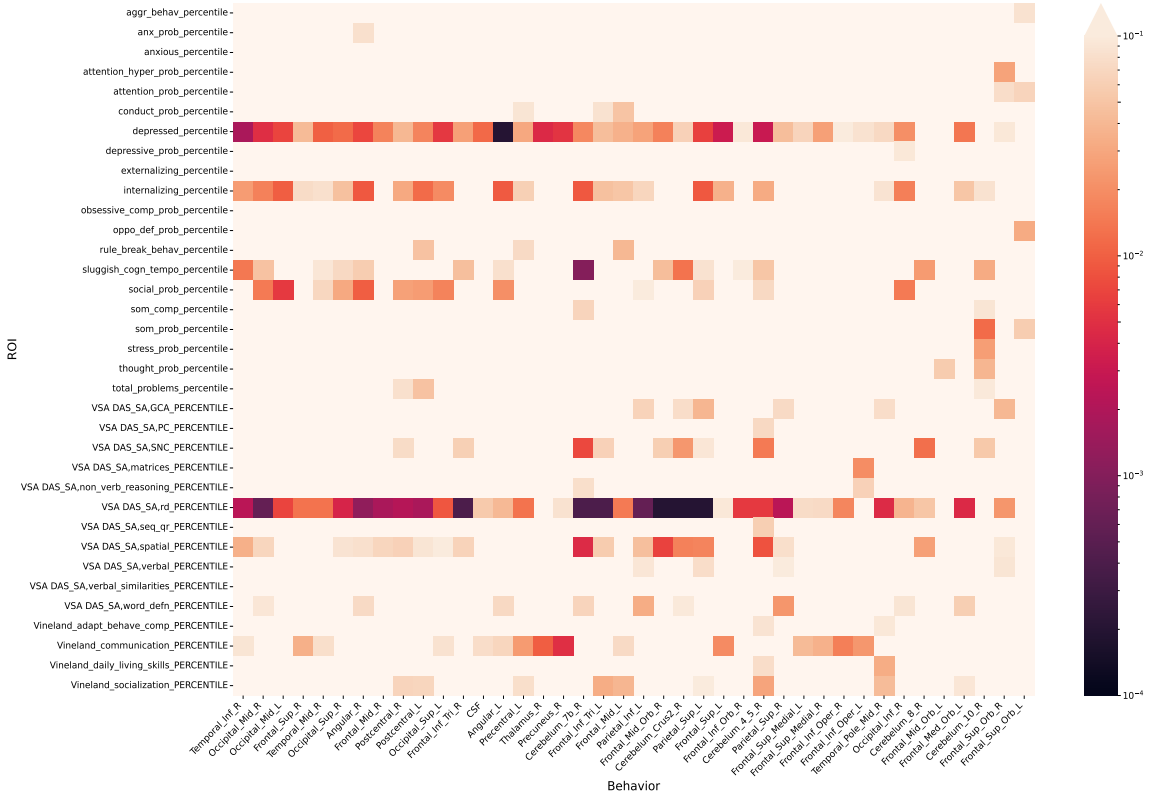


Figure B.1: Heatmap of raw p-values for Behavioral-ROI correlations. Only showing $p < 0.1$.

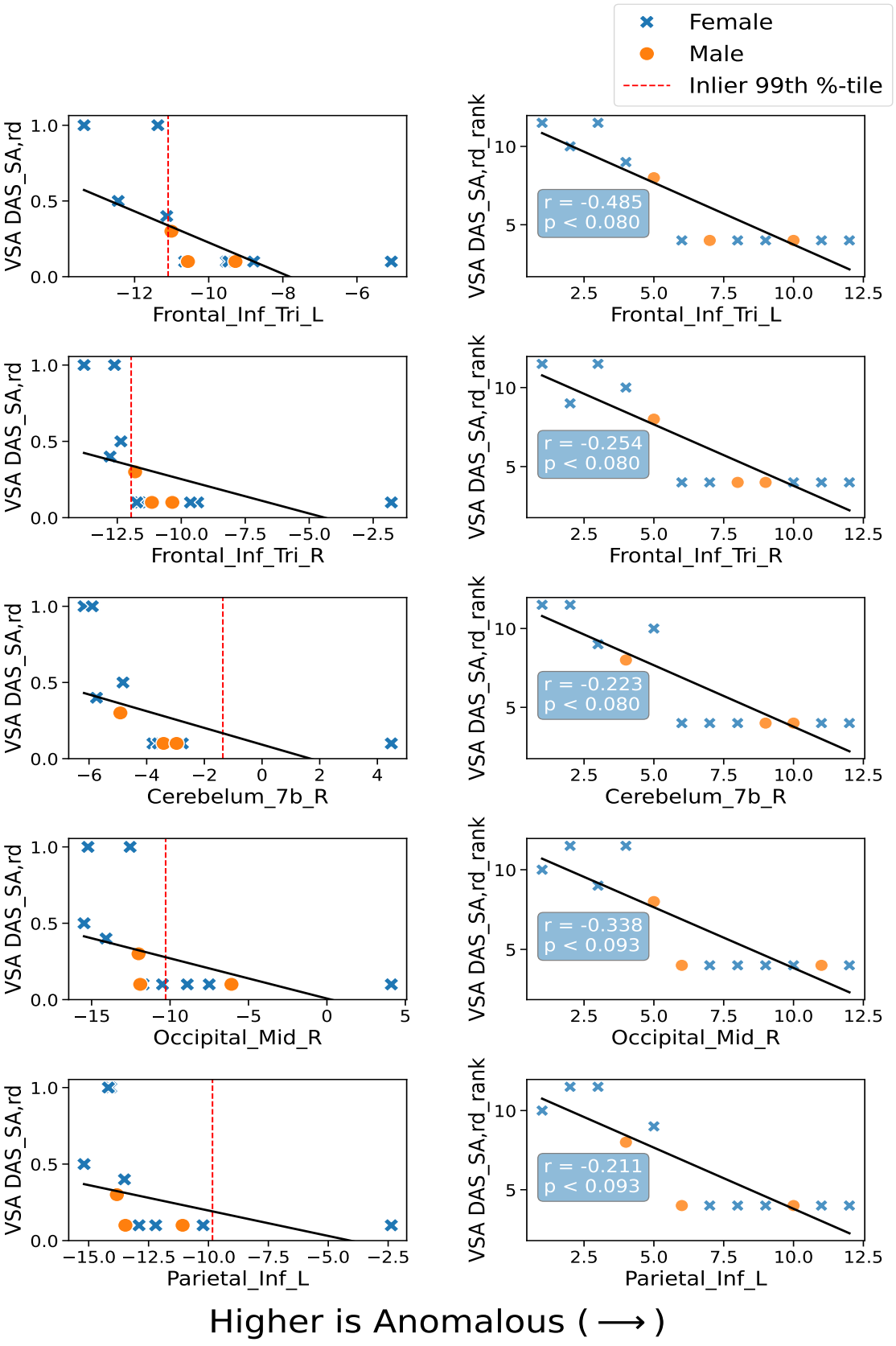


Figure B.2: Continuation of correlations from Figure 6.6

BIBLIOGRAPHY

Abdi, Hervé and Lynne J Williams. 2010. "Principal Component Analysis." *Wiley interdisciplinary reviews: computational statistics* 2(4):433–459.

Achenbach, Thomas M. N.d. The Child Behavior Checklist and Related Instruments. In *The Use of Psychological Testing for Treatment Planning and Outcomes Assessment, 2nd Ed.* Lawrence Erlbaum Associates Publishers pp. 429–466.

Aggarwal, Charu C. 2017. An Introduction to Outlier Analysis. In *Outlier Analysis*. Cham: Springer International Publishing pp. 1–34.

Anderson, Brian DO. 1982. "Reverse-time diffusion equation models." *Stochastic Processes and their Applications* 12(3):313–326.

Austin, Jacob, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow and Rianne van den Berg. 2021. Structured Denoising Diffusion Models in Discrete State-Spaces. In *Advances in Neural Information Processing Systems*, ed. M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang and J. Wortman Vaughan. Vol. 34 Curran Associates, Inc. pp. 17981–17993.

Baur, Christoph, Benedikt Wiestler, Shadi Albarqouni and Nassir Navab. 2019. Deep autoencoding models for unsupervised anomaly segmentation in brain MR images. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Revised Selected Papers, Part I 4*. Springer pp. 161–169.

Baur, Christoph, Stefan Denner, Benedikt Wiestler, Nassir Navab and Shadi Albarqouni. 2021. "Autoencoders for unsupervised anomaly segmentation in brain MR images: A comparative study." *Medical Image Analysis* 69:101952.

Behrendt, Finn, Debayan Bhattacharya, Julia Krüger, Roland Opfer and Alexander Schlaefer. 2023. Patched Diffusion Models for Unsupervised Anomaly Detection in Brain MRI. In *Medical Imaging with Deep Learning*.

Benjamini, Yoav and Yosef Hochberg. 1995. "Controlling the false discovery rate: a practical and powerful approach to multiple testing." *Journal of the Royal statistical society: series B (Methodological)* 57(1):289–300.

Bergmann, Paul, Michael Fauser, David Sattlegger and Carsten Steger. 2019. MVTec AD – A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Bergmann, Paul, Michael Fauser, David Sattlegger and Carsten Steger. 2020. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 4183–4192.

Bruno, Michael A., Eric A. Walker and Hani H. Abujudeh. 2015. "Understanding and Confronting Our Mistakes: The Epidemiology of Error in Radiology and Strategies for Error Reduction." *RadioGraphics* 35(6):1668–1676.

Casey, BJ, Tariq Cannonier, May I Conley, Alexandra O Cohen, Deanna M Barch, Mary M Heitzeg, Mary E Soules, Theresa Teslovich, Danielle V Dellarco, Hugh Garavan et al. 2018. "The adolescent brain cognitive development (ABCD) study: imaging acquisition across 21 sites." *Developmental cognitive neuroscience* 32:43–54.

Chamberlin, J.H., C. Smith, U.J. Schoepf, S. Nance, S. Elojeimy, J. O'Doherty, D. Baruah, J.R. Burt, A. Varga-Szemes and I.M. Kabakus. 2023. "A Deep Convolutional Neural Network Ensemble for Composite Identification of Pulmonary Nodules and Incidental Findings on Routine PET/CT." 78(5):e368–e376.
URL: *https://www.sciencedirect.com/science/article/pii/S000992602300051X*

Chen, Jinghui, Saket Sathe, Charu Aggarwal and Deepak Turaga. 2017. Outlier detection with autoencoder ensembles. In *Proceedings of the 2017 SIAM international conference on data mining*. SIAM pp. 90–98.

Chen, Liang-Chieh, George Papandreou, Florian Schroff and Hartwig Adam. 2017. "Rethinking Atrous Convolution for Semantic Image Segmentation." *ArXiv* abs/1706.05587.

Chen, Yunqiang, Xiang Sean Zhou and T.S. Huang. 2001. One-class SVM for learning in image retrieval. In *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*. Vol. 1 pp. 34–37 vol.1.

da S Senra Filho, Antonio Carlos, Fabrício Henrique Simozo, Antonio Carlos dos Santos and Luiz Otavio Murta Junior. 2019. "Multiple Sclerosis multimodal lesion simulation tool (MS-MIST)." *Biomedical Physics and Engineering Express* 5(3):035003.
URL: *https://dx.doi.org/10.1088/2057-1976/ab08fc*

DeVries, Terrance and Graham W Taylor. 2018*a*. "Learning Confidence for Out-of-Distribution Detection in Neural Networks." *stat* 1050:13.

DeVries, Terrance and Graham W. Taylor. 2018*b*. "Learning Confidence for Out-of-Distribution Detection in Neural Networks.".

Dinh, Laurent, Jascha Sohl-Dickstein and Samy Bengio. 2017. Density estimation using Real NVP. In *International Conference on Learning Representations*.
URL: *https://openreview.net/forum?id=HkpbnH9lx*

Durkan, Conor, Artur Bekasov, Iain Murray and George Papamakarios. 2019. Neural Spline Flows. In *Advances in Neural Information Processing Systems*, ed. H. Wallach, H. Larochelle, A. Beygelzimer, E. Fox and R. Garnett. Vol. 32 Curran Associates, Inc.
URL: *https://proceedings.neurips.cc/paper*$_f$$iles/paper/2019/file/7ac71d433f282034e088473244df8c02−Paper.pdf*

Everingham, M., L. Van Gool, C. K. I. Williams, J. Winn and A. Zisserman. 2010. "The Pascal Visual Object Classes (VOC) Challenge." *International Journal of Computer Vision* 88(2):303–338.

Fedorov, Andriy, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, John Buatti, Stephen Aylward, James V. Miller, Steve Pieper and Ron Kikinis. 2012. "3D Slicer as an Image Computing Platform for the Quantitative Imaging Network." 30(9):1323–1341.

Gilmore, John H, Benjamin Langworthy, Jessica B Girault, Jason Fine, Shaili C Jha, Sun Hyung Kim, Emil Cornea and Martin Styner. 2020. "Individual Variation of Human Cortical Structure Is Established in the First Year of Life." *Biological psychiatry. Cognitive neuroscience and neuroimaging* .

Girault, Jessica B. and Joseph Piven. 2020. "The Neurodevelopment of Autism from Infancy Through Toddlerhood." 30(1):97–114.

Gorishniy, Yury, Ivan Rubachev, Valentin Khrulkov and Artem Babenko. 2021. "Revisiting deep learning models for tabular data." *Advances in Neural Information Processing Systems* 34:18932–18943.

Graham, Mark S, Walter HL Pinaya, Petru-Daniel Tudosiu, Parashkev Nachev, Sebastien Ourselin and Jorge Cardoso. 2023. Denoising Diffusion Models for Out-of-Distribution Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 2947–2956.

Grathwohl, Will, Kuan-Chieh Wang, Joern-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi and Kevin Swersky. 2020. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*.

Graves, Alex, Rupesh Kumar Srivastava, Timothy Atkinson and Faustino Gomez. 2024. "Bayesian Flow Networks.".

Gumbel, Emil Julius. 1954. *Statistical theory of extreme values and some practical applications: a series of lectures*. Vol. 33 US Government Printing Office.

Ha, David and Jürgen Schmidhuber. 2018. "World models." *arXiv preprint arXiv:1803.10122* .

Hamner, Taralee, Manisha D. Udhnani, Karol Z. Osipowicz and Nancy Raitano Lee. 2018. "Pediatric Brain Development in Down Syndrome: A Field in Its Infancy." 24(9):966–976.
URL: *https://www.cambridge.org/core/product/identifier/S1355617718000206/type/journal*$_{a}rticle$

Han, Songqiao, Xiyang Hu, Hailiang Huang, Minqi Jiang and Yue Zhao. 2022. ADBench: Anomaly Detection Benchmark. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Hawkins, Simon, Hongxing He, Graham Williams and Rohan Baxter. 2002. Outlier detection using replicator neural networks. In *Data Warehousing and Knowledge Discovery: 4th International Conference, DaWaK 2002 Aix-en-Provence, France, September 4–6, 2002 Proceedings 4*. Springer pp. 170–180.

Hendrycks, Dan and Kevin Gimpel. 2016. "Gaussian Error Linear Units (GELUs).".

Hendrycks, Dan and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Ho, Jonathan, Ajay Jain and Pieter Abbeel. 2020. "Denoising diffusion probabilistic models." *Advances in Neural Information Processing Systems* 33:6840–6851.

Hoogeboom, Emiel, Didrik Nielsen, Priyank Jaini, Patrick Forré and Max Welling. 2021. Argmax Flows and Multinomial Diffusion: Learning Categorical Distributions. In *Advances in Neural Information Processing Systems*, ed. M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang and J. Wortman Vaughan. Vol. 34 Curran Associates, Inc. pp. 12454–12465.

Hyvärinen, Aapo. 2005. Estimation of Non-Normalized Statistical Models by Score Matching. Technical report.

Jang, Eric, Shixiang Gu and Ben Poole. 2017. Categorical Reparametrization with Gumble-Softmax. In *International Conference on Learning Representations (ICLR 2017)*. OpenReview. net.

Kascenas, Antanas, Nicolas Pugeault and Alison Q. O'Neil. 2022. Denoising Autoencoders for Unsupervised Anomaly Detection in Brain MRI. In *Proceedings of the 5th International Conference on Medical Imaging with Deep Learning*, ed. Ender Konukoglu, Bjoern Menze, Archana Venkataraman, Christian Baumgartner, Qi Dou and Shadi Albarqouni. Vol. 172 of *Proceedings of Machine Learning Research* PMLR pp. 653–664.
URL: *https://proceedings.mlr.press/v172/kascenas22a.html*

Kascenas, Antanas, Pedro Sanchez, Patrick Schrempf, Chaoyang Wang, William Clackett, Shadia S. Mikhael, Jeremy P. Voisey, Keith Goatman, Alexander Weir, Nicolas Pugeault, Sotirios A. Tsaftaris and Alison Q. O'Neil. 2023. "The Role of Noise in Denoising Models for Anomaly Detection in Medical Images." *Medical Image Analysis* p. 102963.

Kascenas, Antanas et al. 2023. "The Role of Noise in Denoising Models for Anomaly Detection in Medical Images." 90:102963.
URL: *https://www.sciencedirect.com/science/article/pii/S1361841523002232*

Kim, Mingyu, Jihye Yun, Yongwon Cho, Keewon Shin, Ryoungwoo Jang, Hyun-jin Bae and Namkug Kim. 2019. "Deep learning in medical imaging." *Neurospine* 16(4):657. Publisher: Korean Spinal Neurosurgery Society.

Kingma, Diederik P and Max Welling. 2013. "Auto-encoding variational bayes." *arXiv preprint arXiv:1312.6114* .

Kinouchi, Makoto, Naoshi Takada, Yoshihiro Kudo and Toshimichi Ikemura. 2002. "Quick learning for batch-learning self-organizing map." *Genome Informatics* 13:266–267.

Kirichenko, Polina, Pavel Izmailov and Andrew G Wilson. 2020*a*. Why Normalizing Flows Fail to Detect Out-of-Distribution Data. In *Advances in Neural Information Processing Systems*, ed. H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan and H. Lin. Vol. 33 Curran Associates, Inc. pp. 20578–20589.

Kirichenko, Polina, Pavel Izmailov and Andrew G Wilson. 2020*b*. Why Normalizing Flows Fail to Detect Out-of-Distribution Data. In *Advances in Neural Information Processing Systems*, ed. H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan and H. Lin. Vol. 33 Curran Associates, Inc. pp. 20578–20589.
URL: *https://proceedings.neurips.cc/paper*$_f iles/paper/2020/file/ecb9fe2fbb99c31f567e9823e884dbec-Paper.pdf$

Kohonen, Teuvo. 1990. "The self-organizing map." *Proceedings of the IEEE* 78(9):1464–1480.

Krizhevsky, Alex, Geoffrey Hinton et al. 2009. "Learning multiple layers of features from tiny images.".

Krizhevsky, Alex, Ilya Sutskever and Geoffrey E Hinton. 2017. "ImageNet classification with deep convolutional neural networks." *Communications of the ACM* 60(6):84–90.

Lee, JG, S Jun, YW Cho, H Lee, GB Kim, JB Seo and N Kim. 2017. "Deep Learning in Medical Imaging: General Overview." *Korean Journal of Radiology* 18(4):570–584.

Lee, Kimin, Honglak Lee, Kibok Lee and Jinwoo Shin. 2018. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Lei, Weihua, Cleber Zanchettin, Zoey E. Ho and Luís A. Nunes Amaral. 2023. "Quantifying the Impact of Uninformative Features on the Performance of Supervised Classification and Dimensionality Reduction Algorithms." 1(4):046118.

URL: *https://pubs.aip.org/aml/article/1/4/046118/2928862/Quantifying-the-impact-of-uninformative-features*

Lenroot, Rhoshel and Pui Ka Yeung. 2013. "Heterogeneity within Autism Spectrum Disorders: What have We Learned from Neuroimaging Studies?" *Frontiers in Human Neuroscience* 7.
URL: *https://www.frontiersin.org/articles/10.3389/fnhum.2013.00733*

Li, Zheng, Yue Zhao, Nicola Botta, Cezar Ionescu and Xiyang Hu. 2020. COPOD: Copula-Based Outlier Detection. In *2020 IEEE International Conference on Data Mining (ICDM)*. pp. 1118–1123.

Li, Zheng, Yue Zhao, Xiyang Hu, Nicola Botta, Cezar Ionescu and George Chen. 2022. "Ecod: Unsupervised outlier detection using empirical cumulative distribution functions." *IEEE Transactions on Knowledge and Data Engineering* . Publisher: IEEE.

Liang, Shiyu, Yixuan Li and R. Srikant. 2017. "Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks." *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings* .

Lin, Tsung-Yi, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Doll'a r and C. Lawrence Zitnick. 2014. "Microsoft COCO: Common Objects in Context." *CoRR* abs/1405.0312.
URL: *http://arxiv.org/abs/1405.0312*

Liu, Jiaqi, Guoyang Xie, Jinbao Wang, Shangnian Li, Chengjie Wang, Feng Zheng and Yaochu Jin. 2024. "Deep industrial image anomaly detection: A survey." *Machine Intelligence Research* 21(1):104–135.

Liu, Zhenzhen, Jin Peng Zhou, Yufan Wang and Kilian Q Weinberger. 2023. "Unsupervised Out-of-Distribution Detection with Diffusion Inpainting.".

Lord, Catherine, Michael Rutter, Susan Goode, Jacquelyn Heemsbergen, Heather Jordan, Lynn Mawhood and Eric Schopler. 2012. "Autism diagnostic observation schedule." *Journal of Autism and Developmental Disorders* .

Lou, Aaron, Chenlin Meng and Stefano Ermon. 2024. "Discrete Diffusion Language Modeling by Estimating the Ratios of the Data Distribution.".
URL: *https://openreview.net/forum?id=71mqtQdKB9*

Lundberg, Scott M and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30*, ed. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett. Curran Associates, Inc. pp. 4765–4774.
URL: *http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf*

Luo, Guoting, Wei Xie, Ronghui Gao, Tao Zheng, Lei Chen and Huaiqiang Sun. 2023. "Unsupervised anomaly detection in brain MRI: Learning abstract distribution from massive healthy brains." *Computers in Biology and Medicine* 154:106610.
URL: *https://www.sciencedirect.com/science/article/pii/S0010482523000756*

Maddison, C, A Mnih and Y Teh. 2017. The concrete distribution: A continuous relaxation of discrete random variables. In *Proceedings of the international conference on learning Representations*. International Conference on Learning Representations.

Maddison, Chris J, Daniel Tarlow and Tom Minka. 2014. "A* sampling." *Advances in neural information processing systems* 27.

Mahmood, Ahsan, Junier Oliva and Martin Andreas Styner. 2021. Multiscale Score Matching for Out-of-Distribution Detection. In *International Conference on Learning Representations*.

McInnes, Leland, John Healy and James Melville. 2020. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.".

Morningstar, Warren, Cusuh Ham, Andrew Gallagher, Balaji Lakshminarayanan, Alex Alemi and Joshua Dillon. 2021. Density of States Estimation for Out of Distribution Detection. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, ed. Arindam Banerjee and Kenji Fukumizu. Vol. 130 of *Proceedings of Machine Learning Research* PMLR pp. 3232–3240.
URL: *https://proceedings.mlr.press/v130/morningstar21a.html*

Nalisnick, Eric, Akihiro Matsukawa, Yee Whye Teh and Balaji Lakshminarayanan. 2020. "Detecting Out-of-Distribution Inputs to Deep Generative Models Using Typicality.".

Nalisnick, Eric, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur and Balaji Lakshminarayanan. 2019. Do Deep Generative Models Know What They Don't Know? In *International Conference on Learning Representations*.

Netzer, Yuval, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu and Andrew Y Ng. 2011. "Reading digits in natural images with unsupervised feature learning.".

Niehoff, Julius Henning, Jana Kalaitzidis, Jan Robert Kroeger, Denise Schoenbeck, Jan Borggrefe and Arwed Elias Michael. 2023. "Evaluation of the Clinical Performance of an AI-based Application for the Automated Analysis of Chest X-rays." 13(1):3680.
URL: *https://doi.org/10.1038/s41598-023-30521-2*

Pang, Guansong, Chunhua Shen, Longbing Cao and Anton Van Den Hengel. 2021. "Deep Learning for Anomaly Detection: A Review." *ACM Comput. Surv.* 54(2). Place: New York, NY, USA Publisher: Association for Computing Machinery.

Pang, Guansong, Longbing Cao and Ling Chen. 2021. "Homophily outlier detection in non-IID categorical data." *Data Mining and Knowledge Discovery* 35:1163–1224.

Papamakarios, George, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed and Balaji Lakshminarayanan. 2021. "Normalizing Flows for Probabilistic Modeling and Inference." *J. Mach. Learn. Res.* 22(1).

Papamakarios, George, Theo Pavlakou and Iain Murray. 2017. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*. pp. 2338–2347.

Parker, Samantha E, Cara T Mai, Mark A Canfield, Russel Rickard, Ying Wang, Robert E Meyer, Patrick Anderson, Craig A Mason, Julianne S Collins, Russell S Kirby et al. 2010. "Updated national birth prevalence estimates for selected birth defects in the United States, 2004–2006." *Birth Defects Research Part A: Clinical and Molecular Teratology* 88(12):1008–1016.

Perez, Ethan, Florian Strub, Harm De Vries, Vincent Dumoulin and Aaron Courville. 2018. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 32.

Pinaya, Walter HL, Mark S Graham, Robert Gray, Pedro F Da Costa, Petru-Daniel Tudosiu, Paul Wright, Yee H Mah, Andrew D MacKinnon, James T Teo, Rolf Jager et al. 2022. Fast unsupervised brain

anomaly detection and segmentation with diffusion models. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer pp. 705–714.

Ren, J., Peter J. Liu, E. Fertig, Jasper Snoek, Ryan Poplin, Mark A. DePristo, Joshua V. Dillon and Balaji Lakshminarayanan. 2019. Likelihood Ratios for Out-of-Distribution Detection. In *NeurIPS*.

Reynolds, Douglas A et al. 2009. "Gaussian mixture models." *Encyclopedia of biometrics* 741(659-663).

Rolls, Edmund T., Chu-Chung Huang, Ching-Po Lin, Jianfeng Feng and Marc Joliot. 2020. "Automated anatomical labelling atlas 3." *NeuroImage* 206:116189.
URL: *https://www.sciencedirect.com/science/article/pii/S1053811919307803*

Ruff, Lukas, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G. Dietterich and Klaus-Robert Müller. 2021. "A Unifying Review of Deep and Shallow Anomaly Detection." *Proceedings of the IEEE* 109(5):756–795.

Ruff, Lukas, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller and Marius Kloft. 2018. Deep One-Class Classification. In *Proceedings of the 35th International Conference on Machine Learning*, ed. Jennifer Dy and Andreas Krause. Vol. 80 of *Proceedings of Machine Learning Research* PMLR pp. 4393–4402.

Schlegl, Thomas, Philipp Seeböck, Sebastian M Waldstein, Georg Langs and Ursula Schmidt-Erfurth. 2019. "f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks." *Medical image analysis* 54:30–44.

Selvaraju, Ramprasaath R., Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh and Dhruv Batra. 2016. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization." *International Journal of Computer Vision* 128:336 – 359.
URL: *https://api.semanticscholar.org/CorpusID:15019293*

Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh and Dhruv Batra. 2017. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*. pp. 618–626.

Shannon, C. E. 1948. "A mathematical theory of communication." *The Bell System Technical Journal* 27(3):379–423.

Shen, Mark D., Meghan R. Swanson, Jason J. Wolff, Jed T. Elison, Jessica B. Girault, Sun Hyung Kim, Rachel G. Smith, Michael M. Graves, Leigh Anne H. Weisenfeld, Lisa Flake, Leigh MacIntyre, Julia L. Gross, Catherine A. Burrows, Vladimir S. Fonov, D. Louis Collins, Alan C. Evans, Guido Gerig, Robert C. McKinstry, Juhi Pandey, Tanya St John, Lonnie Zwaigenbaum, Annette M. Estes, Stephen R. Dager, Robert T. Schultz, Martin A. Styner, Kelly N. Botteron, Heather C. Hazlett, Joseph Piven and IBIS Network. 2022. "Subcortical Brain Development in Autism and Fragile X Syndrome: Evidence for Dynamic, Age- and Disorder-Specific Trajectories in Infancy." 179(8):562–572.

Simonyan, Karen, Andrea Vedaldi and Andrew Zisserman. 2013. "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps." *CoRR* abs/1312.6034.
URL: *https://api.semanticscholar.org/CorpusID:1450294*

Simonyan, Karen and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, ed. Yoshua Bengio and Yann LeCun.

Smith, Samuel L., Pieter-Jan Kindermans and Quoc V. Le. 2018. Don't Decay the Learning Rate, Increase the Batch Size. In *International Conference on Learning Representations*.
URL: *https://openreview.net/forum?id=B1Yy1BxCZ*

Sohl-Dickstein, Jascha, Eric Weiss, Niru Maheswaranathan and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR pp. 2256–2265.

Song, Yang, Jascha Sohl-Dickstein, Diederik Kingma, Abhishek Kumar, Stefano Ermon and Ben Poole. 2020. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*.

Song, Yang, Sahaj Garg, Jiaxin Shi and Stefano Ermon. 2020. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*. PMLR pp. 574–584.

Song, Yang and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*. pp. 11918–11930.

Song, Yang and Stefano Ermon. 2020. "Improved techniques for training score-based generative models." *Advances in neural information processing systems* 33:12438–12448.

Stephens, Rebecca L, Benjamin W Langworthy, Sarah J Short, Jessica B Girault, Martin Styner and John H Gilmore. 2020. "White Matter Development from Birth to 6 Years of Age: A Longitudinal Study." *Cerebral cortex (New York, N.Y. : 1991)* 7:7456.

Sun, Haoran, Lijun Yu, Bo Dai, Dale Schuurmans and Hanjun Dai. 2023. Score-based Continuous-time Discrete Diffusion Models. In *The Eleventh International Conference on Learning Representations*.

Taha, Abdel Aziz and Allan Hanbury. 2015. "Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool." *BMC medical imaging* 15(1):1–28.

Tan, Mingxing and Quoc Le. 2021. Efficientnetv2: Smaller Models and Faster Training. In *International Conference on Machine Learning*. PMLR pp. 10096–10106.

Tschuchnig, Maximilian E and Michael Gadermayr. 2022. Anomaly detection in medical imaging-a mini review. In *Data Science–Analytics and Applications: Proceedings of the 4th International Data Science Conference–iDSC2021*. Springer pp. 33–38.

Valentini, Vincenzo, Luca Boldrini, Andrea Damiani and Ludvig P Muren. 2014. "Recommendations on how to establish evidence from auto-segmentation software in radiotherapy." *Radiotherapy and Oncology* 112(3):317–320.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, ed. I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett. Vol. 30 Curran Associates, Inc.
URL: *https://proceedings.neurips.cc/paper*$_f$$iles/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf*

Vershynin, Roman. 2018. *High-dimensional probability: An introduction with applications in data science*. Vol. 47 Cambridge University Press.

Vincent, Pascal. 2011. "A connection between score matching and denoising autoencoders." *Neural computation* 23(7):1661–1674.

Wang, Yingfan, Haiyang Huang, Cynthia Rudin and Yaron Shaposhnik. 2021. "Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMap, and PaCMAP for Data Visualization." *Journal of Machine Learning Research* 22(201):1–73.
URL: *http://jmlr.org/papers/v22/20-1061.html*

Weerasekera, Akila, Adrian Ion-Mărgineanu, Garry Nolan and Maria Mody. 2022. "Subcortical Brain Morphometry Differences between Adults with Autism Spectrum Disorder and Schizophrenia." *Brain Sciences* 12(4).
URL: *https://www.mdpi.com/2076-3425/12/4/439*

Wu, Suya, Enmao Diao, Khalil Elkhalil, Jie Ding and Vahid Tarokh. 2022. "Score-Based Hypothesis Testing for Unnormalized Models." *IEEE Access* 10:71936–71950.

Wyatt, Julian, Adam Leach, Sebastian M Schmon and Chris G Willcocks. 2022. Anoddpm: Anomaly Detection with Denoising Diffusion Probabilistic Models Using Simplex Noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 650–656.

Xu, Pingmei, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R. Kulkarni and Jianxiong Xiao. 2015. "TurkerGaze: Crowdsourcing Saliency with Webcam based Eye Tracking.".

You, Suhang, Kerem C. Tezcan, Xiaoran Chen and Ender Konukoglu. 2019. Unsupervised Lesion Detection via Image Restoration with a Normative Prior. In *Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning*, ed. M. Jorge Cardoso, Aasa Feragen, Ben Glocker, Ender Konukoglu, Ipek Oguz, Gozde Unal and Tom Vercauteren. Vol. 102 of *Proceedings of Machine Learning Research* PMLR pp. 540–556.
URL: *https://proceedings.mlr.press/v102/you19a.html*

Yu, Fisher, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser and Jianxiong Xiao. 2015. "Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop." *arXiv preprint arXiv:1506.03365* .

Zhai, Shuangfei, Yu Cheng, Weining Lu and Zhongfei Zhang. 2016. Deep Structured Energy Based Models for Anomaly Detection. In *Proceedings of The 33rd International Conference on Machine Learning*, ed. Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48 of *Proceedings of Machine Learning Research* New York, New York, USA: PMLR pp. 1100–1109.

Zhao, Yue, Zain Nasrullah and Zheng Li. 2019. "PyOD: A Python Toolbox for Scalable Outlier Detection." *Journal of Machine Learning Research* 20(96):1–7.

Zong, Bo, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho and Haifeng Chen. 2018. Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection. In *International Conference on Learning Representations*.

Zoph, Barret, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer and William Fedus. 2022. "ST-MoE: Designing Stable and Transferable Sparse Expert Models.".

Zwillinger, Daniel and Stephen Kokoska. 1999. *CRC standard probability and statistics tables and formulae*. Crc Press.