This thesis revisits the fundamental components of deep learning and evaluates their application in the context of medical image analysis. It identifies three main challenges where deep learning falls short in this domain: the integration of expert knowledge, the leveraging of unlabeled data, and the estimation of predictive uncertainty. It demonstrates that expert knowledge can be effectively integrated into deep learning models, that leveraging unlabeled data through self-supervised learning can enhance model performance, and that predictive uncertainty can be improved with more flexible variational inference methods.
Deep Learning for Medical Data

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. P.P.C.C. Verbeek
ten overstaan van een door het College voor Promoties
ingeselde commissie,
in het openbaar te verdedigen in de Agnietenkapel
op maandag 1 juli 2024, te 10:00 uur

door

Bastiaan Sjouke Veeling

geboren te Zeist
PROMOTIECOMMISSIE

Promotor:
prof. dr. M. Welling
Universiteit van Amsterdam

Copromotor:
prof. dr. M. Petković
Technische Universiteit Eindhoven

Overige leden:
prof. dr Z. Akata
University of Tübingen
prof. dr. P.T. Groth
Universiteit van Amsterdam
prof. dr. E. Kanoulas
Universiteit van Amsterdam
dr. S. Mandt
University of California, Irvine
dr. ir. E.J. Bekkers
Universiteit van Amsterdam

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

The work described in this thesis has been primarily carried out at the Amsterdam Machine Learning Lab (AMLab) of the University of Amsterdam and at Philips Research. Funding for the position was graciously provided by Philips Research.

Copyright © 2024 by B. S. Veeling, Amsterdam, The Netherlands.
SUMMARY

This thesis revisits the fundamental components of deep learning and evaluates their application in the context of medical image analysis. It identifies three main challenges where deep learning falls short in this domain: the integration of expert knowledge, the leveraging of unlabeled data, and the estimation of predictive uncertainty. The thesis is structured into parts that address these challenges respectively.

In Part I, the thesis introduces a novel deep learning model that incorporates expert knowledge through roto-reflective equivariance to improve the accuracy and robustness of medical imaging tasks, specifically in the detection of metastatic tissue in histopathology slides. The proposed model outperforms traditional CNN architectures and demonstrates robustness to input perturbations. Next follows an exploration on how to motivate the deep learning community to focus on real-world medical problems by presenting PCam, a dataset derived from the Camelyon16 challenge. PCam is structured to resemble common deep learning benchmarks and demonstrates that improvements on this dataset translate to improvements on the larger Camelyon16 benchmark.

Part II explores the benefits of self-supervised representation learning through Contrastive Predictive Coding (CPC) and proposes Contrastive Perturbative Predictive Coding ($C_2$PC) that enhances CPC’s performance by incorporating specific medical imaging augmentations.

Part III of the thesis addresses the challenge of estimating predictive uncertainty, crucial for high-risk medical decision-making. It introduces a novel variational inference method that leverages multinomial distributions over quantized latent variables. The proposed method exhibits competitive performance in uncertainty estimation and risk assessment compared to existing methods.

The thesis concludes that by addressing the identified challenges, deep learning can be better suited for medical imaging tasks. It demonstrates that expert knowledge can be effectively integrated into deep learning models, that leveraging unlabeled data through self-supervised learning can enhance model performance, and that predictive uncertainty can be improved with more flexible variational inference methods.
LIST OF PUBLICATIONS

This thesis is based on the following publications:


The majority of the work presented in these publications originate from the first author. Where multiple authors contributed equally, names are denoted with *. The other authors contributed in an advisory role, assisted with coding, writing and/or running experiments.

The author has further contributed to the following publications:


neural networks really?” In: *International Conference on Machine Learning (ICML)*.


# CONTENTS

## SUMMARY

## LIST OF PUBLICATIONS

1 INTRODUCTION

1.1 Motivation .................................................. 1
  1.1.1 Medical Decision Making .............................. 1
  1.1.2 Medical Imaging .................................. 2
  1.1.3 Computer Aided Diagnosis ......................... 3
  1.1.4 Deep Learning ................................... 3
  1.1.5 Re-evaluating DL for Medical Image Analysis .... 4

1.2 Setup .......................................................... 5

1.3 Notation ........................................................ 7

1.4 Basics of Deep Learning ................................. 7
  1.4.1 Data .................................................... 7
  1.4.2 Model .................................................. 8
  1.4.3 Optimization ...................................... 8

1.5 Probabilistic inference and Deep Learning .......... 9

I Integrating Expert Knowledge

2 ROTATION EQUIVARIANT CNNS FOR DIGITAL PATHOLOGY 17

2.1 Introduction ............................................... 17

2.2 Methods ..................................................... 19
  2.2.1 Background ........................................ 19
  2.2.2 G-CNN DenseNet architecture .................... 21

2.3 Experimental results ..................................... 22
  2.3.1 Datasets and Evaluation ............................ 22
  2.3.2 Model reliability .................................. 23
  2.3.3 P4M-DenseNet Performance ....................... 24

2.4 Conclusion .................................................. 26
# CONTENTS

## 3 PCAM

3.0.1 Why PCam ........................................ 27

## II Learning Without Labels

4 CONTRASTIVE PERTURBATIVE PREDICTIVE CODING ........................................ 33

4.1 Unsupervised Feature Extraction ........................................ 33

4.2 Contrastive Predictive Coding ........................................ 34

4.3 Contrastive Perturbative Predictive Coding ............................... 37

4.4 Related work ........................................ 37

4.5 Experiments ........................................ 38

4.5.1 Datasets ........................................ 38

4.5.2 PCam: histopathology malignancy classification ................. 39

4.5.3 Small-MURA: Bone X-ray anomaly detection .................. 42

4.6 Conclusions ........................................ 43

5 GREEDY INFORMAX ........................................ 45

5.1 Introduction ........................................ 45

5.2 Background ........................................ 46

5.3 Greedy InfoMax ........................................ 49

5.4 Experiments ........................................ 52

5.4.1 Vision ........................................ 52

5.4.2 Audio ........................................ 55

5.5 Related Work ........................................ 59

5.6 Conclusion ........................................ 60

APPENDICES ........................................ 63

5.A Experimental Setup ........................................ 63

5.A.1 Vision Experiments ........................................ 63

5.A.2 Audio Experiments ........................................ 64

## III Considering Confidence

6 PREDICTIVE UNCERTAINTY THROUGH QUANTIZATION .............................. 71

6.1 Method ........................................ 72

6.2 Related Work ........................................ 75

6.3 Results ........................................ 77

6.3.1 Main results ........................................ 79
CONTENTS | ix

6.3.2 Natural Images .................................................. 79
6.3.3 Analysis of latent variable distributions ............... 81
6.4 Discussion ............................................................ 84

APPENDICES 85
6.A Effect of hyper-parameters on coverage: ................. 85

7 CONCLUSION 89

BIBLIOGRAPHY 93

SAMENVATTING – SUMMARY IN DUTCH 109

ACKNOWLEDGEMENTS 111
1

INTRODUCTION

1.1 MOTIVATION

1.1.1 Medical Decision Making

Accurately diagnosing diseases and determining the most effective treatment is a challenge that has persisted throughout history. Historically, difficult cases were often attributed to the supernatural. Demons and divine intervention were the catch-all medical cause for ancient societies, and treatments were chosen accordingly. We have developed since, and the scientific method has prevailed in medicine. With a growing body of scientific knowledge, high evidence standards for interventions, and an array of modern diagnostic tools, medical doctors are now able to make informed treatment decisions like never before.

Diagnosing disease and predicting treatment outcome is not rocket science; it is significantly more complicated. The challenge stems from the fact that uncertainty clouds every aspect of medical decision making. Whilst we have decent deterministic models of gravity and rocket engines, the human body is still not completely understood. To this day, new organs are discovered that have implications for treatment (Benias et al., 2018; Valstar et al., 2021). Scientific evidence on test and treatment outcome is sparse and biased towards certain demographics. Research on intervention suffers from survivor-ship bias, where the justifiable desire for randomised controlled trials favor treatments that are amenable to double-blind study. The heterogeneity of humans complicates the equation, as does the associated human suffering and monetary cost with many diagnostic tools. Furthermore, the field constantly evolves, where interventions once hailed as the pinnacle of modern treatment have come into
question (Maron et al., 2020; Perera et al., 2022). Decision making within this mist of uncertainty perhaps demands a level of intelligence beyond what us mere mortals may muster manually.

1.1.2 Medical Imaging

This problem is further exacerbated by the advent of modern medical imaging. With technology such as the X-ray, ultrasound and later Computed Tomography (CT) and Magnetic Resonance Imaging (MRI), as well as advances in digitization of histopathology, clinicians can gain a dramatically growing amount of information on their patient. As a result, doctors are overloaded with data that needs to be considered in diagnosing a disease and predicting treatment outcome. Processing the unfamiliar imagery that these methods generate has proven to be of such complexity that specialised professions have arisen to assess and translating data into actionable advice, in the form of the Radiologist and Pathologist. Some of the tasks such experts must undertake are ones that humans are not naturally suited for. Often searching for a needle in a haystack, abstracted away from the human connection with a patient, these experts must stay focused on dense streams of data to find outliers and score risk. In efforts to standardize insights, human experts rely on subjective scales such as the Gleason score (Gleason and Tannenbaum, 1977 and Gleason, 1992) that have tremendous impact on treatment invasiveness, yet come with surprisingly high inter-rater variability (Ozkan et al., 2016). Additionally, it is well known that introducing more humans into a decision processes brings its own set of challenges in regards to communication and handover. And once a diagnosis has taken place, there is little to no feedback loop between an expert’s advice and the actual treatment outcome.

This is where the promise of Computer Aided Diagnosis (CAD) comes in. Computers have infinite energy, unlimited attention span and, noteworthy, do not show up to work hungover. Software is replicable and can be continuously improved. Erroneous behaviour can be re-enacted, studied and remediated. Feedback loops can span months with treatment outcome flowing back as guiding signals for early recognition models. A promising path forward to improve medical decision making.

While there are many aspects to the medical diagnostic process, encompassing the entire process is challenging. In this thesis, we focus on the challenge
of Computer Aided Diagnosis within the scope of interpreting medical imagery. This is a crucial step towards developing a comprehensive system that can digest information from all facets of the medical diagnostic process and make recommendations along the way.

1.1.3 Computer Aided Diagnosis

The process of distilling the expertise of the field of medical image analysis into algorithms has proven difficult. Deriving rules based on intuition and theory carefully honed over centuries of study and practice turns out to be nigh impossible, despite the armies of doctorates that have taken up the challenge.

Fortunately, machine learning provides an alternative to the rigid design of rule based modeling. The field provides proven methods that take in examples annotated by experts, and derives models that aim to capture the underlying principles with which these annotations were made. However, medical imaging data is especially vast and dense, and traditional machine learning approaches rely on hand crafted feature extracting to turn data into numbers that the methods can cope with. This hand crafting of features proves to be an almost equally difficult task as designing rules from the get go.

1.1.4 Deep Learning

This is why the advent of deep learning has brought a sea change in the field of medical image analysis. Through a combination of large randomly initialized feature extractors and stochastic gradient descent to traverse the search space, machine learning models can now be trained directly on raw dense imagery. This approach was popularized with the ImageNet challenge in 2012, where deep learning outperformed traditional machine learning approaches. Since then, most medical image analysis research communities have seen a shift to deep learning as the dominant paradigm.

The effectiveness of deep learning in medical image analysis inspires a promising vision for the future. Deep learning models can be trained on data from all over the world, rather than the handcrafted models designed by experts using small datasets from academic hospitals capturing only subsections of the demographics around the world. Moreover, deep learning models can learn from mistakes months after the initial diagnosis was made, a feedback loop
that happens rarely in practice with human experts. Although often thought of as a black box, a team of experts akin to flight-crash investigators can exactly reproduce the state in which a model made the error, and various tools can be used to provide introspection and guidance on how to mitigate similar mistakes in the future. Additionally, the free replication of deep learning model means expertise can be applied worldwide. This brings access to high quality medical insights to communities that would otherwise be lacking.

Deep learning models can change how primary care is delivered by removing the need for specialized expertise. This enables primary physicians to directly translate medical scans into actionable insights, thereby removing the burden of communication and reducing errors due to handover. Lastly, when a novel disease threatens the world, these models can be rapidly updated to incorporate new diagnoses, offering a nimble response to global health crises.

1.1.5 Re-evaluating DL for Medical Image Analysis

Despite its effectiveness, deep learning has been developed in an empirical manner – a process some equate to alchemy – and is still lacking fundamental first-principle justification. Whilst this has certainly brought tremendous success and progress, the dominant benchmarks that have driven year over year improvements have significant differences from the problems seen in medical imaging. These benchmarks are often single object detection in natural imagery, with a focus on classification accuracy, which misses the nuances of medical image analysis. That is not to say that deep learning has not shown its merit in this domain. It has lead to great success on impactful medical imaging benchmarks (Litjens et al., 2017a). To name a few early works, Gulshan et al., 2016 demonstrate a deep learning model capable of detecting diabetic retinopathy in retinal fundus photographs, comparable in performance to a panel of certified ophthalmologists. Esteva et al., 2017 demonstrate (near) human expert level performance on classifying skin cancer. Wang et al., 2016 demonstrate strong performance on identifying metastatic breast cancer from histopathology whole slide images of sentinel lymph node biopsies.

However, for a task as sensitive as medical decision making, the failure modes of deep learning models are not negligible (Zhang et al., 2016; Litjens et al., 2017a). Problems such as data bias, class imbalance, a lack of interpretability and explainability of decisions, input modality robustness and
1.2 Setup

We reevaluate the toolbox of deep learning with medical image analysis in mind, and find three main themes that require further study. The integration of expert knowledge on modeling constraints, the modeling and calibrated prediction of uncertainty, and the use of the vast amounts of unlabeled medical imaging data. These themes form the basis of the thesis. In part 1 we study the integration of expert knowledge in deep neural networks. In part 2 we study probabilistic modeling to improve calibrated uncertainty to aid decision making based on deep learning models. Finally, in part 3 we study how deep learning models can learn from unlabeled data to augment smaller labeled datasets. We ponder the following research questions throughout the thesis.

**Research Question 1:** Can roto-reflective equivariance improve accuracy and robustness of deep learning models in a large scale medical setting?

Deep learning is often studied in the natural image domain where images have a canonical orientation; the sky is up. Medical images and especially histopathological imagery, differ in this regard. Our main contribution is evaluating the use of group equivariant models studied by Cohen and Welling (2016) in the context of a large scale real world histopathology benchmark. It is hypothesised, but not proven, that this will aid performance. It has been argued that models benefit from less parameter sharing so the model can learn it itself. We explore if incorporating symmetries in rotation and mirroring using this framework of GCNN, aids the sample efficiency of deep learning models in a real world large scale medical imaging setting.

**Research Question 2:** How can we motivate the deep learning research community to optimize models for real world medical problems?

Next, we draw experience from the research from the previous question and derive a deep learning appropriate medical imaging benchmark, to enable researchers and practitioners to test new research ideas on a derived medical imaging task that correlates with real world impact.
Research Question 3: Do medical imaging deep learning models benefit from contrastive predictive coding for self-supervised representation learning?

There is no shortage of medical imaging data, as hospitals around the world capture vast amounts of data on a daily basis. These images are readily available in centralized storage systems called PACS, and modern academic hospitals have systems in place to process this data in an automated fashion. The challenge in medical data comes from collecting expert labels. This is an expensive process that requires input from multiple ‘gold standard’ experts per image to reach consensus. Leveraging unlabeled data is thus a promising approach for the field, as the ratio of labeled to unlabeled data is especially lopsided. In the second part of the thesis, we explore a novel technique for unsupervised representation learning, enabling the bulk of the deep learning model to be trained without downstream labels in the hopes of increasing label efficiency.

Research Question 4: Can we optimize deep learning models effectively without end-to-end backpropagation? Continuing on the topic of unlabelled data, we explore if unsupervised training can be done in a greedy fashion: training individual layers of a neural network without backpropagating through the full network. This form of training is of interest to medical image neural networks in particular. First, unsupervised training easy the labeling burden from medical experts. Second, The amount of memory required to train one layer at a time is orders of magnitudes smaller than training a full network. Finally, models can be trained on small crops from large medical imagery, which is especially impact-full when working with 3D or 4D data.

Research Question 5: Can latent variable quantization yield improved confidence scores in Deep Latent Variable Models? Medical decision making benefits from an accurate representation of the confidence in a certain prediction. As deep learning is prone to overconfidence, studying models that better capture uncertainty is highly relevant for medical imaging. In the third part part, we study the framework of variational inference in deep latent variable models, a form of probabilistic deep learning models. We test an often hypothesised assumption that improved flexibility of the variational posterior will lead to better performance, by introducing a highly flexible mean-field posterior that relies on quantizing the latent space.

Before we address these research questions in the main parts of this thesis, we will briefly introduce notation and the core principles of deep learning in
the following section. Finally, we will conclude and discuss directions of future work in chapter 7.

1.3 NOTATION

Throughout this work we follow the notation as presented in the table below.

<table>
<thead>
<tr>
<th>Examples</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a, b, c</td>
<td>Random scalar</td>
</tr>
<tr>
<td>x, y, z</td>
<td>Random vector</td>
</tr>
<tr>
<td>x_i</td>
<td>Elements of random vector</td>
</tr>
<tr>
<td>X, W, Y</td>
<td>Random matrices</td>
</tr>
<tr>
<td>X_i,j</td>
<td>Elements of random matrices</td>
</tr>
<tr>
<td>p(·)</td>
<td>Distributions (probability density functions)</td>
</tr>
<tr>
<td>q(·)</td>
<td>Often used for variational approximative distributions</td>
</tr>
</tbody>
</table>

1.4 BASICS OF DEEP LEARNING

Deep Learning refers to the modern practice of machine learning using stacked function approximators and the optimization of those using stochastic gradient descent. The term captures both the techniques as well as a broad spectrum of best practices surrounding the design, optimization and approach of machine learning. In the following parts we will briefly discuss the basic principles and practice of deep learning, which the expert reader can safely skip. The relevant background specific to each research question will be discussed in the following chapters.

1.4.1 Data

We start with considering the data. Ideally, we have gathered a large dataset of independently sampled, identically distributed (i.i.d.) data points $x$. We might have an accompanying label $y$ for each datapoint that we aim to predict. This results in a dataset $\mathcal{X} : \{(x_i, y_i \mid 0 \leq i \leq n)\}$ of $n$ datapoints. We divvy this dataset up in three parts: one to optimize our model on $D_{\text{train}}$, one to track
and tune the performance during training $D_{\text{valid}}$ and finally a set to estimate the expected performance on unseen data $D_{\text{test}}$. This separation ensures our model generalizes to the problem, instead of overfitting to the training data.

1.4.2 Model

A deep learning model is simply a parameterized mathematical function that accepts samples in the data space and transforms it into the label space: $f : \mathbb{R}_x \to \mathbb{R}_y$. Typically, this function is of a form of alternating linear transformations followed by scalar non-linearity’s (or activation functions): $f_{\theta}(x) = W^3h(W^2h(W^1x))$. This function is what is referred to as the neural network, as it is loosely inspired by the layers of connected neurons found in the brain. The parameters or weights $\theta : \{W^l\}$ are the knobs that we tune to find the optimal model.

1.4.3 Optimization

We optimizing the parameters of $f$ by alternatingly estimating the error that the model makes on the training data, and making small changes to the parameters to reduce this error using the derivative. This is the essence of Gradient Descent, and works as follows. We define a scalar error function (often called the loss) on the network $\mathcal{L} : \{X, Y, W\} \to \mathbb{R}$. To improve the model, we compute the derivative $\nabla \mathcal{L} = \frac{d}{dW_{i,j}} \mathcal{L}$ for each parameter. We then make a small step in the direction of steepest descent by updating each parameter:

$$W_{i,j}^l = W_{i,j}^l - \eta \nabla_{W_{i,j}^l} \mathcal{L},$$

where the step-size, or learning rate, is $\eta$. We apply this algorithm by drawing small semi-random subsets from the training data, called mini-batches. This has been shown to improve the generalization and reduces the computational cost. We refer to this stepwise optimization as Stochastic Gradient Descent (SGD), in contrast to Gradient Descent, which uses the full training datasets to compute the true gradient.
Learning rate

The learning rate $\eta$ is a hyperparameter that determines the size of the update to the weights at each step of the optimization process. If the learning rate is too large, the optimization algorithm may overshoot the minimum and leading to instability or divergence. On the other hand, if the learning rate is too small, the optimization algorithm may take a long time to converge to the minimum. Therefore, choosing an appropriate learning rate is crucial for successful training.

There are a few different strategies for choosing the learning rate, including fixed learning rates, learning rate schedules, and adaptive learning rates. Fixed learning rates involve setting a constant learning rate for the entire training process. Learning rate schedules involve gradually decreasing the learning rate over time. Adaptive learning rates involve dynamically adjusting the learning rate during training based on the progress of the optimization process.

There are also various techniques for monitoring the learning rate during training, such as learning rate schedules that reduce the learning rate if the validation loss stops improving, or methods that monitor the gradient to detect when the learning rate is too high or too low.

Overall, choosing an appropriate learning rate requires some experimentation and tuning, and there is no one-size-fits-all solution.

1.5 PROBABILISTIC INFERENCE AND DEEP LEARNING

In high-risk domains, prediction errors come at high costs. Luckily such domains often provide a fail-safe: self-driving cars perform an emergency stop, doctors run another diagnostic test, and industrial processes are temporarily halted. For deep learning models, this can be achieved by rejecting datapoints with a confidence score below a predetermined threshold. This way, a low error rate can be guaranteed at the cost of rejecting some predictions. However, estimating high quality confidence scores from neural networks, which create well-ordered...
rankings of correct and incorrect predictions, remains an active area of research.

Deep Latent Variable Models (DLVMs, fig. 1.1) approach this by postulating latent variables $z$ for which the uncertainty in $p(z|x)$ influences the confidence in the target prediction. Recently, efficient inference algorithms have been proposed in the form of variational inference, where an inference neural network is optimized to predict parameters of a variational distribution that approximates an otherwise intractable distribution (Kingma and Welling, 2013; Rezende et al., 2014; Alemi et al., 2016; Achille and Soatto, 2016).

In part III we explore how to quantify uncertainty in deep learning. We approach this through the lens of the Information Bottleneck (IB), as proposed by Tishby et al. (2000). The information bottleneck objective $I(y, z; \theta) - \beta I(x; z; \theta)$ is optimized to maximize the mutual information between $z$ and $y$, whilst minimizing the mutual information between $z$ and $x$. The objective can be efficiently optimized using a variational inference scheme as shown concurrently by Alemi et al. (2016) and Achille and Soatto (2016). Under the Markov assumption $p(z, x, y) = p(z|x)p(y|x)p(x)$, they derive the following lower bound:

$$I(y, z; \theta) - \beta I(x; z; \theta) \geq L = \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_{p(x_n)}[\log q_\theta(y_n|z)] - \beta D_{KL}[p_\theta(z|x_n)\|r(z)],$$

(1.1)

where the expectation with respect to $\mathbb{E}_{p(x_n)}$ is commonly estimated using a single Monte Carlo sample and $r(z)$ is a variational approximation to the marginal distribution $p(z)$. A common choice for $r(z)$ is a simple distribution such as a standard Normal. Alemi et al. (2016) and Achille and Soatto (2016) continue to show that the Variational Auto Encoder (VAE) Evidence Lower Bound (ELBO) proposed in Kingma and Welling (2013) and Rezende et al. (2014) is a special case of the IB bound when $y = x$ and $\beta = 1$:

$$I(z, x) - I(z, i) \geq \mathbb{E}_{p_\theta(z|x_n)}[\log q_\theta(x_n|z)] - D_{KL}[p_\theta(z|x_n)\|r(z)],$$

(1.2)

where $i$ represents the identity of data-point $x_i$. Interestingly, the VAE perspective considers the bound to optimize a variational distribution $q(z|x)$, whilst the IB perspective prescribes that $q(z|x)$ in the ELBO is not a variational pos-
terior but the true encoder $p(z|x)$, and instead $p(y|z)$ and $p(z)$ are the distributions approximated with variational counterparts. However, deriving the ELBO on a conditional distribution $p(y|x)$ leads to a KL divergence between the approximate posterior $q(z|x)$ and the real posterior $p(z|x)$, which is not available at test time, rather than the prior $p(z)$.

Alternatively, equation 1.1 can be interpreted as a domain-translating beta-VAE (Higgins et al., 2016), where an input image is encoded into a latent space and decoded into the target domain. The Lagrange multiplier $\beta$ then controls the trade-off between rate and distortion, as argued by Alemi et al. (2017). However, this does not naturally arise from deriving the ELBO using Jensen’s inequality on the conditional likelihood $p(y|x)$. 
Part I

Integrating Expert Knowledge
Before the advent of deep learning, medical image analysis was approached with classical methods using handcrafted features designed by experts. As deep learning became the go-to method of building image recognition models, expert knowledge has taken a step back. Due to successes of deep learning in the natural image domain (Krizhevsky et al., 2012), many early medical deep learning works have come to rely on the design decisions driven predominantly by their effect on the natural image domain. However, medical images are clearly different modalities and optimal applications of deep learning in this domain requires careful consideration of all decisions that determine a deep learning method’s design.

In this first part of the thesis, we will explore how expert knowledge about the symmetries of a medical image can be exploited by a neural network. Symmetries come about in various form, and the most typical example of a neural architecture that exploits symmetries is the convolutional neural network (CNN). CNNs exploit the fact that an image exhibits translational equivariance: a dog is a dog regardless of where it sits in the frame. Through the reuse of weights at all locations in the image, a properly designed CNN is constrained to honor this symmetry.

In medical image domains however, different symmetries might exist than prevalent in the natural domain. Such is the case with top-down microscopy: where as the sky is always up in the natural world, a microscopic image can be viewed under any rotation and mirroring without changing its semantic meaning. Such an image is called rotationally equivariant. In chapter 4, we explore how recent advancements in group equivariant neural networks can be applied to such medical data, with great effectiveness. By careful reuse of filters and their resulting feature maps, we will demonstrate neural networks that are equivariant to 90 degree rotations and reflections in the horizontal and vertical plane, and we will show how this improves sample efficiency and accu-
racy when applied to a large-scale real world challenge of detecting malignant tissue in lymph node biopsies.

We will close this part in chapter 3, where we present a novel dataset that we have proposed as part of this research project. This dataset distills a plug and play dataset for deep learning researchers to explore how their methods fare in a task that is proxy to a highly relevant real world medical problem.
2

ROTATION EQUIVARIANT CNNS FOR DIGITAL PATHOLOGY

2.1 INTRODUCTION

The field of digital pathology is developing rapidly, following recent advancements in microscopic imaging hardware that allow digitizing glass slides into whole-slide images (WSIs). This digitization has facilitated image analysis algorithms to assist and automate diagnostic tasks. A proven approach is to use convolutional neural networks (CNNs), a type of deep learning model, trained on patches extracted from whole-slide images. The aggregate of these patch-based predictions serves as a slide-level representation used by models to identify metastases, stage cancer or diagnose complications. This approach has been shown to outperform pathologists in a variety of tasks (Liu et al., 2017; Litjens et al., 2017; Bejnordi et al., 2016).

This performance is achieved using off-the-shelf CNN architectures originally designed for natural images (Litjens et al., 2017). The effectiveness of these models can be largely attributed to the efficient sharing of parameters in convolutional layers. As a result, local patterns are encoded independently of their spatial location, and shifting the input leads to a predictable shift in the output. This property, known as translational equivariance, effectively exploits the translational symmetry inherent in natural images leading to strong generalization.

In contrast to natural images, WSIs exhibit not only translational symmetry but rotation and reflection symmetry as well. CNNs do not exploit these symmetries, and as a result are found empirically to spend a large part of their parameter budget on multiple rotated and reflected copies of filters (Zeiler et
Additionally, we find that CNNs trained on histopathology data exhibit erratic fluctuations in predictions under input rotation and reflection. Enforcing equivariance in the model under these transformations is expected to reduce such instabilities, and lower the risk of overfitting by improving parameter sharing.

To encode these symmetries, we leverage recent findings in rotation equivariant CNNs (Cohen and Welling, 2016; Worrall et al., 2017; Weiler et al., 2017), a current topic of interest in the machine learning community. These methods show strong generalization under limited dataset size and are more robust under adversarial perturbations in rotation, translation and local geometric distortions (Dumont et al., 2018). We propose a fully-convolutional patch-classification model that is equivariant to $90^\circ$ rotations and reflection, using the method proposed by Cohen and Welling (2016). We evaluate the model on the Camelyon16 benchmark (Ehteshami Bejnordi et al., 2017), showing significant improvement over a comparable CNN on slide level classification and tumor localization tasks.

As slide-level metrics potentially obscure the relative performance of patch-level models, we further validate on a patch-level task. In this regime, there is currently no benchmark that harbors the high volume, quality and variety of Camelyon16. Thus, we present PatchCamelyon (PCam), a large-scale patch-level dataset derived from Camelyon16 data. Through this dataset, we demonstrate that the proposed model is more accurate and more robust under input rotation and reflection, compared to an equivalent standard CNN.

The contributions of this work are as follows: (1) we propose a novel deep learning model that utilizes symmetries inherent to histopathology\(^1\), (2) demonstrate that rotation equivariance improves model reliability and (3) present a new large-scale histopathology dataset that enables precise model evaluation.

**Related Work** A common approach to improve orientation robustness is to train CNNs using extensive data augmentation, perturbing data with random transformations (Liu et al., 2017; Litjens et al., 2017b). Although this may improve generalization, it fails to capture local symmetries and does not guarantee equivariance at every layer. As CNNs have to learn rotation equivariance from data, they require a larger model capacity to hold copies of identical filters. Even if rotation equivariance is achieved on training data, there is no

guarantee that this generalizes to a test set. Orthogonally, Liu et al. (2017) and Cireşan et al. (2013) propose a test-time augmentation strategy that averages the predictions of 90°-rotated and mirrored versions to improve robustness to orientation-induced instability. As a downside, this comes at 8 times the computational cost and does not provide guarantees on equivariance (Lenc et al., 2015).

Methods that enable equivariance under rotations and other transformations include Harmonic Networks (Worrall et al., 2017), which constrain the set of filters to circular harmonics, allowing for full 360°-equivariance. Weiler et al. (2017) employs steerable filters and evenly samples a small number of rotations. In this work, we focus on the straightforward G-CNN method from Cohen and Welling (2016) applied on discrete rotation/reflection groups. Although these groups do not cover the full continuous rotational symmetry inherent in WSIs, the empirical evidence gathered so far shows that 90° rotation equivariance improves performance significantly (Weiler et al., 2017).

2.2 METHODS

2.2.1 Background

In the mathematical model of CNNs and G-CNNs introduced in Cohen and Welling (2016), input images and output segmentation masks are considered to be functions \( f : \mathbb{Z}^2 \rightarrow \mathbb{R}^K \), where \( K \) denotes the number of channels, and \( f \) is implicitly assumed to be zero outside of some rectangular domain. A standard convolution\(^2\) (denoted \( \ast \)) of an input \( f \) with filter \( \psi \) is defined as:

\[
[f \ast \psi](x) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(y) \psi_k(x - y).
\]  

(2.1)

G-CNNs are a generalization of CNNs that are equivariant under more general symmetry groups, such as the group \( G = p4 \) of 90° rotations, or \( G = p4m \) which additionally includes reflection. In a G-CNN, the feature maps are thought of as functions on this group. For \( p4 \) and \( p4m \), this simply means that feature channels come in groups of 4 or 8, corresponding to the 4 pure

---

2 Technically, this is a cross-correlation
Figure 2.1: Given a canonical input and a rotated duplicate, we demonstrate how a 2-layer G-CNN is equivariant in \( p_4 \). Feature maps of one kernel per layer are shown, and the dashed blue arrows indicate how (intermediate) representations of the two inputs correspond. The \( \mathbb{Z}^2 \to p_4 \) convolution correlates the input with 4 rotated versions of the same kernel. The \( p_4 \to p_4 \) convolution correlates the resulting feature map with the \( p_4 \)-kernel, cyclically-shifting and rotating the kernel for each orientation. The final layer demonstrates how average-pooling over the orientations produces a representation that is locally invariant and globally equivariant to rotation. \textit{Global} average pooling over \( p_4 \) would result in a representation globally invariant to both translation and rotation.

rotations in \( p_4 \) or the 8 roto-reflections in \( p_{4m} \). In the first layer, these are produced using the \((\mathbb{Z}^2 \to G)\)-convolution:

\[
[f * \psi](g) = \sum_{y \in \mathbb{Z}^2} \sum_{k=1}^{K} f_k(y) \psi_k(g^{-1}y), \tag{2.2}
\]

where \( g = (r, t) \) is a roto-translation (in case \( G = p_4 \)) or roto-reflection-translation (in case \( G = p_{4m} \)).

In further layers, both feature maps and filters are functions on \( G \), and these are combined using the \((G \to G)\)-convolution:

\[
[f * \psi](g) = \sum_{h \in G} \sum_{k=1}^{K} f_k(h) \psi_k(g^{-1}h). \tag{2.3}
\]

In the final layer, a group-pooling layer is used to ensure that the output is either invariant (for classification tasks) or equivariant as a function on the plane (for segmentation tasks, where the output is supposed to transform together with the input). In Fig. 2.1 we demonstrate how equivariance is achieved
Figure 2.2: The proposed equivariant DenseNet architecture for the $p4$ group, consisting of 5 Dense Blocks (D.B.) alternated with Transition Blocks (T.B.). The final layer of the model is a $p4 \rightarrow Z^2$ group pooling layer followed by a sigmoid activation. The four orientations in $p4$ are illustrated through primary colors. A $Z^2 \rightarrow p4$ kernel (left), $p4 \rightarrow p4$ kernel (middle) and $p4 \rightarrow Z^2$ kernel (right) illustrate how equivariance arises in the model.

through this process. Non-linear activations and pooling operations are equivariant in $p4m$ (Cohen and Welling, 2016), allowing layers to be freely stacked to enable deep architectures.

2.2.2 G-CNN DenseNet architecture

The proposed patch-classification model is shown in Fig. 2.2 for $p4$ (the $p4m$-variant is a trivial extension). The architecture is based on the densely connected convolutional network (DenseNet) (Huang et al., 2016), which consist of dense blocks with layers that use the stack of all previous layers as input, alternated with transition blocks consisting of a $1 \times 1$ convolutional layer and $2 \times 2$ strided average pooling. We use one layer per dense block due to the limited receptive field of the model, with 5 dense-block/transition-block pairs. The model spatially-pools the input by a factor of $2^5$, the output of which resembles the segmentation resolution used in Liu et al. (2017).

Full-model group equivariance is achieved by replacing all convolution layers with group-equivariant versions (Cohen and Welling, 2016). Batch normalization layers (Ioffe and Szegedy, 2015a) are made group-equivariant by aggregating moments per group feature map rather than spatial feature map (as proposed by Cohen and Welling (2016)). Zero-padding is removed to prevent boundary-effects. The final layer consists of a group-pooling layer followed by
a sigmoid activation, resulting in tumor-probability output on the plane \( \mathbb{Z}^2 \).
As the model is fully convolutional, efficient inference can be achieved at test time by reusing computation of neighbouring patches, reducing segmentation time of a full WSI from hours to \( \sim 2 \) minutes on a NVIDIA Titan XP.

### 2.3 Experimental Results

#### 2.3.1 Datasets and Evaluation

To evaluate the proposed model, we use Camelyon16 (Ehteshami Bejnordi et al., 2017) and PCam. Additional testing is performed on BreakHis (Spanhol et al., 2016). (1) The Camelyon16 dataset contains 400 H&E stained WSIs of sentinel lymph node sections split into 270 slides with pixel-level annotations for training and 130 unlabeled slides for testing. The slides were acquired and digitized at 2 different centers using a 40\( \times \) objective (resultant pixel resolution of 0.243 microns). In the Camelyon16 challenge, model performance is evaluated using the FROC curve for tumor localization. (2) The PCam dataset contains 327,680 patches extracted from Camelyon16 at a size of 96 \( \times \) 96 pixels @ 10\( \times \) magnification, with a 75/12.5/12.5% train/validate/test split, selected using a hard-negative mining regime. (3) The BreakHis dataset contains 7909 H&E stained microscopy images at a size of 700 \( \times \) 460 pixels. The task is to classify the images into benign or malignant cases for multiple magnification factors. We limit our evaluation to the images at 4\( \times \) magnification, for which previous approaches have reported the highest accuracy (Spanhol et al., 2016; Song et al., 2017).

For the evaluation on the WSI-level Camelyon16 benchmarks, we largely follow the pipeline proposed in Liu et al. (2017), uniformly sampling WSIs and drawing tumor/non-tumor patches with equal probability. To prevent overrepresentation of background and non-tissue patches, slides are converted to HSV, blurred, and rejected if the max. pixel saturation lies below 0.07 (range [0,1]) and value above 0.1. This was empirically verified to not drop tissue patches. For computing the FROC score, tumor location candidates are selected with an efficient square non-maximum suppression window rather than radial. The window-size is tuned per model on the validation set. FROC score confidence bounds are computed using 2000 bootstrap samples Liu et al. (2017). Train and
2.3 EXPERIMENTAL RESULTS

Figure 2.3: (a) shows a large input region spanning multiple patches, with the tumor ground truth overlayed in green. The region is predicted under 32 evenly spaced sub-90° rotations, and prediction maps rotated back to original orientation. (b) shows the mean prediction and (c) shows the standard deviation of the predictions across all rotations, using DenseNet (left) and P_4M-DenseNet (right). Both networks are trained on the 12.5% data regime.

Validation splits are created by dividing the available WSIs randomly, maintaining equal tumor/normal ratio. We focus on the WSI data at 10× magnification (4 times smaller than the original dataset, at 0.972 microns per pixel) to fit the compute budget available for this work. Following Liu et al. (2017), we focus on the more-granular tumor-detection FROC metric in favor of slide-level AUC.

Training Details: Models are optimized using Adam (Kingma and Ba, 2014) with batch size 64 and initial learning rate 1e−3 (halved after 20 epochs of no improvement in validation loss). Epochs consists of 312 batches with a batch size of 64. Validation loss is computed using 40,000 sampled patches. Weights with lowest validation loss are selected for test evaluation.

2.3.2 Model reliability

We evaluate stability of predictions under rotation of the input. We present a visual analysis in Fig. 2.3. For a comparable baseline we use an equivalent model with standard convolutions. For a fair model comparison, we keep the number of parameters consistent by multiplying the growth rate of the baseline model by the square root of the group size. Bar the expected fluctuation around the tumor boundary (that arises due to the sub-sampled segmentation), the p4m-model is more robust to transformations even outside the group (sub-90° rotations). In addition, we observe a higher confidence for predictions inside the tumor regions for P_4M-DenseNet as compared to the baseline.
Table 2.1: Performance on PCam, measured by negative log-likelihood, accuracy and AUC. Experiments with additional data augmentation with $90^\circ$ rotations and reflections are marked by +. M indicates matching number of $Z^2$ maps, #W number of weights, $K$ number of $Z^2$ maps per layer.

<table>
<thead>
<tr>
<th>Network</th>
<th>$K$</th>
<th>#W</th>
<th>NLL</th>
<th>Acc</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>P4M-DenseNet</td>
<td>64</td>
<td>119K</td>
<td>0.260</td>
<td>89.8</td>
<td>96.3</td>
</tr>
<tr>
<td>P4M-DenseNet M</td>
<td>24</td>
<td>19K</td>
<td>0.273</td>
<td>89.3</td>
<td>95.8</td>
</tr>
<tr>
<td>P4-DenseNet</td>
<td>48</td>
<td>125K</td>
<td>0.329</td>
<td>89.0</td>
<td>94.5</td>
</tr>
<tr>
<td>DenseNet+</td>
<td>24</td>
<td>128K</td>
<td>0.306</td>
<td>88.1</td>
<td>95.1</td>
</tr>
<tr>
<td>DenseNet+ M</td>
<td>64</td>
<td>902K</td>
<td>0.365</td>
<td>87.2</td>
<td>94.6</td>
</tr>
<tr>
<td>DenseNet</td>
<td>24</td>
<td>128K</td>
<td>0.315</td>
<td>87.6</td>
<td>95.5</td>
</tr>
</tbody>
</table>

2.3.3 P4M-DenseNet Performance

PatchCamelyon (PCam)

We assess the performance of our main contribution, the P4M-DenseNet architecture, on the PCam dataset. Table 2.1 reports the performance. P4M-DenseNet outperforms other models, closely followed by the P4-DenseNet, indicating that both rotation and reflection are useful inductive biases, that can not be learned by data augmentation alone. Keeping the number of $Z^2$ maps fixed in the baseline degrades performance further, demonstrating the sample-efficiency of the P4M model.

Camelyon16

We evaluate our patch-based model on the slide-level tumor localization task of the Camelyon16 challenge. Fig. 2.4 reports the performance on the FROC score, next to those of a pathologist (Ehteshami Bejnordi et al., 2017) and the state-of-the-art approaches reported on this dataset, including Liu et al., 2017; Wang et al., 2016. For the baseline DenseNet, the training data is augmented with $90^\circ$ rotations and reflection. We experiment with multiple data regimes, where the number of WSIs in the training set is incrementally reduced by a factor of two.

The results indicate that the proposed method performs consistently better than all compared methods in terms of the FROC metric. Comparing to the baseline DenseNet results, we see that the superiority of our proposed architecture is predominantly due to the increased parameter sharing by the $p4m$-
### 2.3 Experimental Results

<table>
<thead>
<tr>
<th>Model</th>
<th>Data</th>
<th>FROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>P₄M-DenseNet</td>
<td>100%</td>
<td>84.0 (75.5, 91.5)</td>
</tr>
<tr>
<td>123k params</td>
<td>50%</td>
<td>81.5 (72.2, 89.3)</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>72.6 (58.7, 84.6)</td>
</tr>
<tr>
<td></td>
<td>12.5%</td>
<td>60.7 (46.0, 74.1)</td>
</tr>
<tr>
<td>DenseNet</td>
<td>100%</td>
<td>81.7 (72.1, 90.3)</td>
</tr>
<tr>
<td>126k params</td>
<td>50%</td>
<td>80.0 (69.3, 89.1)</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>71.0 (57.7, 82.0)</td>
</tr>
<tr>
<td></td>
<td>12.5%</td>
<td>55.4 (42.6, 68.5)</td>
</tr>
<tr>
<td>Liu et al. (2017) 10×</td>
<td>100%</td>
<td>79.3 (74.2, 84.1)</td>
</tr>
<tr>
<td>Wang et al. (2016)</td>
<td>100%</td>
<td>80.7*</td>
</tr>
<tr>
<td>Expert Pathologist</td>
<td>–</td>
<td>73.3</td>
</tr>
</tbody>
</table>

**Figure 2.4:** Performance on the Camelyon16 test set. The confidence bounds are obtained using a 2000-fold bootstrap regime. *Challenge winner (Wang et al., 2016) uses 40× resolution and is not directly comparable. Pathologist results from Ehteshami Bejnordi et al. (2017).*

Equivariance, which frees up model capacity and reduces the redundancy of detecting the same histological patterns in different orientations.

We also observe that the performance gap between our model and the baseline increases when we limit the dataset size by removing WSIs. This seems to indicate that the performance in the small-data regime benefits significantly from the sample efficiency of P₄M-DenseNet, with diminishing returns when the amount of data is sufficient for the baseline network to achieve (approximate) rotation equivariance. This performance gap remains for the full data set.

**BreakHis**

As an additional evaluation method, we assess the performance of the proposed model on the binary classification task of BreakHis as described in Section 2.3.1. As training the model from scratch is impractical given the small dataset, we pre-train on Camelyon16 at a similar pixel resolution. Similar to Spanhol et al. (2016), we predict the malignancy of a test image by using the maximum activation of 1000 random crops. We obtain an accuracy of 96.1 ± 3.2
and 93.5 ± 4.7 for P4M-Densenet and the baseline respectively, outperforming previous approaches (Spanhol et al., 2016; Song et al., 2017).

2.4 CONCLUSION

We present a novel histopathology patch-classification model that outperforms a competitive traditional CNN by enforcing rotation and reflection equivariance. A derived patch-level dataset is presented, allowing straightforward and precise evaluation on a challenging histopathology task. We demonstrate that rotation equivariance improves reliability of the model, motivating the application and further research of rotation equivariant models in the medical image analysis domain.

Acknowledgements We thank Geert Litjens, Jakub Tomczak, Dimitrios Mavroeidis and the anonymous reviewers especially for their insightful comments. This research was supported by Philips Research, the SURFSara Lisa cluster and the NVIDIA GPU Grant.
There exists a barrier-to-entry for machine learning researchers to evaluate their methods on large medical datasets. This is especially the case for histopathology, where large datasets such as CamelyonEhteshami Bejnordi et al., 2017 require developing intricate dataloaders, that take care of balancing different types of tissue, and can efficiently work with the terabytes of data available. With the hopes that we will make life a little easier for researchers looking to work with histopathology data, we are launching a dataset derived from Camelyon: PatchCamelyon (PCam). It was developed with input from pathologists and experts on deep learning for histopathology.

PCam is available now at [github.com/basveeling/pcam](https://github.com/basveeling/pcam) and is just as straightforward to work with as CIFAR10 or MNIST.

The PatchCamelyon benchmark is a new and challenging image classification dataset. It consists of 327,680 color images (96 x 96px) extracted from histopathologic scans of lymph node sections. Each image is annotated with a binary label indicating presence of metastatic tissue. PCam provides a new benchmark for machine learning models: bigger than CIFAR10, smaller than imagenet, trainable on a single GPU.

### Why PCam

Fundamental machine learning advancements are predominantly evaluated on straightforward natural-image classification datasets. Think MNIST, CIFAR, SVHN. Medical imaging is becoming one of the major applications of ML and we believe it deserves a spot on the list of go-to ML datasets. Both to challenge future work, and to steer developments into directions that are beneficial for this domain.
We think PCam can play a role in this. It packs the clinically-relevant task of metastasis detection into a straight-forward binary image classification task, akin to CIFAR-10 and MNIST. Models can easily be trained on a single GPU in a couple hours, and achieve competitive scores in the Camelyon16 tasks of tumor detection and WSI diagnosis. Furthermore, the balance between task-difficulty and tractability makes it a prime suspect for fundamental machine learning research on topics as active learning, model uncertainty and explainability.
Part II

Learning Without Labels
MOTIVATION AND SUMMARY

Medical data is often thought to be scarcely available, but this is actually far from the truth. The fact is that hospitals acquire medical imaging data at mind-boggling rates, and this data is already centrally stored in vast Picture Archiving and Communication Systems (PACS). Modern academic hospitals have systems in place that allow optimization of statistical models on data that was acquired that same day.

Rather, it’s the acquisition of high-quality labeling that is poses the main challenge, as well as the privacy considerations that come with distributing sensitive personal data. While the privacy problem can be solved with advancements in federated learning (Rieke et al., 2020) and differential privacy (Abadi et al., 2016). The cost of expert knowledge remains unsolved. Moreover, the lopsided distribution of experts and academic hospitals among varying demographics poses a risk of predictive bias towards ailments affecting affluent areas of the world’s population.

In chapter ?? we explore how self-supervised representation learning can benefit the medical domain.

In chapter 5 we explore how one can extend this unsupervised training towards data domains that are too large to fit in memory.
4

CONTRASTIVE PERTUBATIVE PREDICTIVE CODING

4.1 UNSUPERVISED FEATURE EXTRACTION

In recent years large quantities of medical imaging data have become available to develop data-based models. However, much of this data is unlabeled. Hence, there is a growing interest in unsupervised training of feature extraction models. Current methods rely on some form of generative modeling: an encoder/discriminator learns a representation, and a decoder/generator generates images. Both models are optimized to improve the generation of realistic images. In practice, it turns out that the effectiveness of these representations is limited in comparison to training the model fully supervised with a large labeled dataset.

Recently, an alternative form of unsupervised feature learning, called self-supervised learning, has garnished renewed interest. Rather than optimizing features such that a generative model can reproduce the original input, features are optimized such that a contrastive model can tell consecutive pairs of features apart from random pairs. In recent work Oord et al., 2018 this has been shown to extract features that are very effective for image classification, speaker identification and speech transcription. After training of the feature extractor only a small number of labels is required in order to train a linear classifier on top. This approach is commonly referred to as Contrastive Predictive Coding (CPC) Oord et al., 2018.

In this work, we explore how CPC performs on medical imaging data, focusing on the histopathology dataset PCam Veeling et al., 2018 and bone X-ray dataset MURA Rajpurkar et al., 2017. We find that the method suffers
Figure 4.1: (Zoom in on PDF for details). Histopathology features learned by a fully supervised model (top row), CPC (middle) and the proposed C₂PC method (bottom).

Left: 1024px² input image spanning multiple 32px² patches. Second-to-left: predicted malignancy probability map (brighter is more malignant). Right: The 8 most predictive feature maps, where each pixel represents the feature value for a single patch. C₂PC appears to learn strongly disentangled slow features that change smoothly over multiple patches, despite being extracted in isolation. In contrast, fully supervised features appear highly correlated and noisy.

from non-stationary noise signals found in this data, as well as from the domain shift between the training and test set. Therefore, we propose a method that perturbs each patch following common variation and realistic deformations commonly seen in medical images. In the following we will refer to this method as Contrastive Perpurbative Predictive Coding (C₂PC). We further explore the effect of drawing contrastive samples exclusively from inside the same sub-domain, e.g. data from a single patient, versus the complete dataset and find that the former improves upon the results further. Finally, we show that the linear predictive model trained on the C₂PC-derived features performs well even when exposed to just 1/512th of the available training data.

4.2 CONTRASTIVE PREDICTIVE CODING

Contrastive Predictive Coding (CPC) is a deep learning technique that provides universal unsupervised feature extraction, recently proposed by Oord et al. (2018a). It builds on fundamental work in areas of noise contrastive estimation and slow feature analysis Wiskott and Sejnowski, 2002. It has been shown to be effective across a variety of domains including image classification, speaker
and phone classification as well as reinforcement learning. CPC builds on the
notion that in time series, natural images and videos interesting features only
change slowly over time and space: so-called slow features. It differs from non-
linear ICA methods in that feature vectors are compared jointly, rather than per
feature, which removes the need for the model to learn conditionally indepen-
dent features. Furthermore, it increases the difficulty of the task by performing
detection of the consecutive patch among K-1 non-consecutive patches, rather
than binary classification of pairs of patches being consecutive or not.

Imagine small patches taken from larger input data: two consecutive patches
usually share a large part of the information that is relevant to a downstream
model. CPC exploits this as follows: starting with a patch of interest, a consec-
utive patch is put in a bag together with random patches from the dataset. A
contrastive classification model compares the patch of interest with each of the
patches in the bag, using a shared feature extraction model for the patches. It
incurs a loss when it fails to predict which patch in the bag is the consecutive
one. This loss is then optimized using conventional stochastic gradient descent
methods, training both the feature extraction model and the contrastive clas-
sification model in unison. This forces the feature extraction model to extract
features that distinguish consecutive patches and prevents it from learning
trivial features that do not provide information about the patches, such as lo-
cal nuisance factors and noise. The feature model can afterward be used to
train a model on downstream tasks.

A major downside of CPC is that the feature model will latch on to any
form of non-stationary signal that allows the contrastive model to efficiently
detect neighboring patches. If the data contains non-stationary artifacts, such
as chromatic aberration in a camera lens (which leaves a clear position signa-
ture in photos), the feature model might learn to encode these, rather than
more insightful features. We propose a solution to this issue in the following
section.

At the basis of CPC lies a typical deterministic feature extraction model
\( z = e(\cdot) \), e.g. a convolutional neural network with parameters \( \theta \), encoding
input patches \( x_t \) at timestep \( t \) into feature vectors \( z_t \): \( z_t = e_\theta(x_t) \). A simple log-
billinear model compares pairs of feature vectors and scores how likely these
features come from a pair of consecutive patches versus a pair of randomly selected patches:

\[ f_W(z_a, z_b) = \exp(z_a^T W z_b) \]  

(4.1)

We call this the contrastive model. The contrastive model is trained by taking one patch’s features \( z_t \) and a bag \( \mathcal{X} \) filled with the consecutive patch features \( z_t \) as well as randomly selected patch features drawn from the training dataset. The contrastive model \( f_W(\cdot) \) and feature model \( e_\theta(\cdot) \) are then jointly trained to solve the classification task of selecting the correct consecutive patch from the bag, minimizing the following loss:

\[ \mathcal{L} = -\log \frac{f(z_t, z_{t+1})}{\sum_{z_i \in \mathcal{X}} f(z_t, z_i)}, \]  

(4.2)

Which can be minimized using a typical mini-batch stochastic gradient descent method, using mini-batches consisting of multiple bags.

To further improve the predictive quality of the features we can contrast patches a varying number of steps ahead by jointly train multiple contrastive models. We can simply learn a model \( f_k(\cdot) \) for each step size \( k \), whilst using the same feature extraction model. As larger step sizes make the task more difficult, we introduce a context aggregation model \( c_t = a(z_0, \cdots, z_t) \) shared among the step sizes. This allows the contrastive model \( f_{W_k}(c_t, z_{t+k}) \) to consider a brief history of the signal, encoded into \( c_t \), to predict which patch \( z_i \) in the bag is the consecutive one. This results in the loss:

\[ \mathcal{L}_{CPC} = -\sum_k \log \frac{f_k(c_t, z_{t+k})}{\sum_{z_i \in \mathcal{X}_k} f(z_t, z_i)}, \]  

(4.3)

To extend the method to 2D images, we have to induce an arbitrary ordering on patches of an image. We can simply consider a top-down approach where we contrast with patches underneath the patch of focus. In the 2D regime the context aggregation model \( c = a(\cdot) \) can take shape of a small autoregressive PixelCNN-style neural network, where we are careful to maintain a receptive field of solely patches that are above and up to \( w \) steps to the side of the patch of focus, \( z_{y=i, x=j} \), resulting in \( c_{i,j} = a(z_{y\leq i, x=j\pm w}) \).
4.3 CONTRASTIVE PERTURBATIVE PREDICTIVE CODING

In order to alleviate the issue of CPC features collapsing to non-stationary nuisance signals, we utilize expert-driven data augmentation insights. The same data augmentation techniques are known to boost performance of a variety of deep learning based medical imaging methods, e.g Liu et al., 2017; Ronneberger et al., 2015.

We replace the deterministic input image $x$ with a random variable $\tilde{x} \sim r(x)$ drawn from a expert-determined random perturbation function $r(\cdot)$. $r(x)$ defines a distribution over the region in the input space around a datapoint $x$. This region is considered equivalent to $x$ in the context of the downstream prediction task. $r(\cdot)$ can potentially include color perturbation, elastic deformation, artificial measurement noise and artifacts, small translations and rotations, and more. We draw two samples $(\tilde{x}^a, \tilde{x}^b)$ from $r(x)$. The first sample is used to compute $c$ and the second sample to compute the features $z$ used by the contrastive model.

By being careful to only use augmentations that maintain the ordering of the patches, the feature extraction model is forced to learn features that are either invariant to the perturbations or can be consistently matched up by the linear contrastive model. This provides a simple vector for expert knowledge of symmetry modes to be injected in the feature extraction model, without the need to enforce these symmetries in the neural network using expensive equivariance methods Veeling et al. (2018).

4.4 RELATED WORK

In the field of medical image analysis, reducing the labeling burden required from experts is a recurring theme. A common technique when leveraging existing deep neural network architectures is using weights trained using ImageNet Rajpurkar et al., 2017. This approach is limited however, as the local and global statistics of medical imagery differ greatly from photos seen in ImageNet. Recent work training deep models on a large histopathology dataset find that ImageNet pretraining does not improve performance Liu et al., 2017.
Beyond borrowing weights, generative models have been used to pretrain feature models, using methods such as (Convolutional) Variational Auto Encoders (CVAE) Sedai et al., 2017 Uzunova et al., 2018 as well as their non-variational counterparts. Likewise, variations of GANs have been explored for pretraining and feature extraction Donahue et al., 2016. Another area of interest is the use of self-supervised losses, such as solving a jigsaw puzzle of patches drawn from an image Noroozi and Favaro, 2016 predicting color channels from gray-scaled images Larsson et al., 2016 Vondrick et al., 2018 and in-painting small windows removed from an input image Pathak et al., 2016.

Concurrently, Tellez et al., 2018 and Lu et al., 2019 have explored contrastive training for Whole Slide Images. While they only focus on histopathology data we show additional results on a bone X-ray anomaly detection task. Furthermore, we highlight the importance of extensive data augmentation during contrastive training, as well as the choice of the origin of the patches, see Table 4.1.

4.5 EXPERIMENTS

4.5.1 Datasets

We evaluate CPC and C2PC on two challenging large-scale medical imaging datasets: the histopathology dataset PCamVeeling et al., 2018 Ehteshami Bejnordi et al., 2017 and the bone X-ray anomaly detection task MURARajpurkar et al., 2017. In the following sections, we will demonstrate to what extent the CPC-learned features allow a linear classifier model to reach competitive performance on these two benchmarks. We will compare CPC against fully-supervised models, convolutional (variational) autoencoders (CVAEs, trained using a residual bilinear upsampling based decoder with L2 reconstruction loss) as well as randomly initialized feature models. Furthermore, we will explore if the classification model can cope with fewer labels when using the pretrained features.
Data augmentation & pre-processing

For the unsupervised pretraining methods, we use the following data augmentation methods:

- Color perturbation (contrast, hue, brightness and saturation).
- Elastic deformation.
- Per-channel elastic deformation (inducing artificial chromatic aberration).
- Random rotation up to 5 degrees.
- Random translation up to 8 pixels.
- For histopathology: invert colors such that background pixels are black and match zero-padding.

For training the predictive models, we increase random rotation up to 15 degrees and add random vertical and horizontal flips.

Hyperparameters

: We train the C(2)PC models using 4 GPUs with a batchsize of 256. For each bag $X$ we draw 128 contrastive patches randomly from the images available on one GPU. For experiments that only contrast inside one Whole Slide Image (WSI) (referred to as Same WSI), the batch consists of 16 groups of 16 PCam images sampled from one WSI, and contrastive patches are drawn only inside the group. All models use a 34-layer pre-activation resnetHe et al., [2016a] up to the pooling layer, with a tanh activation function on the 256 output features. No batch normalizationIoffe and Szegedy, [2015a] is used.

4.5.2 PCam: histopathology malignancy classification

PCam consists of 96px² histopathology images half of which contain malignant tissue. For CPC and C2PC we draw patches of 32px² with 50% overlap, leading to a grid of 5x5 patches. Each patch is with neighbors from 2 up to 4 steps below, skipping the immediate overlapping neighbor. A 4-layer PixelCNN-style model is used for the context aggregation model. For prediction, a global average pooling layer is used to aggregate the 5x5x256 feature map into a 1x256
Table 4.1: PCam classification accuracy on test set. C2PC achieves near-supervised performance, whilst the prediction model is simply a linear logistic regression model on top of unsupervised-extracted features. On small label regimes, the method outperforms fully supervised training. Results reported from a logistic regression model trained using the frozen pretrained feature model as input, with full data-augmentation. We evaluate the test set performance by training a logistic regression model on varying fractions of the 131,072 train labels, evaluating test performance using weights with the highest validation accuracy.
feature vector, followed by a logistic regression model. The feature model sees patches in isolation, and zero-padding is used to prevent information leaking between (non-overlapping) patches. Results are shown in Table 4.1 and discussed below.

Effect of different contrastive sample sources

There are large variations in the appearance of histopathology slides captured across different labs. Each lab has varying protocols with regards to the quantity of staining applied to the slides, and each slide digitizer brand comes with unique artifacts and measurement noise. Even between slides from the same lab, distribution shifts can occur. We explore if we can steer CPC towards more domain-robust features by drawing contrastive samples only from the same WSI. We hypothesize that in this scenario, it’s more cost effective for CPC to extract features that are useful across domains, as features that discriminate one domain from another do not provide any value. In Table 4.1 we compare how C(2)PC performs on both scenarios (Same WSI indicating sampling contrastive samples solely inside the WSI). We find that contrasting only with patches from the same WSI slightly increases performance.

C2PC versus CPC

C2PC enjoys a strong performance boost over CPC, which we attribute to the hypothesized noise and expert-knowledge based perturbation robustness. Test set performance on C2PC surprisingly increases slightly on the smaller label regimes. This can be explained as follows: The validation set of PCam lies much closer to the training set, and its performance drops slightly as expected in the smaller label regimes. With fewer labels, the model overfits the training labels more quickly. As a result, the early-stopping method selects weights that end up more favorable for the test set domain.

Feature visualization

In Figure 4.1 we visualize the 8 most predictive features extracted from a large input image spanning multiple patches. We find that CPC-based features appear much smoother than those learned in a fully supervised manner. This hints at the features being useful for other forms of tissue classification tasks as well.
### Table 4.2: Small-MURA Classification results. The AUROC on the validation set is reported, test set performance from the original MURA paper shown for reference.

<table>
<thead>
<tr>
<th>Method</th>
<th>AUROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>C2PC</td>
<td>71.8%</td>
</tr>
<tr>
<td>CPC</td>
<td>70.1%</td>
</tr>
<tr>
<td>CAE</td>
<td>55.6%</td>
</tr>
<tr>
<td>Random</td>
<td>60.8%</td>
</tr>
<tr>
<td>Fully Supervised</td>
<td>86.8%</td>
</tr>
<tr>
<td>Fully Supervised (No Batch Normalization)</td>
<td>57.7%</td>
</tr>
<tr>
<td>MURA DenseNet-169 (batch norm, ensemble)Rajpurkar et al., 2017</td>
<td>92.9%</td>
</tr>
</tbody>
</table>

4.5.3 Small-MURA: Bone X-ray anomaly detection

To improve tractability, we downsample the MURA dataset Rajpurkar et al., 2017 by a factor 2, resulting in images of 160px². For CPC we draw patches of 32px² with 50% overlap, leading to 9x9 patches per image. We contrast from 2 up to 6 steps below. For pretraining, batches consist of multiple studies, with one image randomly drawn per study. For the predictive model, we follow Rajpurkar et al. (2017)'s use of a weighted training loss and take the arithmetic mean of predicted anomaly probabilities from the images available in a study. A 34-layer ResNet [pre act ???] is used for the feature model.

Results are shown in Table 4.2. Due to resource constraints, the experiments are restricted to small models trained without batch normalization (except for the fully supervised model where batch normalization appears essential). This results in a lower supervised performance baseline compared to the large ensemble models participating in the MURA competition. Of the unsupervised feature extraction models evaluated, C2PC performs best. However, experiments on the original images with deeper feature extraction models remain in order. Comparing the results in Table 4.1 and Table 4.2 it becomes apparent that CPC is better suited for certain combinations of image types and downstream tasks. We argue that in the case of histopathology images CPC finds slow-changing, texture-based features that are predictive of malignancy. However, in case of the bone X-ray images it appears harder to learn meaningful slow features due to the heterogeneity of the extracted image patches.
4.6 CONCLUSIONS

We have explored the effectiveness of Contrastive Predictive Coding for medical image analysis. On the histopathology dataset PCam we find that CPC suffers from non-stationary artifacts and trivial solutions. We demonstrate that our proposed C2PC alleviates these issues, and outperforms fully supervised models in small label regimes. On the bone X-ray anomaly detection dataset MURA we outperform other pretraining baselines, which motivates further study of the use of CPC in this field. Having explored the use of C(2)PC as a pretraining method, further work could study how to effectively incorporate C2PC in a semi-supervised regime, exploiting both labels and expert equivalence class knowledge to achieve best-of-both-worlds results. Contrastive Predictive Coding provides a big leap forward in unsupervised pretraining. And with expert labels coming at high cost, the medical community stands to benefit tremendously from such advances.
Modern deep learning models are typically optimized using end-to-end backpropagation and a global, supervised loss function. Although empirically proven to be highly successful (Krizhevsky et al., 2012; Szegedy et al., 2015), this approach is considered biologically implausible. For one, supervised learning requires large labeled datasets to ensure generalization. In contrast, children can learn to recognize a new category based on a handful of samples. Additionally, despite some evidence for top-down connections in the brain, there does not appear to be a global objective that is optimized by backpropagating error signals (Crick, 1989; Marblestone et al., 2016). Instead, the biological brain is highly modular and learns predominantly based on local information (Caporale and Dan, 2008).

In addition to lacking a natural counterpart, the supervised training of neural networks with end-to-end backpropagation suffers from practical disadvantages as well. Supervised learning requires labeled inputs, which are expensive to obtain. As a result, it is not applicable to the majority of available data, and suffers from a higher risk of overfitting, as the number of parameters required for a deep model often exceeds the number of labeled datapoints at hand. At the same time, end-to-end backpropagation creates a substantial memory overhead in a naïve implementation, as the entire computational graph, including all parameters, activations and gradients, needs to fit in a processing unit’s working memory. Current approaches to prevent this require either the recomputation of intermediate outputs (Salimans and Bulatov, 2017) or expensive reversible layers (Jacobsen et al., 2018). This inhibits the application of deep learning models to high-dimensional input data that surpass current memory...
constraints. This problem is perpetuated as end-to-end training does not allow for an exact way of asynchronously optimizing individual layers (Jaderberg et al., 2017). In a globally optimized network, every layer needs to wait for its predecessors to provide its inputs, as well as for its successors to provide gradients. This forward and backward locking of the network caused by the backpropagation algorithm impedes the efficiency of hardware accelerator design due to a lack of locality.

In this paper, we introduce a novel learning approach, Greedy InfoMax (GIM), that improves upon these problems. Drawing inspiration from biological constraints, we remove end-to-end backpropagation by dividing a deep architecture into gradient-isolated modules that we train using a greedy, self-supervised loss per module. Given unlabeled high-dimensional sequential or spatial data, we encode it iteratively, module by module. By using a loss that enforces the individual modules to maximally preserve the information of their inputs, we enable the stacked model to collectively create compact representations that can be used for downstream tasks. Our contributions are as follows:

1. The proposed Greedy InfoMax algorithm achieves strong performance on audio and image classification tasks despite greedy self-supervised training.

2. This enables asynchronous, decoupled training of neural networks, allowing for training arbitrarily deep networks on larger-than-memory input data.

3. We show that mutual information maximization is especially suited for layer-by-layer greedy optimization, and argue that this reduces the problem of vanishing gradients.

5.2 Background

In order to create compact representations from data that are useful for downstream tasks, we assume that natural data exhibit so-called slow features (Wiskott and Sejnowski, 2002). It is theorized that such features are highly effective for downstream tasks such as object detection or speech recognition. To illustrate:

---

1 Our code is available at [https://github.com/loeweX/Greedy_InfoMax](https://github.com/loeweX/Greedy_InfoMax)
5.2 Background

For the self-supervised learning of representations, we stack a number of modules through which the input is forward-propagated in the usual way, but gradients do not propagate backward. Instead, every module is trained greedily using a local loss. (Right) Every encoding module maps its inputs $z_{t-1}^m$ at time-step $t$ to $g_{enc}^m(\text{GradientBlock}(z_{t-1}^m)) = z_t^m$, which is used as the input for the following module. The InfoNCE objective is used for its greedy optimization. This loss is calculated by contrasting the predictions of a module for its future representations $z_{t+k}^m$ against negative samples $z_j^m$, which enforces each module to maximally preserve the information of its inputs. We optionally employ an additional autoregressive module $g_{ar}$, which is not depicted here.

A patch of a few milliseconds of raw speech utterances shares information with neighboring patches such as the speaker identity, emotion, and phonemes, while it does not necessarily share these with random patches drawn from other utterances. Similarly, a small patch from a natural image shares many aspects with neighboring patches such as the depicted object or lighting conditions.

Recent work (Oord et al., 2018b; Hjelm et al., 2019) has proposed how we can exploit this to learn representations that maximize the mutual information shared among neighbors. In this work, we focus specifically on Contrastive Predictive Coding (CPC) (Oord et al., 2018b). This self-supervised end-to-end learning approach extracts useful representations from sequential inputs by maximizing the mutual information between the extracted representations of temporally nearby patches.

In order to achieve this, CPC first processes the sequential input signal $x$ using a deep encoding model $g_{enc}(x_t) = z_t$, and additionally produces a representation $c_t$ that aggregates the information of all patches up to time-step $t$. Figure 5.1: The Greedy InfoMax Learning Approach. (Left) For the self-supervised learning of representations, we stack a number of modules through which the input is forward-propagated in the usual way, but gradients do not propagate backward. Instead, every module is trained greedily using a local loss. (Right) Every encoding module maps its inputs $z_{t-1}^m$ at time-step $t$ to $g_{enc}^m(\text{GradientBlock}(z_{t-1}^m)) = z_t^m$, which is used as the input for the following module. The InfoNCE objective is used for its greedy optimization. This loss is calculated by contrasting the predictions of a module for its future representations $z_{t+k}^m$ against negative samples $z_j^m$, which enforces each module to maximally preserve the information of its inputs. We optionally employ an additional autoregressive module $g_{ar}$, which is not depicted here.
using an autoregressive model $g_{ar}(z_{0:t}) = c_t$. Then, the mutual information between the extracted representations $z_{t+k}$ and $c_t$ of temporally nearby patches is maximized by employing a specifically designed global probabilistic loss: Following the principles of Noise Contrastive Estimation (NCE) (Gutmann and Hyvärinen, 2010), CPC takes a bag $X = \{z_{t+k}, z_{j_1}, z_{j_2}, \ldots z_{j_{N-1}}\}$ for each delay $k$, with one “positive sample” $z_{t+k}$ which is the encoding of the input that follows $k$ time-steps after $c_t$, and $N-1$ “negative samples” $z_{j_n}$ which are uniformly drawn from all available encoded input sequences. Each pair of encodings $(z_j, c_t)$ is scored using a function $f(\cdot)$ to predict how likely it is that the given $z_j$ is the positive sample $z_{t+k}$. In practice, Oord et al. (2018b) use a log-bilinear model $f_k(z_j, c_t) = \exp(z_j^T W_k c_t)$ with a unique weight-matrix $W_k$ for each $k$-steps-ahead prediction. The scores from $f(\cdot)$ are used to predict which sample in the bag $X$ is correct, leading to the InfoNCE loss:

$$L_N = - \sum_k \mathbb{E}_X \left[ \log \frac{f_k(z_{i+k}, c_t)}{\sum_{z_j \in X} f_k(z_j, c_t)} \right]. \quad (5.1)$$

This loss is used to optimize both the encoding model $g_{enc}$ and the autoregressive model $g_{ar}$ to extract the features that are consistent over neighboring patches but which diverge between random pairs of patches. At the same time, the scoring model $f_k$ learns to use those features to correctly classify the matching pair. In practice, the loss is trained using stochastic gradient descent with mini-batches drawn from a large dataset of sequences, and negative samples drawn uniformly from all sequences in the minibatch. Note, that no min-max issues arise as found in adversarial training.
As a result of this configuration, one can derive that the optimal solution for \( f \) is proportional to the following density ratio (Oord et al., 2018b):

\[
f_k(z_{t+k}, c_t) \propto \frac{p(z_{t+k}|c_t)}{p(z_{t+k})}.
\]

(5.2)

This insight allows us to reformulate \(-\mathcal{L}_N\) as a lower bound on the mutual information \( I(z_{t+k}, c_t) \), as demonstrated in the appendix of Oord et al. (2018b) and proven by Poole et al. (2018). Minimizing the loss \( \mathcal{L}_N \) thus optimizes the mutual information between consecutive patch representations \( I(z_{t+k}, c_t) \), which in itself lower bounds the mutual information \( I(x_{t+k}, c_t) \) between the future input \( x_{t+k} \) and the current representation \( c_t \). Hyvarinen and Morioka, 2016 show that a similar patch-contrastive setup leads to the extraction of a set of conditionally-independent components, such as Gabor-like filters found in the early biological vision system.

**Layer-wise information preservation in neuroscience**

Linsker (1988) developed the InfoMax principle in 1988. It theorizes that the brain learns to process its perceptions by maximally preserving the information of the input activities in each layer. On top of this, neuroscience suggests that the brain predicts its future inputs and learns by minimizing this prediction error (Friston, 2010). Empirical evidence indicates, for example, that retinal cells carry significant mutual information between the current and the future state of their own activity (Palmer et al., 2015). Rao and Ballard (1999) indicate that this process may happen at each layer within the brain. Our proposal draws motivation from these theories, resulting in a method that learns to preserve the information between the input and the output of each layer by learning representations that are predictive of future inputs.

### 5.3 Greedy InfoMax

In this paper, we pose the question if we can effectively optimize the mutual information between representations at each layer of a model in isolation, enjoying the many practical benefits that greedy training (decoupled, isolated training of parts of a model) provides. In doing so, we introduce a novel approach for self-supervised representation learning: Greedy InfoMax (GIM). As depicted on the left side of Figure 5.1, we take a conventional deep learning
architecture and divide it by depth into a stack of $M$ modules. This decoupling can happen at the individual layer level or, for example, at the level of blocks found in residual networks (He et al., 2016). Rather than training this model end-to-end, we prevent gradients from flowing between modules and employ a local self-supervised loss instead, additionally reducing the issue of vanishing gradients.

As shown on the right side of Figure 5.1, each encoding module $g_{enc}^m$ within our architecture maps the output from the previous module $z_{t-1}^m$ to an encoding $z_t^m = g_{enc}^m(\text{GradientBlock}(z_{t-1}^m))$. No gradients are flowing between modules, which is enforced using a gradient blocking operator defined as GradientBlock$(x) \triangleq x, \nabla \text{GradientBlock}(x) \triangleq 0$. Oord et al. (2018) propose to use the output of an autoregressive model $g_{ar}(z_0:t) = c_t$ to contrast against future predictions $z_{t+k}$. However, our preliminary results showed that this did not improve results if applied at every module in the stack and optimizing it requires backpropagation through time, which is considered biologically implausible. Therefore, we train each module $g_{enc}^m$ using the following module-local InfoNCE loss:

$$f_k^m(z_{t+k}^m, z_t^m) = \exp(z_{t+k}^m^T W_k^m z_t^m)$$

$$L_N^m = - \sum_k \mathbb{E}_x \left[ \log \frac{f_k^m(z_{t+k}^m, z_t^m)}{\sum_{z_{j}^m \in X} f_k^m(z_{j}^m, z_t^m)} \right].$$

After convergence of all modules, the scoring functions $f_k^m(\cdot)$ can be discarded, leaving a conventional feed-forward neural network architecture that extracts features $z_t^M$ for downstream tasks:

$$z_t^M = g_{enc}^M \left( g_{enc}^{M-1} \left( \cdots g_{enc}^1(x_i) \right) \right).$$

For certain downstream tasks, a broad context is essential. For example, in speech recognition, the receptive field of $z_t^M$ might not carry enough information to distinguish phonetic structures. To provide this context, we reintroduce the autoregressive model $g_{ar}$ as an independent module that we optionally append to the stack of encoding modules, resulting in a context-aggregate representation $c_t^M = g_{ar}^M \left( \text{GradientBlock}(z_{0:t}^{M-1}) \right)$. In practice, a GRU or PixelCNN-style model can serve in this role. We train this module independently using the following altered scoring function:

$$f_k^M(z_{t+k}^{M-1}, c_t^M) = \exp \left( \text{GradientBlock} \left( z_{t+k}^{M-1} \right)^T W_k^M c_t^M \right).$$
5.3 Greedy InfoMax

Iterative Mutual Information Maximization

Similarly to the InfoNCE loss in Equation (5.1), our module-local InfoNCE loss in Equation (5.4) maximizes a lower bound on the mutual information $I(z_{m}^{t+k}, z_{m}^{t})$ between nearby patch representations, encouraging the extraction of slow features.

Most importantly, it follows from Oord et al. (2018), that the module-local InfoNCE loss also maximizes the lower bound of the mutual information $I(z_{m}^{t-1}, z_{m}^{t})$ between the future input to a module and its current representation. This can be seen as a maximization of the mutual information between the input and the output of a module, subject to the constraint of temporal disparity. Thus, the InfoNCE loss can successfully enforce each module to maximally preserve the information of its inputs, while providing the necessary regularization (Krause et al., 2010; Hu et al., 2017) for circumventing degenerate solutions. These factors contribute to ensuring that the greedily optimized modules provide meaningful inputs to their successors and that the network as a whole provides useful features for downstream tasks without the use of a global error signal.

Practical Benefits

Applying GIM to high-dimensional inputs, we can optimize each module in sequence to decrease the memory costs during training. In the most memory-constrained scenario, individual modules can be trained, frozen, and their outputs stored as a dataset for the next module, which effectively removes the depth of the network as a factor of the memory complexity.

Additionally, GIM allows for training models on larger-than-memory input data with architectures that would otherwise exceed memory limitations. Leveraging the conventional pooling and strided layers found in common network architectures, we can start with small patches of the input, greedily train the first module, extract the now compressed representation spanning larger windows of the input and train the following module using these.

Last but not least, GIM provides a highly flexible framework for the training of neural networks. It enables the training of individual parts of an architecture at varying update frequencies. When a higher level of abstraction is needed, GIM allows for adding new modules on top at any moment of the optimization process without having to fine-tune previous results.
Table 5.1: STL-10 classification results on the test set. The GIM model outperforms the CPC model, despite a lack of end-to-end backpropagation and without the use of a global objective. (± standard dev. over 4 training runs.)

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deep InfoMax (Hjelm et al., 2019)</td>
<td>78.2</td>
</tr>
<tr>
<td>Predsim (Nøkland and Eidnes, 2019)</td>
<td>80.8</td>
</tr>
<tr>
<td>Randomly initialized</td>
<td>27.0</td>
</tr>
<tr>
<td>Supervised</td>
<td>71.4</td>
</tr>
<tr>
<td>Greedy Supervised</td>
<td>65.2</td>
</tr>
<tr>
<td>CPC</td>
<td>80.5 ± 3.1</td>
</tr>
<tr>
<td><strong>Greedy InfoMax (GIM)</strong></td>
<td><strong>81.9 ± 0.3</strong></td>
</tr>
</tbody>
</table>

Table 5.2: GPU memory consumption during training. All models consist of the ResNet-50 architecture and only differ in their training approach. GIM allows efficient greedy training.

<table>
<thead>
<tr>
<th>Method</th>
<th>Memory (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td>6.3</td>
</tr>
<tr>
<td>CPC</td>
<td>7.7</td>
</tr>
<tr>
<td>GIM - all modules</td>
<td>7.0</td>
</tr>
<tr>
<td>GIM - 1st module</td>
<td>2.5</td>
</tr>
</tbody>
</table>

5.4 EXPERIMENTS

We test the applicability of the GIM approach to the visual and audio domain. In both settings, a feature-extraction model is divided by depth into modules and trained without labels using GIM. The representations created by the final (frozen) module are then used as the input for a linear classifier, whose accuracy scores provide us with a proxy for the quality and generalizability of the representations created by the self-supervised model.

5.4.1 Vision

To apply Greedy InfoMax to natural images, we impose a top-down ordering on 2D images. We follow Oord et al. (2018b) and Hénaff et al. (2019) by extracting a grid of partly-overlapping patches from the image to restrict the receptive fields of the representations. For each patch $x_{i,j}$ in row $i$ and column $j$ of this grid, we predict up to $K$ patches $x_{i+K,j}$ in the rows underneath, skipping the first overlapping patch $x_{i+1,j}$. Random contrastive samples are drawn with replacement from all samples available inside a batch, using 16...
contrastive samples for each evaluation of the loss. No autoregressive module \( g_{ar} \) is used for GIM in this regime.

**Experimental details** We focus on the STL-10 dataset (Coates et al., 2011) which provides an additional unlabeled training dataset. For data augmentation, we take random \( 64 \times 64 \) crops from the \( 96 \times 96 \) images, flip horizontally with probability 0.5 and convert to grayscale. We divide each image of \( 64 \times 64 \) pixels into a total of \( 7 \times 7 \) local patches, each of size \( 16 \times 16 \) with 8 pixels overlap. The patches are encoded by a ResNet-50 v2 model (He et al., 2016b) without batch normalization (Ioffe and Szegedy, 2015b). We split the model into three gradient-isolated modules that we train in sync and with a constant learning rate. After convergence, a linear classifier is trained – without finetuning the representations – using a conventional softmax activation and cross-entropy loss. This linear classifier accepts the patch representations \( z_{M}^{i,j} \) from the final module and first average-pools these, resulting in a single vector representation \( z^M \). Remaining implementation details are presented in Section 5.4.1.

**Results** As shown in Table 5.1, Greedy InfoMax (GIM) outperforms its end-to-end trained CPC counterpart, despite its unsupervised features being optimized greedily without any backpropagation between modules. An equivalent randomly initialized feature extraction model exhibits poor performance, showing that GIM extracts useful features. Training the feature extraction model end-to-end and fully supervised performs worse, likely due to the small size of the annotated dataset resulting in overfitting. Although this could potentially be circumvented through regularization techniques (DeVries and Taylor, 2017), the self-supervised methods do not appear to require regularization as they benefit from the full unlabeled dataset. Using a greedy supervised approach for training the feature model impedes performance, which suggests that mutual information maximization is unique in its direct applicability to greedy optimization.

In comparison with the recently proposed Deep InfoMax model from Hjelm et al. (2019) which uses a slightly different end-to-end mutual information maximization approach, AlexNet (Krizhevsky et al., 2012) as their feature-extraction model and an additional hidden layer in the supervised classification model, GIM comes out favorably. Finally, we see that we outperform the state-of-the-
art biologically inspired Predsim model from Nøkland and Eidnes (2019), which trains individual layers of a VGG like architecture (Simonyan and Zisserman, 2014) using two supervised loss functions.

In Figure 5.2, we visualize patches that neurons in intermediate modules of the GIM model are sensitive to. This demonstrates that modules later in the model focus on increasingly abstract features. Overall, the results demonstrate that complicated visual tasks can be approached using greedy self-supervised optimization, which can utilize large-scale unlabeled datasets.

**Asynchronous Memory Usage**  GIM provides a significant practical advantage arising from the greedy nature of optimization: modules can be trained in isolation given cached outputs from previous modules, effectively removing the depth of the network as a factor of the memory complexity. Measuring the allocated GPU memory of the previously studied models during training (Table 5.2), indicates that this theoretical benefit holds in practice as well. After splitting the architecture into three separately trainable modules, we can reduce the GPU memory consumption by a factor of 2.8 by training the modules asynchronously (GIM - 1st module) compared to training them simultaneously (GIM - all modules).

We evaluate whether training modules asynchronously influences the quality of the representations. Focusing on the extreme case, we optimize each module until convergence and fix its parameters, before we train the next module on top of it. This iteratively trained model achieves an accuracy of
Table 5.3: Results for classifying speaker identity and phone labels in the LibriSpeech dataset. All models use the same audio input sizes and the same architecture. Greedy InfoMax creates representations that are useful for audio classification tasks despite its greedy training and lack of a global objective.

\[a\] In the original implementation, Oord et al. (2018b) achieved 64.6% for the phone and 97.4% for the speaker classification task. \(b\) Baseline results from Oord et al. (2018b).

79.8% on the image classification downstream task. Thus, the performance declines slightly in comparison to the simultaneously trained model, as previously shown in Table 5.1 with 81.9% accuracy.

The training curves of the two models as shown in Figure 5.3 provide some insight into this decreased performance. The learning curves of the first module (Figure 5.3a) reflect that there is no difference in its training in the two models. Modules two and three (Figures 5.3b and 5.3c), however, reveal a crucial difference. The iteratively trained modules show a larger divergence between the training and validation loss, indicating stronger overfitting. We tentatively attribute this to the regularizing effect from the initially noisy inputs received by the higher modules when training simultaneously.

5.4.2 Audio

We evaluate GIM in the audio domain on the sequence-global task of speaker classification and the local task of phone classification (distinct phonetic sounds that make up pronunciations of words). These two tasks are interesting for self-supervised representation learning as the former requires representations
that discriminate speakers but are invariant to content, while the latter requires the opposite. Strong performance on both tasks thus suggests strong generalization and disentanglement.

**Experimental details** We follow the setup of Oord et al. (2018b) unless specified otherwise and use a 100-hour subset of the publicly available LibriSpeech dataset (Panayotov et al., 2015). It contains the utterances of 251 different speakers with aligned phone labels divided into 41 classes. These phone labels were provided by Oord et al. (2018b) who obtained them by force-aligning phone sequences using the Kaldi toolkit (Povey et al., 2011) and pre-trained models on Librispeech (Panayotov, 2014). We first train the self-supervised model consisting of five convolutional layers and one autoregressive module, a single-layer gated recurrent unit (GRU). After convergence, a linear multi-class classifier is trained on top of the context-aggregate representation \( c^M \) without fine-tuning the representations. Remaining implementation details are presented in appendix 5.A.2.

**Results** Following table 5.3, we analyze the performance of models on phone and speaker classification accuracy. *Randomly initialized* features perform poorly, demonstrating that both tasks require complex representations. The traditional, hand-engineered MFCC features are commonly used in speech recognition systems (Ganchev et al., 2005), and improve over the random features, but provide limited linear separability on both tasks. On the speaker classification task, CPC and GIM outperform the supervised baselines despite their feature models having been trained without labels, and GIM without end-to-end backpropagation. In this setting, both GIM and Greedy Supervised, where individual layers are trained greedily with a supervised loss function, achieve similar results to their respective end-to-end trained counterparts (CPC and Supervised). When classifying phones, CPC does not reach the supervised performance (64.9% versus 77.7%). GIM achieves 62.5%, while Greedy Supervised accomplishes 73.4%. Thus, in contrast to the vision experiments (Section 5.4.1), we see similar differences in performance between the greedily trained models (GIM and Greedy Supervised) when compared to their respective end-to-end optimized counterparts (CPC and Supervised).
<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Speaker Classification</strong></td>
<td></td>
</tr>
<tr>
<td>Greedy InfoMax (GIM)</td>
<td>99.4</td>
</tr>
<tr>
<td>GIM without BPTT</td>
<td>99.2</td>
</tr>
<tr>
<td>GIM without $g_{ar}$</td>
<td>99.1</td>
</tr>
<tr>
<td><strong>Phone Classification</strong></td>
<td></td>
</tr>
<tr>
<td>Greedy InfoMax (GIM)</td>
<td>62.5</td>
</tr>
<tr>
<td>GIM without BPTT</td>
<td>55.5</td>
</tr>
<tr>
<td>GIM without $g_{ar}$</td>
<td>50.8</td>
</tr>
</tbody>
</table>

Table 5.4: Ablation studies on the LibriSpeech dataset for removing the biologically implausible and memory-heavy backpropagation through time.

Overall, the discrepancy between better-than-supervised performance on the speaker task and less-than-optimal performance on the phone task suggests that GIM and CPC are biased towards extracting sequence-global features.

**Ablation Study** The local greedy training enabled by GIM provides a step towards biologically plausible optimization and improves memory efficiency. However, the autoregressive module $g_{ar}$ aggregates its inputs over multiple patches and employs Backpropagation Through Time (BPTT), which puts a damper on both benefits. In Table 5.4, we present results on the performance of ablated models that restrict the flow of gradients through time.

In order to limit the flow of gradients through time, we modify the autoregressive module. In general, the autoregressive module $g_{ar}$ takes the current input $z_t$, as well as the hidden state of the previous time-step $h_{t-1}$, in order to produce its output $c_t$, i.e. $c_t = g_{ar}(z_t, h_{t-1})$ (omitting the module-index $m$ here for brevity). In the standard GIM model, we block the flow of gradients to the previous module, such that $c_t = g_{ar}($GradientBlock$(z_t), h_{t-1})$. In the ablation **GIM without BPTT**, we remove BPTT by blocking the flow of gradients between time-steps, such that $c_t = g_{ar}($GradientBlock$(z_t), $GradientBlock$(h_{t-1}))$. For the ablation **GIM without $g_{ar}$**, we remove the autoregressive module entirely. Here, the linear classifier is applied to the representation created by the last encoding module (i.e. $z_t$).

In Table 5.4 we present the performance of the ablated models. Together, these two ablations indicate a crucial difference between the tested downstream
tasks. For the phone classification task, we see a steady decline of the performance when we reduce the modeling of temporal dependencies, indicating their importance for solving this task. When classifying the speaker identity, reducing the modeling of temporal dependencies in the ablated models barely influences their performance.

Together with the image classification results from Section 5.4.1 where no autoregressive module was employed either, this indicates that the GIM approach performs best on downstream tasks where temporal or context dependencies do not need to be modeled by an autoregressive module. In these settings, GIM can outperform the CPC model, which makes use of end-to-end backpropagation, a global objective, and BPTT.

INTERMEDIATE MODULE REPRESENTATIONS The greedy layer-wise training of GIM allows us to train arbitrarily deep models without ever running into a memory constraint. We investigate how the created representations develop in each individual module by training a linear classifier on top of each module and measuring their performance on the speaker classification task. With results presented in Figure 5.4, we first observe that each GIM module improves upon the representations of their predecessor. Interestingly, CPC exhibits similar performance in intermediate modules despite these modules relying solely on the error signal from the global loss function on the last module. This is in stark contrast with the supervised end-to-end model, whose intermediate
layers lag behind their greedily trained counterparts. This suggests that, in contrast to the supervised loss, the InfoMax principle “stacks well”, such that the greedy, iterative application of the InfoNCE loss performs similar to its global application.

5.5 RELATED WORK

We have studied the effectiveness of the self-supervised CPC approach (Oord et al., 2018; Hénaff et al., 2019) when applied to gradient-isolated modules, freeing the method from end-to-end backpropagation. There are a number of optimization algorithms that eliminate the need for backpropagation altogether (Scellier and Bengio, 2017; Lillicrap et al., 2016; Kohan et al., 2018; Balduzzi et al., 2015; Lee et al., 2015; Ororbia et al., 2018; Xiao et al., 2019). In contrast to our method, these methods employ a global supervised loss function and focus on finding more biologically plausible ways to assign credit to neurons.

A recently published work by Nøkland and Eidnes, 2019 likewise demonstrates that backpropagation-free layer-wise training is possible. Their similarity loss might be vaguely interpreted as another way of enforcing clustered representations. However, while our method achieves this entirely in a self-supervised fashion by clustering temporally or spatially nearby inputs, their similarity loss groups representations based on their class labels. Likewise, Belilovsky et al. (2019) showed that greedy layer-wise training with a supervised loss can scale to ImageNet. In an attempt to validate information bottleneck theory, Elad et al. (2018) develop a supervised, layer-wise training method that maximizes the mutual information between the outputs of a layer and the target whilst minimizing the mutual information between the inputs and outputs. In contrast to our proposal, these methods all rely on labeled data.

Jaderberg et al. (2017) develop decoupled neural interfaces, which enjoy the same asynchronous training benefits as Greedy InfoMax (GIM), but achieve this by taking an end-to-end supervised loss and locally predicting its gradients. Hinton et al. (2006) and Bengio et al. (2007) focus on deep belief networks and propose a greedy layer-wise unsupervised pretraining method based on Restricted Boltzmann Machine principles, followed by optimizing globally using a supervised loss. Lee et al. (2009) use convolutional deep belief networks
for unsupervised pretraining on the TIMIT audio dataset and then evaluate their performance by training supervised classifiers on top. Gao et al. (2018) and Ver Steeg and Galstyan (2015) explore total correlation explanation, which is related to mutual information maximization, and show that it can be applied for layer-by-layer training.

Several recent works investigated the utilization of mutual information maximization in a representation learning setting (McAllester, 2018; Oord et al., 2018; Hjelm et al., 2019; Belghazi et al., 2018). Poole et al. (2018) analyse these recent works under a common framework and highlight that InfoNCE exhibits low variance at a cost of high bias and propose new lower bounds that allow for balancing this bias/variance trade-off. However, the analysis of these improved bounds in the context of inter-patch mutual information optimization remains in order, and thus we focus on the original CPC InfoNCE loss to bias the learned representations towards slow features (Wiskott and Sejnowski, 2002).

Outside the information-theoretic framework, context prediction methods have been explored for unsupervised representation learning. A prominent approach in language processing is Word2Vec (Mikolov et al., 2013), in which a word is directly predicted given its context (continuous skip-gram). Likewise, Doersch et al., 2015 study such an approach for the visual domain. Similarly, graph neural networks use contrastive principles to learn unsupervised node embeddings based on their neighbors (Nickel et al., 2011; Perozzi et al., 2014; Nickel et al., 2015; Kipf and Welling, 2016; Veličković et al., 2018). Noise contrastive estimation has also been explored for independent component analysis (Hyvarinen et al., 2018; Hyvarinen and Morioka, 2016; Hyvarinen and Morioka, 2017). Schmidhuber, 1992 proposes a method where individual features are minimized such that they cannot be predicted from other features, forcing them to extract independent factors that carry statistical information, at the risk of neurons latching onto local independent noise sources in the input.

5.6 Conclusion

We presented Greedy InfoMax, a novel self-supervised greedy learning approach. The relatively strong performance demonstrates that deep neural net-
works do not necessarily require end-to-end backpropagation of a supervised loss on perceptual tasks. Our proposal enables greedy self-supervised training, which makes the model less vulnerable to overfitting, reduces the vanishing gradient problem and enables memory-efficient asynchronous distributed training. While the biological plausibility of our proposal is limited by the use of negative samples and within-module backpropagation, the results provide evidence that the theorized self-organization in biological perceptual networks is at least feasible and effective in artificial networks, providing food for thought on the credit assignment discussion in perceptual networks (Bengio et al., 2015; Linsker, 1988).
CHAPTER APPENDIX

5.A EXPERIMENTAL SETUP

We use PyTorch (Paszke et al., 2017) for all our experiments.

5.A.1 Vision Experiments

In our vision experiments, we employ the ResNet-50 v2 architecture (He et al., 2016b), in which we remove the max-pooling layer and adjust the first convolutional layer in such a way that the size of the feature map stays constant. Thus, the first convolutional layer uses a kernel size of 5, a stride of 1 and a padding of 2. Additionally, we do not employ batch normalization (Ioffe and Szegedy, 2015b).

We train our model on 8 GPUs (GeForce 1080 Ti) each with a minibatch of 16 images. We train it for 300 epochs using Adam (Kingma and Ba, 2014b) and a learning rate of 1e-4 and use the same random seed in all our experiments.

For the self-supervised training using the InfoNCE objective, we need to contrast the predictions of the model for its future representations against negative samples. We draw these samples uniformly at random from across the input batch that is being evaluated. Thus, the negative samples can contain samples from the same image at different patch locations, as well as from different images. We found that including the positive sample (i.e. the future representation that is currently to be predicted) in the negative samples did not have a negative effect on the final performance. For each evaluation of the InfoNCE loss, we use 16 negative samples and predict up to $k = 5$ rows into the future. For contrasting patches against one another, we spatially mean-pool the representations of each patch.

Before applying the linear logistic regression classifier on the output of the third residual block, we spatially mean-pool the created representations of size...
7 × 7 × 1024 again. Thus, the final representation from which we learn to predict class labels is a 1024-dimensional vector. We use the Adam optimizer for the training of the linear logistic regression classifier and set its learning rate to 1e-3. We optimized this hyperparameter by splitting the labeled training set provided by the STL-10 dataset into a validation set consisting of 20% of the images and a corresponding training set with the remaining images.

5.2 Audio Experiments

The detailed description of our employed architecture is given in Table 5.1. We train our model on 4 GPUs (GeForce 1080 Ti) each with a minibatch of 8 examples. Our model is optimized with Adam and a learning rate of 2e-4 for 300 epochs. We use the same random seed for all our experiments. Overall, our hyperparameters were chosen to be consistent with Oord et al. (2018b).

<table>
<thead>
<tr>
<th>Layer</th>
<th>Output Size</th>
<th>Parameters</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Sequence Length × Channels)</td>
<td>Kernel</td>
<td>Stride</td>
</tr>
<tr>
<td>Input</td>
<td>20480 × 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conv1</td>
<td>4095 × 512</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>Conv2</td>
<td>1023 × 512</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>Conv3</td>
<td>512 × 512</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Conv4</td>
<td>257 × 512</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Conv5</td>
<td>128 × 512</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>GRU</td>
<td>128 × 256</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 5.1: General outline of our architecture for the audio experiments.

For applying the InfoNCE objective on these layers, we randomly sample a time-window of size 128 to decrease the dimensionality.

Similarly to the vision experiments, we take the negative samples uniformly at random from across the batch that is currently evaluated. Again, this may include the positive sample. In our audio experiments, we use a total of 10 negative samples and predict up to $k = 12$ time-steps into the future.

We train the linear logistic regression classifier using the representations of the top, autoregressive module without pooling. Again, we employ the Adam
optimizer but select different learning rates than before. For this hyperparameter search, we split the training set provided by Oord et al. (2018b) into two random subsets using 25% of the samples as a validation set. In the speaker classification experiment, we used a learning rate of 1e-3, while we set it to 1e-4 for the phone classification experiment.
Part III

Considering Confidence
MOTIVATION AND SUMMARY

A difficult dimension of medical diagnosis is the inherent uncertainty associated with the diagnostic toolbox. Misdiagnosis is a major cost driver in the medical system, in both human and monetary terms (Kassei, 2015). A proper estimate of the confidence in a certain diagnosis can be of great help when making appropriate cost-benefit trade-offs that are associated with any further invasive diagnostic steps and interventions.

The holy grail in this domain is fully-calibrated uncertainty. Each prediction is associated with a probability score that reflects the frequentist view of how often this prediction would be correct. To achieve this, a full Bayesian treatment is required in the modeling of the problem. This incorporates prior knowledge, modeling uncertainty and data uncertainty. In this final part of the thesis, we explore probabilistic and Bayesian treatments of deep learning; an active area of research.
PREDICTIVE UNCERTAINTY THROUGH QUANTIZATION

Variational inference relies on a tractable class of distributions that can be optimized to closely resemble the true distribution (fig. 1.2), and it is hypothesized that more flexible classes lead to more faithful approximations and thus better performance (Jordan et al., 1999). To explore this hypothesis, we propose a novel tractable class of highly flexible variational distributions. Considering that neural networks with low-precision activations exhibit good performance (Holi and Hwang, 1993; Hubara et al., 2016), we make the modeling assumption that latent variables can be expressed under a strong quantization scheme, without loss of predictive fidelity. If this assumption holds, it becomes tractable to model a scalar latent variable with a flexible multinomial distribution over the quantization bins (fig. 6.1).

By re-positioning the variational distribution from a potentially limited description of moments, as found in commonly applied conjugate distributions, to a direct expression of probabilities per discrete value, a variety of benefits arise. As the support of the discrete distribution is constrained, the method does not suffer from normalization issues, relieving the model from batch normalization techniques (Ioffe and Szegedy, 2015a). Furthermore, interesting priors can be explored and the model is able to learn unique activation functions per neuron.

More concretely, the contributions of this work are as follows:

Figure 6.1: SQUAD quantizes the domain of $z$ to model a flexible and tractable variational distribution.
We propose a variational inference method that leverages multinomial distributions on quantized latent variables.

We show that the emerging predictive distributions are multi-modal, enabling flexible distributions in variational inference.

We demonstrate that the proposed method applied to the information bottleneck objective computes competitive uncertainty over the predictions and that this manifests in better performance under strong risk guarantees.

6.1 METHOD

In this work, we follow the IB interpretation of the bound in equation 1.1 and leave the evaluation of our proposed variational inference scheme in other models such as the VAE for further work. At the heart of our proposal lies the assumption that neural networks can be effective under strong activation quantization schemes. We start with presenting the derivation of the model in the context of a single latent-layer with an information bottleneck. Following

Figure 6.2: The left diagram visualizes the computational graph of SQUAD at training time, providing a detailed view on how an individual latent variable is sampled. The right diagram visualizes how the proposed matrix-factorization variant improves the parameter efficiency of the model.
the single data-point loss in equation 1.1 and dropping the subscript \( n \) for clarity, we obtain:

\[
\mathcal{L} = \mathcal{L}_{\text{error}} - \mathcal{L}_{\text{KL}} = \mathbb{E}_{p_{\theta}(z|x)}[\log q_{\theta}(y|z)] - \beta D_{\text{KL}}[p_{\theta}(z|x)\| r(z)]. \tag{6.1}
\]

To impose a flexible, multi-modal distribution over \( z \), we first make a mean-field assumption \( p(z|x) = \prod_k p(z_k|x) \). We then quantize the domain of each of the \( K \) scalar latent variables \( z_k \) such that only a small set of potential values remain: \( z_k \in v_1, \ldots, v_C \) \( \forall k \) with e.g. \( C = 11 \), see fig. 6.1. We use the same quantization scheme, defined by \( v \), for all the latents. The specific scheme is considered part of the prior, and is discussed in figure 6.3.

Although the number of values for each \( z_k \) is relatively small, the number of possible values that \( z \) can take on grows exponentially. We thus take to a monte-carlo estimation scheme, when sampling over all possible values of \( z \). To optimize the parameters \( \theta \) with Stochastic Gradient Descent (SGD), we need to derive a fully differentiable sampling scheme that allows us to sample values of \( z \). To formulate this, we re-parametrize the expectation over \( z \) in equation 6.1 using a set of variables \( s_k \in \{1, \ldots, C\} \) which index the value vector \( v \), allowing us to use a softmax function to represent the distribution over \( z \):

\[
p_{\theta}(z_k = v_c|x) \triangleq p_{\theta}(s_k = c|x) = \text{softmax}(Wx + b)c. \tag{6.2}
\]

These indexing values \( s \) are then used in conjunction with values \( v \) as in input for \( q_{\theta}(y|z) \), which is modelled with a small network \( f_{\theta}(\cdot) \): (abusing notation to indicate element-wise indexing with \( v[s] \)):

\[
\mathcal{L}_{\text{error}} = \mathbb{E}_{s \sim p_{\theta}}[\log f_{\theta}(v[s])].
\]

To enable sampling from the discrete variables \( s \), we use the Gumbel-Max trick (Gumbel, 1954), denoted \( \text{gumb}() \), re-parameterizing the expectation \( \mathbb{E}_{s \sim p} \) with uniform noise \( \epsilon \sim U(0,1) \):

\[
\mathcal{L}_{\text{error}} = \mathbb{E}_\epsilon \left[ \log f_{\theta}(v \left[ \text{arg max}_c \text{gumb}(p_{\theta}(s|x), \epsilon) \right]) \right]. \tag{6.3}
\]

As the argmax is not differentiable, we approximate this expectation using the Gumbel-Softmax trick (Maddison et al., 2016; Jang et al., 2016), which generates samples that smoothly deform into one-hot samples as the softmax temperature \( \tau \) approaches 0. Using the inner product (denoted \( \cdot \)) of the approximate one-hot samples and \( v \), we create samples from \( z \):

\[
\mathcal{L}_{\text{error}} \approx \mathbb{E}_\epsilon \left[ \log f_{\theta}(v \cdot \text{softmax gumb}(p_{\theta}(s|x), \epsilon)) \right]. \tag{6.4}
\]
In practice, we anneal $\tau$ from 1.0 to 0.5 during the training process, as proposed by Yang et al. (2017) to reduce gradient variance initially, at the risk of introducing bias.

To conclude our derivation, we use a fixed SQUAD distribution to model the variational marginal $r(\mathbf{z})$ as shown in figure 6.3. We can then derive the KL term analytically following the definition for discrete distributions. Using the fact that the KL divergence is additive for independent variables, we get our final loss:

$$L = \mathbb{E}_\epsilon \left[ \log f_\theta \left( \mathbf{v} \cdot \text{softmax} gumb \left( \mathbf{p}_\theta(\mathbf{s}|\mathbf{x}), \epsilon \right) \right) \right] - \beta \sum_{k=1}^{K} \sum_{c=1}^{C} p_\theta(s_k = c|\mathbf{x}) \log \frac{p_\theta(s_k = c|\mathbf{x})}{r(z_k = v_c)}.$$

For the remainder of this work, we will refer to the latent variables as $\mathbf{z}$ in lieu of $\mathbf{s}$, for clarity.

At test time, we can approximate the predictive function $p(\mathbf{y}^*|\mathbf{x}^*)$ for a new data-point $\mathbf{x}^*$ by taking $T$ samples from the latent variables $\mathbf{z}$ i.e. $\hat{\mathbf{z}}_t \sim p(\mathbf{z}|\mathbf{x}^*)$, and averaging the predictions for $\mathbf{y}^*$:

$$p_\theta(\mathbf{y}^*|\mathbf{x}^*) \approx \int q_\theta(\mathbf{y}^*|\mathbf{z}) p_\theta(\mathbf{z}|\mathbf{x}^*) d\mathbf{z} \approx \frac{1}{T} \sum_{t=1}^{T} q_\theta(\mathbf{y}^*|\hat{\mathbf{z}}_t).$$

A downside of mean-field variational approximations is that latents are assumed to be uncorrelated. We can bring some correlation to the latents, and extend the flexibility of the proposed model, by creating a hierarchical set of latent variables. We maintain a mean-field assumption for the prior, and the joint posterior distribution of $L$ layers of latents factorizes as follows:

$$p_\theta(\mathbf{z}_1, \ldots, \mathbf{z}_L|\mathbf{x}) = p_\theta(\mathbf{z}_L|\mathbf{z}_{L-1}) \cdots p_\theta(\mathbf{z}_1|\mathbf{x}),$$

With $q_\theta(\mathbf{y}|\mathbf{z}_1, \ldots, \mathbf{z}_L) = q_\theta(\mathbf{y}|\mathbf{z}_L)$. This is straightforwardly implemented with a simple ancestral sampling scheme.

Interestingly, the strong quantization proposed in our method can itself be considered an additional information bottleneck, as it exactly upper-bounds the number of bits per latent variable. Such bottlenecks are theorized to have a beneficial effect on generalization (Tishby et al., 2000; Achille and Soatto, 2016; Alemi et al., 2017; Alemi et al., 2016), and we can directly control this bottleneck by varying the number of quantization bins.

The computational complexity of the method, as well as the number of model parameters $\theta$, scale linearly in $C$, i.e. $O(C)$ (with $C$ the number of quantization bins). It is thus suitable for large-scale inference. We would like to
stress that the proposed method differs from work that leverages the Gumbel-Softmax trick to model categorical latent variables: our proposal models continuous scalar latent variables by quantizing their domain and modeling belief with a multinomial distribution. Categorical latent variable models would incur a much larger polynomial complexity penalty of $O(C^2)$.

The method is easily integrated into existing deep neural network architectures, and SQUAD layer implementations are provided at [github.com/anonymized](https://github.com/anonymized).

**Matrix-factorization variant** In the natural image domain, we anticipate a need for a high amount of information per variable, for which a small number of bins does not suffice. To improving the tractability of using a large number of quantization bins, we propose a variant of SQUAD that uses a matrix factorization scheme to improve the parameter efficiency. Formally, equation 6.2 becomes:

$$p(s_k = c | x) = \text{softmax}(W''_{k,c}(W'_k x + b'_k) + b''_{k,c}),$$

with full layer weights $W''$ and $W'$ respectively of shape $(K, B, C)$ and $(|X|, K, B)$, where $K$ denotes the number of neurons, $B$ the number of factorization inputs, $C$ number of quantization bins and $|X|$ the input dimensionality. To improve the parameter efficiency, we can learn $W'$ per layer as well, resulting in shape $(|X|, 1, B)$, which is found to be beneficial for large C by the hyper-parameter search presented in section 6.3. We depict this alternative model on the right side of figure 6.2 and will refer to it as SQUAD-factorized. We leave further extensions such as Network-in-Network (Lin et al., 2013) for future work.

### 6.2 Related Work

Outside the realm of DLVMs, other methods have been explored for predictive uncertainty. Lakshminarayanan et al. (2017) propose deep ensembles: straightforward averaging of predictions from a small set of separately adversarially trained DNNs. Although highly scalable, this method requires retraining a model up to 10 times, which can be inhibitably expensive for large datasets.

Gal and Ghahramani (2015b) propose the use of dropout (Srivastava et al., 2014) at test time (Monte Carlo Dropout, or MC-Dropout) and present a Bayesian neural network interpretation of this method. A follow-up work by Gal et al.
Figure 6.3: In the IB bound, the marginal \( p(z) \) is approximated with a fixed distribution \( r(z) \). Using our proposed SQUAD distribution we can impose a variety of interesting forms for \( r(z) \) via the spacing \( v \) and weighting \( r(z_k = v_c) \) of the quantization bins. For the values \( v \), we compare linearly spaced bins (left) versus bins with equal probability mass under a normal distribution (right). Furthermore, we explore the effect of allowing the bin values to be optimized with SGD on a per-neuron or per-layer basis, to allow the model to optimize the quantization scheme with the highest fidelity. For the prior probabilities, we explore a uniform prior (left) and probability mass of the bins under a normal distribution (middle).

(2017) explores the use of Gumbel-Softmax to smoothly deform the dropout noise to allow optimization of the dropout rate during training. A downside of MC-Dropout (MCD) is the limited flexibility of the fixed bi-modal delta-peak distribution imposed on the weights, which requires a large number of samples for good estimates of uncertainty. Oord et al. (2017a) propose the use of vector quantization in variational inference, quantizing a multi-dimensional embedding, rather than individual latent variables, and explore this in the context of auto-encoders.

In the space of learning non-linearity’s, Su et al. (2017) explore a flexible non-linearity that can assume the form of most canonical activations. More flexible distributions have been explored for distributional reinforcement learning by Dabney et al. (2017a) using quantile regression, of which can be seen as a special case of SQUAD where the bin values are learned but have fixed uniform probability. Categorical distributions on scalar variables have been used to model more flexible Bayesian neural network posteriors as by Shayer et al. (2017). The use of a mixture of diracs distribution to approximate a variety of distributions was proposed by Schrempf et al. (2006).
Figure 6.4: Risk/coverage curve (with log-axes for discernibility) of 2-layer models on notMNIST. Lines closer to the lower-right are better. As the selective classifier lowers the confidence threshold, coverage increase at the cost of greater classification risk. Area around curves represents 90% confidence bounds computed using 10 initializations/splits.

6.3 RESULTS

Quantifying the quality of uncertainty estimates of models remains an open problem. Various methods have been explored in previous works, such as relative entropy (Louizos and Welling, 2017; Gal and Ghahramani, 2015a), probability calibration, and proper scoring rules (Lakshminarayanan et al., 2017). Although interesting in their own right, these metrics do not directly measure a good ranking of predictions, nor indicate applicability in high-risk domains. Proper scoring rules are the exception, but a model with good ranking ability does not necessarily exhibit good performance on proper scoring rules: any score that provides relative ordering suffices and does not have to reflect true calibrated probabilities. In fact, well-ranked confidence scores can be re-calibrated (Niculescu-Mizil and Caruana, 2005) after training to improve performance on proper scoring rules and calibration metrics.

In order to evaluate the applicability of the model in high-risk fields such as medicine, we want to quantify how models perform under a desired risk guarantee. We propose to use the Selection with Guaranteed Risk (SGR) method\(^1\)

\(^1\) We deviate slightly from Geifman and El-Yaniv (2017) in that we use Softmax Response (SR) – the probability taken from the softmax output for the most likely class – as the confidence score for all methods. Geifman and El-Yaniv (2017) instead proposed to use the variance of the probabilities for MCDropout, but our experiments showed that SR paints MCDropout in a more favorable light.
introduced by Geifman and El-Yaniv [2017] to measure this. In summary, the SGR method uses a confidence score, e.g. the estimated probability of the prediction, and rejects the prediction if this score is below a certain threshold. This threshold is optimized on a hold-out set to the tightest possible value that still guarantees a desired minimal error rate with high probability (e.g. 99%). SGR is formally proven, and empirically shown, to ensure this minimal error rate under i.i.d. assumptions. We propose to use SGR as evaluation metric for confidence estimation methods. Specifically, on a hold-out set we measure the percentage of predictions that are accepted by SGR under a specific risk guarantee, i.e. the coverage. To illustrate, one can imagine that a trivial solution to guarantee no errors is to reject all data-points, which has 0% coverage. By lowering the confidence threshold, coverage goes up, but the accuracy of the method goes down. SGR finds the optimal threshold that ensures the accuracy does not drop below a predetermined threshold, and by measuring the coverage under this threshold we can effectively compute the real world value of a confidence estimation methods.

To limit the influence of hyper-parameters on the comparison, we use a extensive hyper-parameter scheme on all variants and baselines. Specifically we use the automated optimization method TPE (Bergstra et al., [2011]) over a maximum of 1000 evaluations per model per experiment. The hyper-parameters are optimized for coverage at 2% risk ($\delta = 0.01$) on fashionMNIST (Xiao et al., [2017]), and we evaluate the optimal hyperparameters on another dataset, notMNIST, to measure how sensitive the hyperparameters are to a change in the data-distribution. Larger models are evaluated on Street View Housing Numbers (SVHN) (Netzer et al., [2011]) using the same optimization scheme. For the details on the evaluated search space and chosen hyper-parameters we refer the reader to the appendix.

We compare our model against plain Multi-Layer Perceptrons (MLPs) with ReLU activations, MCDropout using Maxout activations (Goodfellow et al., [2013]) Chang and Chen, [2015]) and an information bottleneck model using mean-

---

2 All models are optimized with ADAM (Kingma and Ba, [2014a]), weight initialization as proposed by He et al. ([2015]), a weight decay of $10^{-5}$ and adaptive learning rate decay scheme $\times$ 10ox reduction after 10 epochs of no validation accuracy improvement $\parallel$ and use early stopping after 20 epochs of no improvement.

3 We found this baseline to perform stronger in comparison to conventional ReLU MCdropout models, under equal number of latent variables.
field Gaussian distributions. We evaluate the complementary deep ensembles technique (Lakshminarayanan et al., 2017) for all methods.

6.3.1 Main results

We start our analysis by comparing the predictive uncertainty of 2-layer models with 32 latent variables per layer. In figure 6.4 we visualize the risk/coverage trade-off achieved using the predicted uncertainty as a selective threshold, and present coverage results in table 6.1. Overall, we find that SQUAD performs significantly better than plain MLPs and deep Gaussian IB models, and we tentatively attribute this to the increased flexibility of the multinomial distribution. Compared to a Maxout MCdropout model with a similar number of weights, SQUAD appears to have a slight though not significant advantage, despite the strong quantization scheme, especially at low risk guarantees. Deep ensembles improve results for all methods, which fits the hypothesis that ensembles integrate over a form of weight uncertainty. When comparing the optimal hyperparameters found for fashionMNIST on the similar notMNIST dataset, we find that SQUAD shows strong performance, as shown in table 6.2. This provides some evidence that despite the increase in number hyperparameters in SQUAD, the optimal settings are more robust to a change in the data distribution.

6.3.2 Natural Images

To explore larger models trained on natural image datasets, we re-tune hyperparameters on 256-latent 2-layer models over 100 TPE evaluations. As SVHN contains natural images in color, we anticipate a need for a higher amount of information per variable. We thus explore the effect of the matrix-factorized variant.

As shown in table 6.3, SQUAD-factorized outperforms the non-factorized variant. Considering the computational cost at the optimum of a 4-neuron factorization ($B = 4$) with $C=37$ quantization bins, the model clocks 3.4 million weights. In comparison, the optimum for the presented MCdropout results has $C=11$, using 9.0 million weights. On an NVIDIA Titan Xp, the dropout baseline takes 13s per epoch on average, while SQUAD-factorized spans just 9s.
**Table 6.1**: SGR coverage results on Fashion MNIST. We present coverage percentage of the test dataset for three pre-determined risk-guarantees (one per column), where higher coverage is better, as well as negative log-likelihood and overall accuracy. The results indicate that SQUAD provides competitive uncertainty, especially at low-risk guarantees. Bayesian approximations via deep ensembles improve coverage all over the board, and for SQUAD in particular. When SGR can not guarantee the required risk level at high probability, 0% coverage is reported. (*2 std. deviations shown in parentheses, optimal results in bold.*)

<table>
<thead>
<tr>
<th>Model</th>
<th>cov@risk .5%</th>
<th>cov@risk 1%</th>
<th>cov@risk 2%</th>
<th>NLL</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain MLP</td>
<td>29.1 (±20.71)</td>
<td>45.9 (±4.04)</td>
<td>60.4 (±3.17)</td>
<td>0.408 (±0.036)</td>
<td>87.7 (±42)</td>
</tr>
<tr>
<td>Maxout MCDropout</td>
<td>41.9 (±9.86)</td>
<td>56.5 (±2.30)</td>
<td><strong>69.9</strong> (±1.48)</td>
<td>0.299 (±0.008)</td>
<td><strong>89.5</strong> (±28)</td>
</tr>
<tr>
<td>DLGM</td>
<td>0.0 (±0.00)</td>
<td>33.5 (±2.42)</td>
<td>47.0 (±1.47)</td>
<td>0.446 (±0.007)</td>
<td>84.3 (±15)</td>
</tr>
<tr>
<td>SQUAD</td>
<td><strong>42.9</strong> (±7.19)</td>
<td><strong>58.3</strong> (±3.06)</td>
<td>69.5 (±1.55)</td>
<td><strong>0.293</strong> (±0.008)</td>
<td><strong>89.5</strong> (±35)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>cov@risk .5%</th>
<th>cov@risk 1%</th>
<th>cov@risk 2%</th>
<th>NLL</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain MLP Ensemble</td>
<td>40.6</td>
<td>58.3</td>
<td>70.2</td>
<td>0.296</td>
<td>89.3</td>
</tr>
<tr>
<td>Max. MCD. Ensemble</td>
<td><strong>48.2</strong></td>
<td>59.1</td>
<td>72.2</td>
<td><strong>0.271</strong></td>
<td><strong>90.2</strong></td>
</tr>
<tr>
<td>DLGM Ensemble</td>
<td>0.0</td>
<td>34.3</td>
<td>47.8</td>
<td>0.435</td>
<td>84.7</td>
</tr>
<tr>
<td>SQUAD Ensemble</td>
<td>47.5</td>
<td><strong>61.6</strong></td>
<td><strong>73.1</strong></td>
<td>0.273</td>
<td>90.1</td>
</tr>
</tbody>
</table>

**Table 6.2**: SQUAD exhibits strong performance on notMNIST using the optimal hyperparameters found for fashionMNIST.
Table 6.3: Results on SVHN indicate that the quantization scheme imposed by SQUAD models might hinder performance, but that this is effectively compensated by the SQUAD-factorized variant using a larger amount of bins. Even with \( T = 4 \) MC samples at test time, SQUAD performs well.

To evaluate the sample efficiency of the methods, we compare results at \( T = 4 \) samples. We find that SQUAD’s results suffer less from under-sampling the predictive distribution than MCdropout. We tentatively attribute the sample efficiency to the flexible approximating posterior on the activations, which is in stark contrast to the rigid approximating distribution that MCdropout imposes on the weights. In conclusion, SQUAD comes out favorably in a resource-constrained environment.

6.3.3 Analysis of latent variable distributions

In order to evaluate if the proposed variational distribution does not simply collapse into single mode predictions, we want to find out what type of distributions the model predicts over the latent variables. We visualize the forms of predicted distributions in figure 6.5. Although this showcases only a small subset of potential multi-modal behavior that emerges, this demonstrates that the model indeed utilizes the distribution to its full potential. To provide an intuition on how these predicted distributions emerge, we present figure 6.7 in the appendix.

In figure 6.6 we visualize one of the activation functions that the method learns for a 1-dimensional input SQUAD-factorized model. The learned activation functions resemble “peaked” sigmoid activations, which can be inter-
Figure 6.5: By analyzing the emerging predicted distributions of individual neurons in a converged SQUAD model, we find that the flexible variational distribution is used to its full advantage. Figure (A) visualizes a subset of interesting stereotypical distributions we hope to find in the model. Figure (B) summarizes distributions predicted by the model similar to stereotypes, discovered by looking at predicted distributions with low KL. Figure (C) shows how often distributions similar to stereotypes arise, as measured by the KL distance (lower KL is closer to stereotypes).
Figure 6.6: By using a 1-dimensional matrix factorization for a SQUAD-factorized distribution, we can visualize the type of (stochastic) activation functions learned by the method. After training the model as usual on fashion-MNIST, we take a random neuron from the first layer. We visualize how the predicted distribution of the output changes as a function of the 1-dimensional input. The left y-axis indicates the probability per value as shown using the green line. The right y-axis indicates the value and in blue the most likely value is shown, and the gray dots represent samples from the neuron. The red line depicts the expected output of the neuron. The shape of the expected output is akin to a peaky sigmoid activation, and similar shapes are found in the other neurons of the network as well. This provides food for thought on the design of activation functions for conventional neural networks.
interpreted as a combination of an RBF kernel and sigmoid. This provides food for thought on how non-linearity’s for conventional neural networks can be designed, and the effect of using such a non-linearity can be studied in further work.

6.4 DISCUSSION

In this work, we have proposed a new flexible class of variational distributions. To measure the effectiveness for real world classification, we applied the class to a deep variational information bottleneck model. By placing a quantization-based distribution on the activations, we can compute uncertainty estimates over the outputs. We proposed an evaluation scheme motivated by the need in real-world domains to guarantee a minimal risk. The results presented indicate that SQUAD provides an improvement over plain neural networks and Gaussian information bottleneck models. In comparison to a MCDropout model, which approximates a Bayesian neural network, we get competitive performance. Moreover, qualitatively we find that the flexible distribution is used to its full advantage. The method learns interesting non-linearity’s, is tractable and scale-able, and as the output domain is constrained, no batch normalization techniques are required.

Various directions for future work arise. The improvement of ensemble methods over individual models indicates that there remains room for improvement for capturing the full uncertainty of the output, and thus a fully Bayesian approach to SQUAD which would include weight uncertainty, shows promise. The flexible class allows us to define a wide variety of interesting priors, which provides opportunity to study interesting priors that are hard to define as a continuous density. Likewise, more effective initialization of parameters for the proposed method requires further attention. Orthogonally, the proposed class can be applied to other variational objectives as well, such as the variational auto-encoder. Finally, the discretized nature of the variables allows for the analytical computation of other divergences such as mutual information and the Jensen-Shannon divergence, the effectiveness of which remains to be studied.
6.A EFFECT OF HYPER-PARAMETERS ON COVERAGE:

The optimal configuration of hyper-parameters and bin priors have been determined using 700 evaluations selected using TPE. The space of parameters explored is as follows, presented in the hyperopt API for transparency:

# Shared
C: quniform(2, 10, 1) * 2 + 1,
dropout rate: uniform(0.01, .95),
lr: loguniform(log(0.0001), log(0.01)),
batch_size: qloguniform(log(32), log(512), 1)
# SQUAD & Gaussian
kl_multiplier: loguniform(log(1e-6), log(0.01)),
init_scale: loguniform(log(1e-3), log(20)),
# SQUAD
use_bin_probs: choice(["uni", "gaus"]),
use_bins: choice(["equal_prob_gaus",
                  "linearly_spaced"]),
learn_bin_values: choice([
                          "per_neuron", "per_layer", "fixed"]),

In figure 6.A.1 we visualize the pairwise effect of these hyper-parameters on the coverage. The optimal configuration found in for the main SQUAD model are: batch size: 244, KL multiplier: 0.0027, learn bin values: per layer, $p(z)$: uniform, $\nu$: linearly spread over (-3.5,3.5), lr: 0.0008, C: 15, initialization scale: 3.214.
Figure 6..7: This figure serves to provide intuition on how a variety of distributions come about in our model. We show the set of weights used to predict the probability for the C bins of a randomly selected latent variable $z_{l=1,k=12}$ from the first layer in a converged 2-layer SQUAD model (reshaped to a 28x28 squares for comparison with the data). We then present 5 data-points for which the neuron predicts a stereotypical distribution, as visualized in the last bar-plot.
**Figure 6.A.1**: This figure visualizes the pairwise relationship between hyper-parameters of SQUAD and the effect on coverage. The top-60 configurations are highlighted. Green values are good, red values are bad. We have filtered on the optimal settings for bin values and prior to reduce clutter.
CONCLUSION

In this thesis we revisited the core components of the deep learning toolkit and reevaluated them in the context of medical image analysis. We identified three main areas where deep learning is falling short. In part I we explored the benefit of integrating expert knowledge on structure and constraints in the model. Part II followed with a study on how deep learning can better leverage the vast amounts of unlabeled medical imaging data available. Finally, in part III we explored how to extract better uncertainty estimates from the model to provide diagnostic risk assessment. Let us now consider the research questions posed at the beginning of this work.

**Research Question 1:** Can roto-reflective equivariance improve accuracy and robustness of deep learning models in a large scale medical setting?

Certain medical imaging problems come with different symmetry modes and nuisance factors than found in the real world images typically studied in deep learning for computer vision. We explored this problem in the context of histopathology for the task of detecting metastatic tissue. Using the framework of GCNNs, we designed a novel architecture that captures the rotational symmetries of slide images. This new model outperforms traditional CNN architectures, even under strict equivalence of computational cost and parameter count. Moreover, the new model proved more robust to perturbations. We can conclude that in the right circumstances and especially in the face of limited data, incorporating expert knowledge in the form of model invariance improves the robustness and accuracy of deep learning models.

Looking forward, we consider that the research area of group equivariant neural networks have progressed since Cohen and Welling, 2016. Incorporating these novel methods to enable scale equivariance, continuous rotation equivariance and perhaps color equivariance, can potentially improve perfor-
mance further. However, the effectiveness of integrating expert knowledge as priors in a deep learning model remains a point of contention. As discussed in the bitter lesson, Sutton, 2019, expert knowledge might hamper a deep learning model from discovering efficient solutions to a decision problem. General methods that benefit from scale in data and compute have historically outperformed custom methods. GPT-style language models contain little to no concepts from linguistics, and have proven to be highly effective. Incorporating invariances should be evaluated on a case-by-case basis. And as the size of models and datasets grows, we might find that they provide a diminishing impact on performance.

**Research Question 2:** *How can we motivate the deep learning research community to optimize models for real world medical problems?*

To aid the study of deep learning for medical problems, we proposed PCAM, a dataset derived from Camelyon16. This dataset is structured similarly to typical deep learning benchmarks such as CIFAR-10. We demonstrate improving on the PCAM benchmark translates to improvements on the large-scale Camelyon16 benchmark. Since its release, PCAM has been the center of a Kaggle challenge, with 1149 teams competing to improve performance on the challenge. The accompanying paper has been cited 175 times since, and the dataset has been used as a benchmark for groundbreaking research such as CLIP (Radford et al., 2021).

**Research Question 3:** *Do medical imaging deep learning models benefit from contrastive predictive coding for self-supervised representation learning?*

Next, we studied how contrastive representation learning can benefit medical imaging. We identified that CPC on its own performs well, but suffers from nuisance factors found in medical images. By carefully incorporating a set of augmentations specific to the medical tasks, we found improved representations which translated to better downstream prediction in both histopathology and X-ray data.

**Research Question 4:** *Can we optimize deep learning models effectively without end-to-end backpropagation?*

As the scale of deep learning architecture grows, the cost of building the model training infrastructure explodes. To enable synchronized gradient backpropagation across the full network, computational nodes need to connected with high-bandwidth channels and physically collocated. We explore contrastive representation learning as a tool for training large models without the
need for end-to-end backpropagation. We found that our proposed method, GIM, outperforms its supervised end-to-end counterpart and enables training layers of a network in relative isolation. Future work can study the effectiveness of GIM in a true distributed setting; can we train a neural network efficiently using GIM by distributing the model over multiple computational nodes with little communication between the nodes?

**Research Question 5:** Can latent variable quantization enable more expressive approximate mean-field posterior distributions and yield improved confidence scores in DLVMs?

Finally, we explored the problem of calibrated uncertainty. We considered the hypothesis that variational inference suffers from a too restrictive class of approximate posterior functions. Based on this, we proposed a novel mean-field approximate posterior based on quantizing latent variables called SQUAD. We demonstrate that this posterior can be effectively trained in a DLVM setting, does not need batch norm, and yields uncertainty quality that reduces risk.

Since this work, the use of quantization as a way of modeling density over a continuous domain has taken off. Quantization such as the VQ-VAE (Oord et al., 2017b) has played a crucial role in the image generation tool DALL-E (Ramesh et al., 2021) and quantile regression in reinforcement learning (Dabney et al., 2017b). This continues to be an interesting approach to approximate posterior modeling, and can be studied in a variety of variational inference settings.

The search for a proper Bayesian treatment of deep learning is ongoing. This area of research has fallen out of favor, as various studies to train large deep learning models in a Bayesian manner reached negative conclusions (Wenzel et al., 2020). Only recently, Shen et al., 2024 demonstrate that with an adapted optimizer, achieving well-calibrated uncertainty from a large probabilistic neural networks is possible.

**CLOSING WORDS**

Deep learning brings a tremendously exciting realm of possibilities to the field of medical image analysis and medicine in general. Capturing medical expertise in statistical models enables humanity to improve diagnostic quality. Moreover, with the spread of computation and internet to every corner of the
globe, it enables medical expertise to be easily replicated and made available to all communities. There are still technical roadblocks to overcome to make this vision a reality, but with the pace and breadth at which the field of machine learning is progressing, the future seems bright.
BIBLIOGRAPHY


Dong-Hyun Lee, Saizheng Zhang, Asja Fischer, and Yoshua Bengio (2015). “Dif-


Aaron van den Oord, Yazhe Li, and Oriol Vinyals (July 2018a). “Representation Learning with Contrastive Predictive Coding.” In: arXiv: [1807.03748 [cs.LG]]


Yuesong Shen, Nico Daheim, Bai Cong, Peter Nickl, Gian Maria Marconi, Clement Bazan, Rio Yokota, Iryna Gurevych, Daniel Cremers, Mohammad Emtiyaz...


Matthijs H Valstar, Bernadette S de Bakker, Roel J H M Steenbakkers, Kees H de Jong, Laura A Smit, Thomas J W Klein Nulent, Robert J J van Es, Ingrid Hofland, Bart de Keizer, Bas Jasperse, Alfons J M Balm, Arjen van der Schaaf,


SAMENVATTING - SUMMARY IN DUTCH

Deze thesis heroverweegt de fundamentele componenten van deep learning en evalueren hun toepassing in de context van medische beeldanalyse. Het identificeert drie belangrijkste uitdagingen waar deep learning in dit domein tekortschiet: de integratie van deskundige kennis, het gebruiken van unlabeled data en het inschatten van model onzekerheid. De thesis is gestructureerd in delen die respectievelijk deze uitdagingen aanpakken.

In Deel I introduceert de thesis een nieuw deep learning model dat deskundige kennis integreert door middel van roto-reflective equivariance, om zo de nauwkeurigheid en robuustheid van medische beeldvormingstaken te verbeteren, specifiek bij de detectie van metastatische weefsels in histopathologische preparaten. Het voorgestelde model presteert beter dan traditionele CNN-architecturen en is robuust onder input-pertubaties. Vervolgens wordt onderzocht hoe de deep learning-gemeenschap gemotiveerd kan worden om zich te richten op reële medische problemen. Een nieuwe dataset, PCam wordt geïntroduceerd, afgeleid van de Camelyon16-challenge.

Deel II onderzoekt de voordelen van self-supervised representation learning door middel van Contrastive Predictive Coding (CPC) en introduceert Contrastive Perturbative Predictive Coding (C2PC), dat de prestaties van CPC verbetert door specifieke medische beeldvormingsaugmentaties te integreren.

Deel III van de thesis behandelt de uitdaging van het schatten van model onzekerheid, cruciaal voor medische besluitvorming met hoog risico. Het introduceert een nieuwe variational inference methode die gebruik maakt van multinomiale verdelingen over quantized latente variabelen. De voorgestelde methode toont competitieve prestaties in onzekerheid schatten en risico analyse vergeleken met bestaande methoden.

De thesis concludeert dat door de geïdentificeerde uitdagingen aan te pakken, deep learning beter geschikt kan worden gemaakt voor medische beeldvorm-
ingstaken. Het toont aan dat deskundige kennis effectief kan worden geïnte-greerd in deep learning modellen, dat het benutten van unlabeled data door middel van self-supervised learning de modelprestaties kan verbeteren, en dat model onzekerheid kan worden verbeterd met meer flexibele variational inference methoden.
ACKNOWLEDGEMENTS

Out of all sections of this thesis, this is both the most daunting and most rewarding one to write. There is an enormous group of incredible people whom have graciously given me advice, time, support, friendship, trust, care, faith, patience, assistance, motivation, energy, endorsement, encouragement, attention and understanding. While it might take a village to raise a child, it appears to take a small army to bring a PhD to fruition. My words will certainly fall short of paying proper tribute to you, but know that I am forever grateful.

First and foremost, I want to thank my promotor Max Welling, who has played an essential role in my academic career. From the start of my masters degree in artificial intelligence at the University of Amsterdam, where Max encouraged us to set a high bar in the academic pursuit, to advising me on my MSc thesis and creating the opportunity and encouragement for me to pursue a PhD. Max has been very generous with his time and I would always leave our meetings with new ideas and renewed drive. You have created many opportunities both during and after the PhD for a great many people, including myself, and my gratitude can not be overstated.

I also want to thank co-promotor Milan Petkovic, for his essential advice through the years, for giving me the opportunity to work closely with the wonderful people at Philips Research, and for encouraging me to pursue ideas and collaborations both within and outside the scope of the project.

There are some key people without whom this thesis and my defence would not have happened. I am incredibly grateful to have two of them accept the ceremonial role of paranymphs for my defence, as they both had a huge influence on me during my PhD. Maximilian, thank you for keeping me sane, for our many walks, chats and trips to the spar. Thomas, thank you for all the advice, help, brainstorm, and VR minigolf.

My colleagues at the UvA were always a great source of both inspiration and welcome distraction. In no particular other: Rianne, thank you for your in-
exhaustible positivity, feedback and advice. As with Max, I seem to can’t help myself from following where you lead. Patrick, thank you for many whiteboarding sessions. Arsenii, for advice and contagious excitement. Daniel and Shi, for being great office mates. Jakub, for endless optimism. Tamis, for greening our offices. Zeynep, for your advice. Taco, for being a mentor. Karen, for advice and help. Tim, Victor, Emiel, Jorn, Peter, Christos, Herke, Erik, Priyank, Qi, Elise, Putri, Maurice, Heiko, Wouter, Tom, Eric, David, Teddy, Sara, Gjorgji, Pascal, and many others, thank you all for advice, feedback, frisbee games and more, you made the ride so much better. Marco, for a wonderful collaboration and discussions on all things information. Babak, Sharv, thanks for the board game diversions. Andy, thank you for all the food, hikes, and discussions.

My wonderful colleagues at MSR should not go unmentioned. Thank you all for contributing to the incredible warm, friendly and ambitious culture we now have in Amsterdam and beyond. You helped me find great joy in research again after the bulk of my PhD was behind me, and taught me all so much. Thank you to Phillip, Paris, Rich, and Johannes for a great collaboration. Ryota, thank you for your invaluable mentorship and advice.

Collaborating with Jim and Jasper was a joy, thank you for joining forces. Adam, for teaching me more about the brain. Govert, for an exciting deep dive into compositionality. Sindy and Peter, thank you for an incredible collaboration.

All of this would not have been possible without the generous support from Philips Research. Special thanks to Bart Bakker, who always prioritized enabling research over anything else. Binyam, Ulf, Stojan, Dimitrios, Harold, Nicola, Karan, Vlado, Shakith, Karl, and Frits, for many joyful whiteboarding sessions, discussions, introductions and demonstrations of real messy wet lab science.

I also have to thank Frits and Bart for introducing me to Ruud van Sloun, whom’s contagious optimism and out of the box thinking came at the most welcome time. Our collaborations were a great source of joy and and ever the more reason to jump on the train to Eindhoven. Working together with the incredible Iris, Hans and Liz was a wonderful opportunity, and I can only hope our paths will cross again.

My internship at Google Brain Berlin was a formative period. Sebastian, thank you for your mentorship and drive, as well as Jakob. My fellow interns Kuba, Linh, Kevin and Jasper, Rodolphe, Florian and Stephan, for being amaz-
ing collaborators. It was great to work alongside Thomas, Alexey, Aravindh, Aidan, Sebastian and many others. Durk, thank you for humoring one of my more speculative research ideas. Continuing as a student researcher in Amsterdam was a dream come true and I have to thank Tim especially for his guidance and support. All my colleagues at Google Brain Amsterdam were a great source of inspiration and aspiration, including and certainly not limited to Lasse, Nal, Capser, Jonathan, Avital, Mostafa; thank you. I also want to thank Carlos, Ruth and Axel, for giving me a home away from home.

I want to thank Felice Arends, Odette Compen-Hendrix, Erik Hitipeuw, Kris de Jong and Boy Menist for helping me navigate both Philips and the UvA, and for enabling me in pursuing internships. I also want to give special thanks to Dennis Koelma and Bas Terwijn for their advice and for enabling bits to flow where they have to go. On SURFStara’s side, I have to thank Vali for collaborating and enabling some of my more excessive number crunching stints.

I am also grateful to the organizers and attendees of the wonderful Deep Bayes summer school by Dimitri Vetrov’s group. Likewise, the summer school by Microsoft Research was an incredible opportunity to learn and network. Equally important was the DLMedia consortium who let me hitch a ride for the many interesting talks, the summer school and more.

The following organizations have been vital for my PhD. Philips Research, for the financial support for my PhD. SURFsara, for allowing me to test-drive the LISA GPU supercomputer. NVIDIA, for their generous GPU grant.

Of the people close to me, I want to thank Hüs, for supporting me during this wonderful and chaotic period of our lives. My grandpa, George, for teaching me to take things serious and inspiring me to take an academic career. My parents, for their unwavering support, advice, feedback on drafts and love. My sisters, for being there for me. My uncles Henk and Frans, for their advice inspiring to be ambitious.

I want to thank the wonderful folks of Het Wilde Wat, for creating a sense of community, a creative outlet and a great source of joy, and Ben, for introducing me to this wonderful group and being a great friend.

Luisa, thank you for being a voice of reason and inspiring me, as well as the patient feedback on drafts. Joost, for many discussions, trips, advice, tireless feedback and encouragement. Marcel, for reminding me to not take things too serious. Bernice, for listening and riding together. Bertina, for great dinners and walks. Joost for listening when it is needed and sharing the incredible
foresight to pursue a degree in AI. Thijs, for inspiring me to take initiative. David, for reminding us to celebrate all our wins. Otto, for making me laugh. Suin, for incredible korean food. Richard and Rien, for being great housemates. Maartje, for being a close friend and partner paranymph, as well as Maurits, Ties and Jörg, for our many dinners and great company. Wendy, for being a great travel partner. Damla, Ercin, Tunca, Amu, for providing a place of calm amongst all the chaos. Thiviyan, for listening and sharing. Marloes, for welcoming me to Copenhagen. Jan Willem, for sharing notes and advice on career. Miriam, for showing me what strength and grace in the face of adversity is, and being there regardless. Thomas, for good conversations and musical discovery. Fritjof and Roger, for being there from the start of the AI adventure in Amsterdam and staying in touch. Janna, for going above and beyond and helping me see things in a different light. Agner, for your friendship and helping me take my mind off work. Benjy, for your encouragements and introducing me to both new and old music. Gregor, for pushing me to ride faster. Ana and Thomas, for taking karaoke to the next level. Johannes, for showing me the beauty of Austria and for a wonderful collaboration. Jose, for sharing the excitement for the metaverse where we’ll surely spend much time in the future. Elise, for keeping it metal. Thijs, Giulia and Chin-Wei, for rocking together.

Finally, I want to thank the committee members, Prof. Dr. Zeynep Akata, Prof. Dr. Paul Groth, Prof Dr. Evangelous Kanoulas, Dr. Stephan Mandt, and Dr. Erik Bekkers for generously taking the time to read this thesis and sharing their expertise. It is an honour to be able to discuss my work with all of you.

Extending a PhD project beyond the typically allotted 4 years means that the list of people who come to impact the thesis and myself grows even larger. If you are reading this, you most likely are one of them. If my memory has lapsed in the haste of writing this and I have failed to acknowledge you: I beg your forgiveness, and invite you to reach out so I can promptly buy you a drink to make up and reminisce on the good old days.