ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Max-Planck-Institute für
Intelligente Systeme

Master Thesis

Learning to Diagnose Diabetes from Magnetic Resonance Tomography

Spring Term 2019

Supervised by:

Prof. Dr. Schölkopf

Dr. Stefan Bauer

Patrick Schwab

Author:

Benedikt Dietz

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.

Contents

Acknowledgments

This master thesis has been conducted at the Swiss Federal Institute of Technology (ETH) Zürich, as a collaboration with the ETH Zürich Institute for Machine Learning and Max Planck Institute for Intelligent Systems in Tübingen. Furthermore, we owe gratitude to the University Clinic of Tübingen for the opportunity to work on this dataset and in particular to PD Dr. Robert Wagner and PD Dr. sc. hum. Jürgen Machann, who accompanied the project with medical expertise. In addition, I would like to extend my sincerest gratitude to Prof. Dr. Schölkopf and Max Planck Institute for Intelligent Systems for giving me the opportunity to participate in this research as well as to my supervisors Dr. Stefan Bauer and Patrick Schwab for their guidance and support throughout the project.

Abstract

Medical image analysis utilising machine learning techniques has shown promising results in recent years [1, 2, 3]. Despite its considerable global prevalence, image analysis literature concerning diabetes and diabetes-related features is sparse and has predominately been focused on Diabetic Retinopathy [4]. As part of a prediabetes study conducted by the University Clinic of Tübingen, a dataset, consisting of approximately 2.5k full-body Magnetic Resonance Tomography (MRT) scans in addition to medical data about the respective patients has been collected.

We build a Convolutional Neural Network (CNN) model and train it on the MRT dataset to determine whether features indicative of diabetes are present. To this end, we utilise the Densely Connected Convolutional Network (DenseNet) architecture, proposed by Huang et al. [5], to predict three different binary diabetes diagnoses in addition to gender, age, body mass index, Insulin sensitivity, and Glycated Hemoglobin (HbA1c). We show that diabetes prediction based on MRT scans achieves an area under the receiver operating characteristic curve (AUROC) of approximately 0.8*. We give an outline of the network architecture in addition to a summary of several performance metrics for the regressions and classifications. Furthermore, we used the trained network to generate gradient heat maps, indicating which areas of the body were considered relevant for the respective prediction. These heat maps are independently classified by medical experts for validation. Additionally, we provide an analysis of the high-dimensional representation of the network before the final output nodes, referred to as the embedding layer. We use an unsupervised unsupervised k-means clustering in addition to a t-Distributed Stochastic Neighbor Embedding (t-SNE) of the embedding layer to analyse and visualise the network's learnt representation of the samples.*

Predictive performance concerning the classification of different diabetes definitions varies according to their respective prevalence in our dataset. Diabetes is least represented with $\sim$ 2% *positives and achieves a maximum area under the ROC curve of* 0.87 *while prediabetes, with* $\sim$ 45% *positives, achieves a score of* $\sim$ 0.7*. However, F1-score and precision improve for more prevalent diabetes labels. Gender classification achieves an AUROC of* $\geqslant$ 0.99*. We find that the predictive performance for continuous features is more sensitive to network hy-*

perparameters. Generally, body mass index tends to perform best whereas Insulin sensitivity and HbA1c prove to be more challenging.

The results presented in this thesis provide a proof of concept for the feasibility of MRT-based diabetes classification. Future work is need to validate our results, improve the model's performance and assess the clinical utility of our approach..

Acronyms and Abbreviations

D_α	Diabetes
D_β	Pre-Diabetes
D_γ	Diabetes IFG+IGT
GPU	Graphics Processing Unit
CUDA	Compute Unified Device Architecture
HbA1c	Glycated Hemoglobin
BMI	Body Mass Index
IS	Insulin Sensitivity
MRT	Magnetic Resonance Tomography
CT	Computed Tomography
D_{KL}	Kullback-Leibler Divergence
ELU	Exponential Linear Units
t-SNE	t-Distributed Stochastic Neighbour Embedding
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DenseNet	Densely Connected Convolutional Network
ResNet	Residual Network
DICOM	Digital Imaging and Communications in Medicine
ETH	Swiss Federal Institute of Technology

List of Figures

List of Tables

Chapter 1

Introduction

Due to the considerable complexity of biological systems, such as the human body, and despite significant advances in modern medicine, the underlying processes behind many diseases remain elusive. Modern tools have enabled the collection of large quantities of medical data, ranging from simple measurements, such as temperature or blood pressure, to high-dimensional information, such as images, genome- or time-series data. However, the analysis and classification of collected, high-dimensional data is not trivial. Transforming the vast amounts of medical data into valuable insights therefore remains a key challenge in health care.

Considerable improvements in computational hardware performance in recent years have enabled the field of *Machine Learning* (ML) to advance at a fast pace and produce impressive results. Various models and approaches have been proposed for different applications. Prominent examples include natural language processing [6, 7], image recognition [8] or complex games [9].

The ability of ML models to extract high-level feature representations from unstructured data holds a large potential for the application of modern ML techniques in health care. Concerning medical diagnosis and prognosis, algorithmic approaches have proven to be capable of exceeding human expert performance at several tasks, such as, among others, the classification of Pneumonia from X-Ray scans [10] or lung cancer detection from CT scans [11]. Other successful applications include genome research and the discovery and development of drugs [12], as well as clinical decision support or monitoring systems [13]. Image-based diagnostics models represent a significant amount of related research and have been applied to, *e.g.* detection of cancerous tissue [11] or diagnosis of different diseases [14, 1]. Routine screening procedures, as are common for *e.g.* several kinds of cancer, provide data sources that are well-suited for the application of ML image recognition methods. Taking the entire structure and physiology of patients into account is interesting for several diseases, though respective datasets are often not publicly available.

Diabetes is a metabolic disease with a considerable and increasing global prevalence [15, 16]. Defects in Insulin-secretion and -action result in chronic hyperglycemia, which is characteristic for diabetes and is associated with damage to different organs [15]. The development of diabetes is correlated with several pathogenic processes and its symptoms affect different organs and body functions [15]. Despite significant research efforts, there still remain unanswered questions regarding the correlation between diabetes and various structural features.

As part of a prediabetes study, the University Clinic Tübingen has collected a dataset of full-body *Magnetic Resonance Tomography* (MRT) scans of study participants along with their respective diabetes-related information. In this thesis, we use this dataset to determine the predictive performance of machine-learning models in diagnosing diabetes from full-body MRT scans.

To this end, we have adopted and trained a state-of-the-art image recognition architecture to diagnose diabetes using three different diagnostic clinical tests as ground truth.. Additionally, the model predicted several diabetes-related features, consisting of age, body mass index, Insulin Sensitivity and *Glycated Hemoglobin* (HbA1c). To analyse the model's optimised, high-dimensional representation of the MRT scans, we used dimensionality reduction to visualise the high-level representations learnt by our model. We have also used k-means clustering and analysed the generated cluster distributions. Lastly, we used the model's gradients with respect to the different target labels to generate heat maps, that highlight areas of importance for the given prediction and target label. For validation and analysis of the generated heat maps, we recruited medical experts to manually classify whether certain anatomical regions are highlighted in the heat maps.

The presented work begins with an outline of related literature, regarding ML image recognition models and their medical applications, in addition to a brief overview of dimensionality reduction techniques, as well as the k-means algorithm. Chapter 3 provides an account of the used methods and implementation details before we present our results in Chapter 4. Finally, we conclude with a discussion of the project, its limitations and proposed future work.

Chapter 2

Related Work

Machine Learning and *Deep Learning* in particular have attracted a lot of attention in recent years, both from inside and outside academia. The current popularity of *Artificial Intelligence* (AI) is mostly due to impressive demonstrations of its potential in various real-world scenarios. Deep Learning models have been able to match and even surpass human benchmark performances in several applications, such as, among others, image recognition [8], natural language processing [6, 7] or complex games [9].

Many of these recent advances have utilised *Artificial Neural Networks* (ANN). The term refers to a specific type of machine learning model consisting of a graph structure with various layers, each performing some non-linear transformation on its input and passing it forward, thus mapping arbitrary input values to their respective outputs [17]. Each layer is associated with a certain number of randomly initialised parameters $\boldsymbol{\theta}$, called *weights*, defining the (non-linear) transformation. Through an iterative process of alternating forward- and back-propagation [17], the weights are optimised with regard to a defined loss, *i.e. trained*. Individual layers gradually learn to build meaningful representations of their input with respect to the given objective. This framework is roughly inspired by the human brain [18]. Deep Learning generally refers to networks with more than one 'hidden' layer between the input- and output nodes. Through the combination of simple non-linear transformations, *i.e.* multiple layers, complex and abstract representations can be learnt by representing them as nested hierarchies of concepts [17].

Researchers have devised numerous variations of neural networks for different applications. *Convolutional neural networks* (CNNs) [19], refer to a sub-category of neural networks that perform convolutions instead of matrix multiplications inside the hidden layers and most state-of-the-art image recognition algorithms build upon this approach [17, 1]. The literature concerning CNN image analysis and its medical applications can be categorised in a number of main topics:

classification, object- or lesion classification, detection, segmentation and registration.

The main focus of this work is to assess data from full-body imaging in order to determine whether diabetes or features indicative of diabetes are present. Hence on image classification. Further literature on the various fields of research is presented in [1, 20]. Related work considering Deep Learning image analysis and classification as well as its medical applications are outlined in the following chapter.

2.1 Deep Convolutional Neural Networks

CNNs are used for input data with a grid-like topology [17], such as images, to capture spatial dependencies. The essential component of CNNs is the convolution operation which is applied to the input instead of a matrix multiplication of the entire input array. The convolution *kernel* represents the filter applied to the image and, thus, the operation's weights [17]. The convolution S of a two-dimensional input image I, using a kernel K is defined as

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i-m, j-n)K(m,n). \tag{2.1}$$

In classical feed-forward MLP models, each input unit, *i.e.* image pixel, interacts with each unit of the consecutive layer. Hence these layers are referred to as *Densely Connected Layer*. In convolutional layers, the kernel size can be chosen to be much smaller than the arbitrarily large input, resulting in *sparse connectivity* [17]. Furthermore, convolutions use the same set of parameters on every input unit, referred to as *parameter sharing*. Therefore, the number of parameters does not depend on the dimension of the input, as for MLPs, but on the kernel size. Lastly, *equivariance* to translations means that the network does not need to learn individual feature detectors for objects appearing at different positions in the input image. The properties of sparse connectivity, parameter sharing and translational equivariance result in drastic improvements to the model's memory requirements and efficiency [17].

Convolutional networks have been researched since the late seventies [1, 21] and were applied on medical images as early as 1995 [22] for the detection of lung nodules. Deep CNNs were introduced roughly 20 years ago in the work of LeCun [19] to successfully classify hand-written digits. Figure 2.1 summarises the architecture of LeNet-5 by LeCun et al. [19], which illustrates the general concept of CNNs. After substantial improvements in hardware and especially *Graphics Processing Unit* (GPU) supported training had been achieved, the research on

Deep Learning image recognition evolved quickly with important contributions to the CNN framework such as *AlexNet* [23] and *VGG networks* [24].

AlexNet and *VGG* arguably represent today's standard implementation of CNNs using max-pooling layers [19] for effective dimensionality reduction and rectified linear units (ReLU) [25] as activation functions to address the issue of vanishing gradients with increasing network depth [26]. Additionally, *Dropout* [23], which randomly drops a predefined fraction of nodes during each iteration to avoid over-fitting the training set, is a commonly used component in CNNs. Typically, the input, *i.e.* state of the model is increasingly down-sampled and the number of representation maps is increased with every convolution layer. The outputs of the convolutional layers in the CNN are low-dimensional feature maps of the respective input. Finally, in most CNN models, the feature maps are flattened to a single one-dimensional array and passed to a set of densely connected layers, to map the convolutional representation to the desired output.

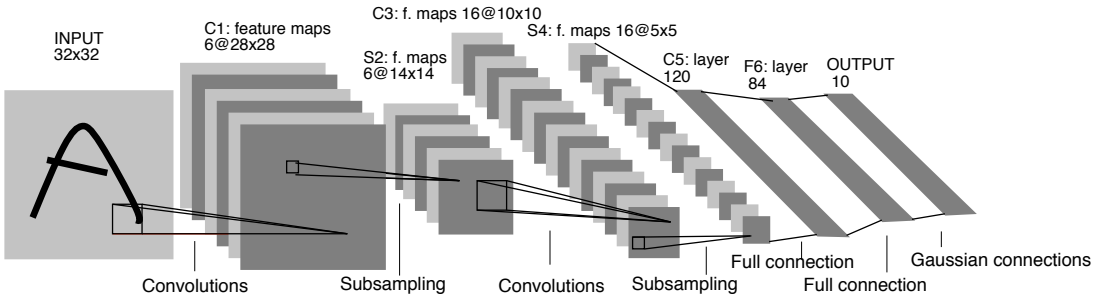

Figure 2.1: Illustration of LeNet-5 CNN. Image taken from [19]. The convolutions generate multiple activation map from their input. These are down-sampled using pooling operations and after the convolution layers, the flattened representations are passed to fully connected layers to map them to the final output.

The resulting sequential graph is iteratively computed using training images and differentiated with regard to a predefined loss function to optimise its weights using gradient descent. This framework has successfully proven its potential by achieving state of the art benchmark results on various image recognition data sets [27, 28, 29] and has become the architecture of choice for image recognition taks [30]. Further model components used in the presented work were outlined in the following chapters.

2.1.1 Batch Normalisation

Gradient-based optimisation of deep networks is not trivial. The loss function is non-convex and exhibits flat regions, sharp minima and kinks, which lead to vanishing- or exploding gradients and makes training very sensible to hyperparameter choices and initialisation. Particularly, low learning rates are essential due to sudden changes in the loss functions [31]. Batch normalisation [31] addresses this issue by significantly smoothing the loss function as well as the gradients [32].

Batch normalisation introduces additional layers in the network, which control, *i.e.* normalise the first two moments of layer input distributions. This normalisation leads to significant smoothing of the optimisation landscape, resulting in lower loss gradient magnitudes [32]. Regarding the gradient-based optimisation, the reduced loss gradients and overall smoother optimisation problem enable the use of increased learning rates and make it less sensible to hyperparameters, initialisation and overfitting. The approach has proven to considerably improve training of deep networks and has been widely adopted in Deep Learning and CNN implementations [33, 5, 34].

2.1.2 Bottleneck Layer

In traditional CNN implementations all output feature maps of a given layer act as input to its subsequent layer, thus resulting in an exponential increase of the number of parameters with increasing depth of the network. This considerably complicates optimisation, increases memory requirements and training time and requires some type of regularisation to avoid over-fitting. Hence both depth and width of the network are limited by this computational bottleneck [35]. The works of [36, 33, 37] have shown that dimension reduction using $1*1$ convolutions prior to convolutional layers with larger filter sizes can remove redundancies in the feature maps and therefore considerably improve computational efficiency. This approach has been used in various state-of-the-art image recognition benchmarks [34, 35, 5].

2.1.3 Skip Connections

Another issue that emerges with deeper networks is that the back-propagated gradients converge to zero as the number of layers and feature maps increases. This *vanishing gradient problem* is reduced with the use of ReLU [25] activation functions and batch normalisation layers. Skip connections provide an additional effective and intuitive approach by introducing short paths, *i.e.* skip connection between layers, thereby improving imformation flow through the network.

Highway Networks [38] introduced an approach inspired by *Long Short Term Memory* [39] recurrent neural networks, where gate units are trained to control information flow. The gates allow information to skip several layers without attenuation, enabling training of very deep networks. *Residual Networks* (ResNet) [33] have also adopted skip connections using explicit identity mappings to skip convolution pairs as illustrated in Figure 2.2. Highway networks and ResNets were both among the first to successfully implement networks with over one hundred and much more layers [5].

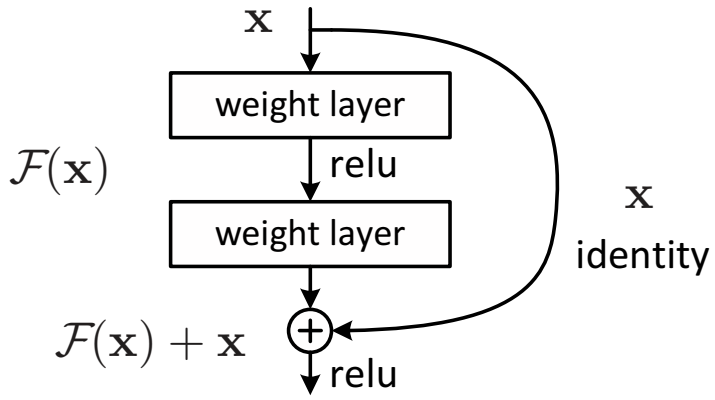

Figure 2.2: Illustration of ResNet Building Block. Image taken from [33]. The schematic illustrates the basic building block of the ResNet architecture. An identity mapping is used to let the input x skip two consecutive convolutional layers with an intermediate ReLu activation. These two convolutional layers apply the mapping $\mathcal{F}(x)$ to the input x. The identity mapping x is added to the the output of the two preceding weight layers, $\mathcal{F}(x)$, before being passed forward. The skip connection approach improves information flow and enables much deeper networks by reducing the number of layers, the gradients have to travel through.

Deep networks with stochastic depth [40] have adopted the ResNet architecture and improved its performance by randomly dropping a considerable fraction of layers during training, thus effectively having less layers during training but a much deeper network for inference. This has proven successful and shown that there is a substantial amount of redundancy in very deep networks [5].

2.1.4 Dense Convolutional Networks

Dense Convolutional Networks (DenseNets) have been introduced by Huang et al. [5] who adopted the skip connection approach by maximising the connectivity between layers of matching feature map dimensions. Each layer receives the outputs of all preceding layers as input and passes its output to all subsequent layers, thus preserving the feed-forward nature of CNNs [5]. Contrary to ResNets, DenseNet feature maps are combined through concatenation instead of summing them up, thus explicitly preserving earlier representations. Hence each DenseNet layer reads the accumulated state of all preceding layers, adds its contribution and passes it forward.

This model enables the network to exploit feature reuse and eliminates the need to relearn redundant feature maps, thus significantly improving memory efficiency [5]. Due to the dense connection of layers, each layer has access to the loss gradients as well as the network input, which results in an implicit deep supervision [41], as well as efficient training, even for very deep networks. The parameter efficiency of DenseNets has the additional benefit of a regularising effect, which helps to avoid over-fitting, particularly on smaller datasets [5].

Pooling, *i.e.* down-sampling, is an essential part of convolutional networks. However, concatenation of feature maps is only viable for corresponding dimensions. To this end, *dense blocks* are introduced. Dense blocks contain a number of densely connected layers, each computing a consecutive composite function. Each layer produces k_{Growth} feature maps, where k_{Growth} represents the network's *Growth rate* [5]. Figure 2.3 shows an illustration of a dense block with five layers and growth rate $k_{\text{Growth}} = 4$.

Down-sampling is performed using *Transition Layers* between individual dense blocks. Transition layers perform batch normalisation, a $1 * 1$ convolution and finally a max-pooling operation. Since each layers generates k_{Growth} feature maps, the layers generally have more inputs than outputs. The $1*1$ convolution has been found to act as a bottleneck layer, reducing the number of input feature maps, thus improving computational efficiency [36, 33]. If the output of a dense block contains m feature maps, a fixed parameter ϕ, ($\phi \in [0, 1]$) can be introduced for futher compression of the model to let the transition layer produce $\phi \cdot m$ feature maps. Finally, a $2 * 2$ average pooling layer reduces the feature map dimensions before passing them to the subsequent dense block.

The remaining model is analogous to traditional CNNs, passing the final output representation of all convolutional layers to fully connected layers, mapping the flattened feature representations to the desired output nodes. The connectivity pattern of DenseNets has proven to result in very compact models, able to

achieve state-of-the-art performance on image recognition benchmarks [5]. Furthermore, Huang et al. [5] found it facilitates optimisation of very deep networks and mitigates over-fitting on smaller datasets, resulting in a well suited feature extractor for image recognition tasks.

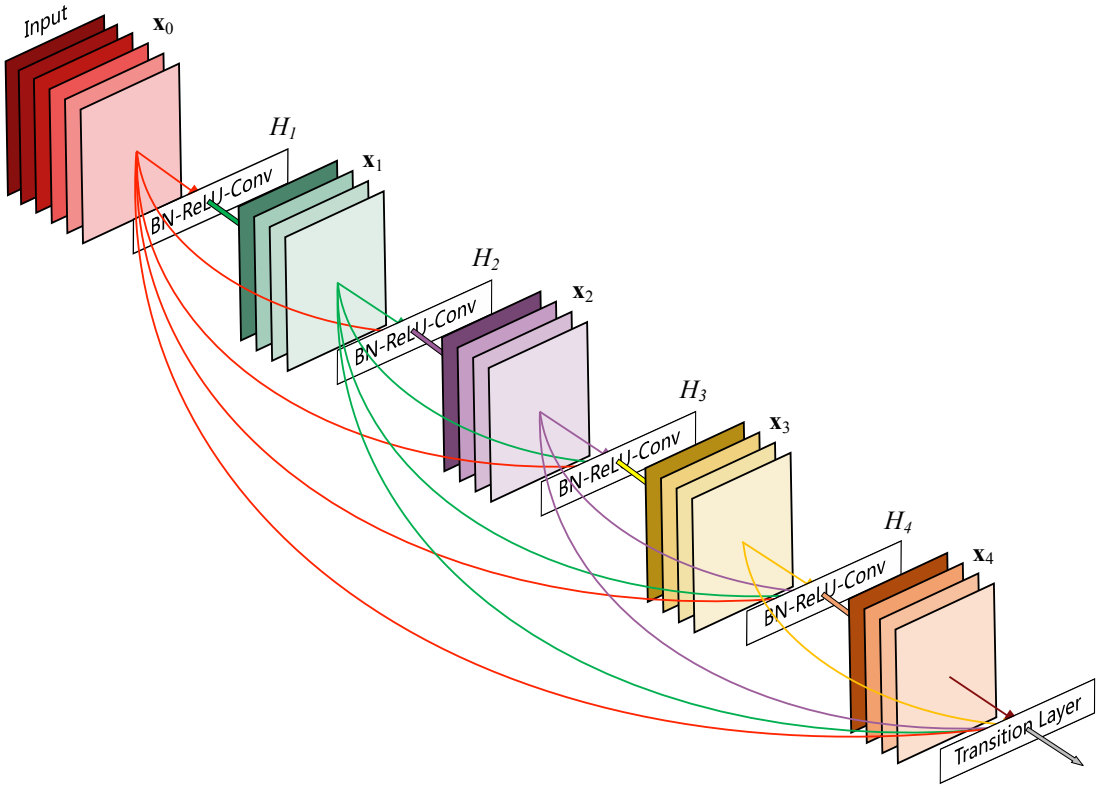

Figure 2.3: Dense Block within *DenseNet* Model. Image taken from [33]. The DenseNet model consists of multiple sequential dense blocks. The illustration shows a dense block with five convolutional layers and $k_{Growth} = 4$. In between two consecutive layers, the transfer function H is applied, consisting of batch normalisation, ReLu as well a a convolutional layer used for compression. The growth factor k_{Growth}, governs the number of activation maps produced by each of the five convolutional layers. As indicated by the connection lines, the output of each convolutional layer is used as input for all subsequent layers, where feature maps of different layers are concatenated to explicitly preserve knowledge [5]. The final output of a dense block is passed to a transition layer which is responsible for dimensionality reduction, *i.e.* down-sampling.

2.1.5 Visualisation of Convolutional Networks

Image classification systems based on deep CNNs have been proven successful in numerous tasks, though they generally act as a black box, *i.e.* do not provide information about the reasoning behind the yielded network prediction. Interpretation and understanding of the classification system, however, is highly valuable for a wide range of applications as it can provide a means of verification as well as additional information to human experts [42]. Hence, a significant amount of research has been conducted to address the black box issue and achieve certain amounts of interpretability [43, 44, 45, 46].The literature concerning the visualization of CNN inference can roughly be split into two categories: Perturbation-based methods and backpropagation-based methods [47].

The basic idea behind perturbation-based methods is to compute the importance of a feature by removing, or occluding the considered feature and comparing the resulting network output to the respective output of the original full set of features. Considering image inputs, the features are represented by pixels or groups of pixels. The original image, *i.e.* set of features $\bar{x}$, is fed through the network to produce a benchmark output. Afterwards, a square of pixels is occluded, such that the respective part of the image cannot be seen by the network. The outputs of the altered image x are computed for every possible position of the occlusion square and difference to the benchmark output represents a measure of importance from which a heat map is generated [47]. The size of the occlusion square has a considerable effect on the generated map. Perturbation visualisations for different occlusion sizes are depicted in the bottom row of Figure 2.4. The approach has the advantage of straight-forward implementation and often produces satisfying results. However, computational requirements increase significantly with larger feature maps and the occlusion harshly alters the analysed images.

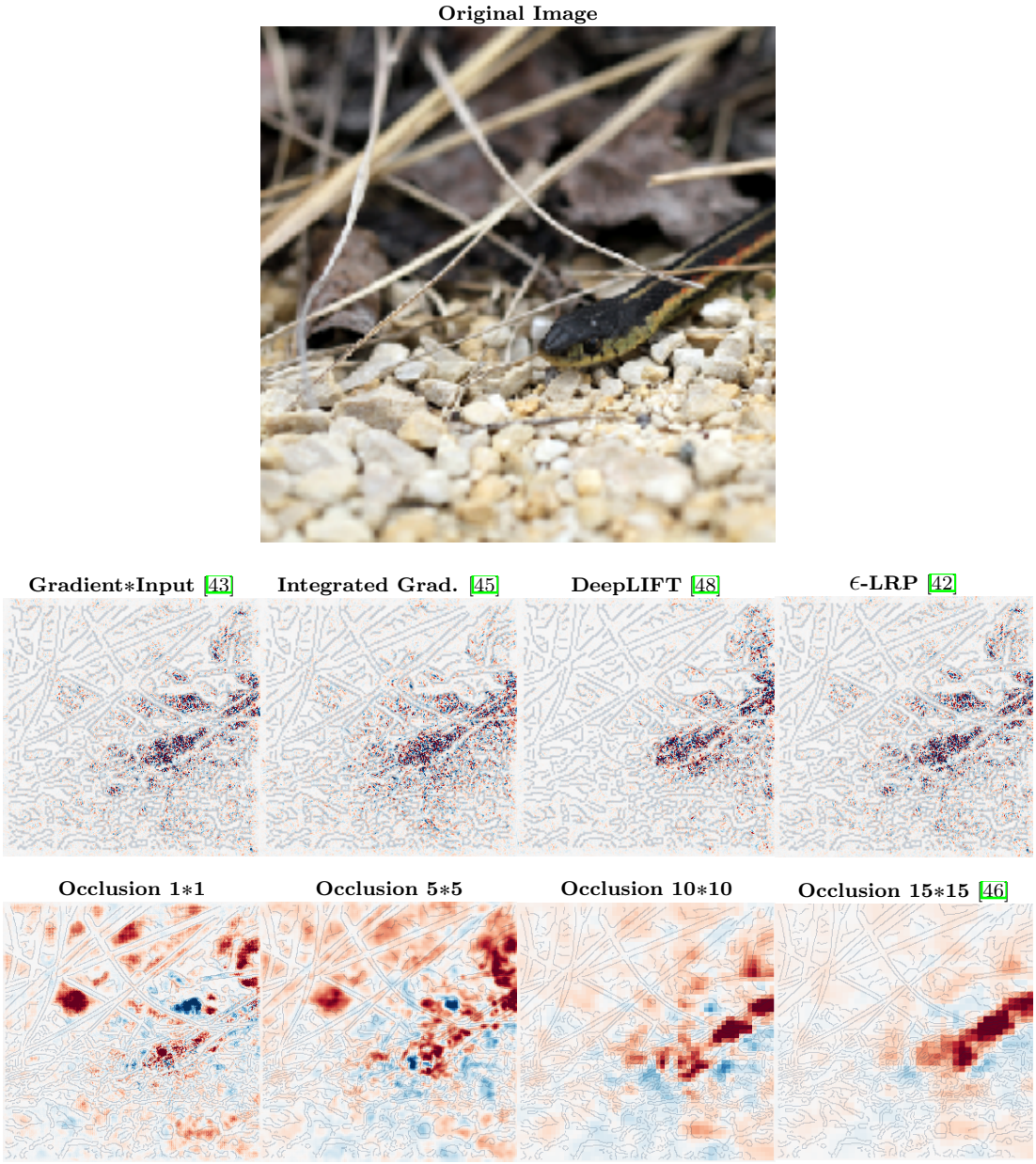

Figure 2.4: Visualisation Examples. Images taken from [47]. The single image in the top row shows the original image used for the comparison. All depicted approaches used a network trained on the classification of images (*e.g.* a certain species of snake for the provided example). Gradient-based algorithms are demonstrated in the middle row. The bottom row shows an occlusion-based approach, utilising different window-sizes. The choice of size for the occlusion has a significant effect on the produced activation maps.

Similarly to perturbation approaches, backpropagation-based methods are used to generate activation maps of the input to visualise the network's attention. In principle, backpropagation-based methods compute the gradient of the output node with respect to the input features. The magnitude of the gradient of a specific feature, *i.e.* pixel is an indicator for its importance concerning the network's output. All feature attributions are computed in a single forward- and backward-pass through the network, resulting in improved computational efficiency [47]. Attribution maps using gradients has first been introduced by Simonyan et al. [30], who constructed *saliency maps* using the output node's absolute partial derivative with respect to all input features, *i.e.* the input image pixels.

Gradients $*$ *Input* [43] has been introduced by Shrikumar et al. [43] to improve the quality of attribution maps. Gradients from the network backward pass are multiplied with the input itself to generate the attribution maps. The method is straightforward in implementation and provides a computationally efficient and intuitive approach to CNN visualisation. Sundararajan et al. [45] later extended the Gradients $*$ Input approach to *Integrated Gradients*. Similar to various other methods, Integrated Gradients introduces a reference input $\bar{x}$. The gradients are computed for inputs which linearly vary from the reference $\bar{x}$ to the considered sample x. The attribution maps represent the average of the acquired gradients over the path from $\bar{x}$ to x. Respective examples are depicted in Figure 2.4.

Bach et al. [42] proposed *Layer-wise Relevance Propagation* (LRP), which utilises a backward pass through the network to compute a *relevance* score with regard to the output for each layer and unit. The relevance is computed recursively, starting at the final layer. Applying the recursive ϵ-rule to map a given layer's relevance distribution to the subsequent layer results in the ϵ-LRP algorithm, demonstrated in Figure 2.4. Similarly, *DeepLift* [48] uses a recursive approach to compute layer-wise attributions for each node, representing its relative effect for a given input. These attributions are compared to benchmark values acquired from a network pass of some reference input, which is often chosen to be zero. Ancona et al. [47] recently showed, that both ϵ-LRP as well as DeepLift can be reformulated as backpropagation methods using modified gradient functions.

2.2 Clustering and Visualisation

High-dimensional data and representations are essential and inevitable in deep neural networks. Furthermore high-dimensional data is produced in numerous applications, ranging from images to time-series data and many others. However, two- or three-dimensional representations provide an intuitive interpretability which is lost in higher dimensions and dimensionality reduction in general is a core problem in machine learning [49]. In addition to dimensionality reduction, clustering algorithms provide a tool to group data of arbitrary dimensionality into meaningful groups. We provide a brief outline of dimensionality reduction techniques as well as an introduction to the *k-means* algorithm, which we utilised to visualise and cluster our dataset, respectively.

2.2.1 Visualisation

Visualisation of high-dimensional data is a non-trivial task since it needs to be mapped from a high-dimensional space to two- or three dimensions, whilst preserving as much of its internal structure as possible. A way to accomplish this is to utilise dimensionality reduction methods, which are designed to map the N-dimensional space $\mathcal{X} \in \mathbb{R}^N$ to the n-dimensional embedding space $\mathcal{Y} \in \mathbb{R}^n$, where $n = 2$ to produce two-dimensional representations that can be visualised in images. The idea is to preserve the pair-wise distances, *i.e.* points that are dissimilar, *i.e.* far apart in the original space $\mathcal{X}$, should stay far away from each other in the embedding space $\mathcal{Y}$ and vice versa.

Various dimensionality reduction techniques have been proposed in literature [50, 51, 49]. These include linear methods, such as *Principal Component Analysis* (PCA) [52] as well as non-linear approaches like *Stochastic Neighbour Embeddings* (SNE) [53] amongst others [54, 55, 53]. High-dimensional data tends to lie on or near low-dimensional non-linear manifolds [51]. Hence, linear models often produce unsatisfactory results when applied to real , high-dimensional datasets [49]. Meanwhile, non-linear mappings often struggle to capture both, the local as well as global structure of the data [51, 49]. A particularly successful and popular method in recent years has been proposed by Maaten and Hinton [49] to address this issue: *T-Distributed Stochastic Neighbour Embedding* (t-SNE) is an extension to the *Stochastic Neighbour Embedding* (SNE) with better optimisation properties and improved results. Subsequent contributions, such as *LargeVis* [51] or *TriMap* [56], have built on the SNE and t-SNE frameworks, mostly to improve computational efficiency and hyperparameter-sensitivity.

Stochastic Neighbour Embedding (SNE)

Euclidean distances are computed for all points x_i in the high-dimensional space $\mathcal{X}$ and transformed to conditional probabilities $p_{i|j}$, representing a measure of similarity between the samples x_i and x_j [49]. For nearby points in the high-dimensional space, $p_{i|j}$ takes higher values and converges to zero for far apart samples and $p_{i|i}$ is zero by definition. $p_{i|j}$ is given by

$$p_{i|j} = \frac{\exp\left(-\frac{1}{2\sigma_i^2}||x_i - x_j||^2\right)}{\sum_{k\neq i}\exp\left(-\frac{1}{2\sigma_i^2}||x_i - x_k||^2\right)}, \tag{2.2}$$

where σ_i is a parameter that is determined through binary search and represents the variance of a Gaussian distribution around point x_i. The desired representation of $\boldsymbol{x}$ in the low-dimensional space $\mathcal{Y}$ is denoted by $\boldsymbol{y}$ and a similar conditional probability distribution $q_{i|j}$ can be computed as

$$q_{i|j} = \frac{\exp\left(-||y_i - y_j||^2\right)}{\sum_{k\neq i}\exp\left(-||y_i - y_k||^2\right)} \tag{2.3}$$

with $q_{i|i} = 0$. Hence the objective is to find an embedding $\boldsymbol{y}$ with conditional distribution $q_{i|j}$ that models the high-dimensional similarities $p_{i|j}$ as accurately as possible. An appropriate cost function for this optimisation is presented by the previously mentioned KL-divergence $D_{KL}(p||q)$. The resulting cost function is given by

$$C = \sum_i D_{KL}(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}. \tag{2.4}$$

.

P_i represent the conditional probability distribution of all points $x_{i\neq j}$ given high-dimensional point x_i and likewise Q_i for data points in the embedding space. Optimising this cost function with regard to the embedding vectors y_i yields the low-dimensional representation of the original data. The optimisation is performed using gradient descent and simulated annealing, however it is highly sensible to hyper-parameter choices. Furthermore, points tend to accumulate and converge to high-density clusters, referred to as the *'crowding problem'* [49].

t-SNE adds two major extensions to the framework to address both of these issues, by introducing a symmetrical version of the cost function with simplified gradients as well as replacing the Gaussian as measure of similarity in the embedding space with a Student-t distribution.

2.2.2 k-means Clustering

Unsupervised clustering algorithms try to partition unlabelled data into meaningful groups, *i.e.* clusters of samples. *k-means* [57] represents the standard method for this task. The algorithm introduces a pre-specified number (k) of centroids μ_k and assigns each sample x_n to one of the centroids. The points assigned to a given centroid are referred to as a *cluster*. Both, the position of the centroids as well as the assignment of samples, are iteratively minimised with respect to a *distortion measure* [58], *i.e.* loss function, J given by

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||x_n - \mu_k||^2, \tag{2.5}$$

where r_{nk} is a binary variable indicating wether sample x_n is assigned to centroid μ_k. Optimisation is done iteratively in two consecutive steps for the centroid positions and cluster assignments to find a local minimum of the cost function J.

2.3 Medical Image Analysis

The increasing abundance and availability of medical data is posing great challenges and opportunities to health care and medical research [20]. Amongst others, image recognition has been a very successful field in neural networks and has found numerous medical applications ranging back roughly twenty years [22]. Following the trend in artificial intelligence research, medical studies have increasingly adopted the CNN approach as well as its novel advancements and tendency towards deeper networks. The three main applications of CNNs in medical image analysis are: Semantic segmentation, detection and classification.

Segmentation models attempt to partition image content into meaningful regions such as tumours, organs or lesions in order to improve screening costs and accuracy. Both supervised and unsupervised approaches, as well as their combination have been utilised for this task. Object detection networks predict the presence of certain features or objects in a given image, e.g. cancerous tissue and classifiers predict target features given image data such as binary diagnoses or regression of age just to name an example. These are mostly trained in a semi- or fully- supervised manner. Most of the benchmark models for these tasks build on CNNs and are often times combined, however there exist some distinct differences and the literature reviewed focuses particularly on medical classification tasks. Relevant contributions are roughly categorised and summarised in this section. Further literature is presented and reviewed in, for example, Alipanahi et al. [1], Zhou et al. [2].

2.3.1 Brain Magnetic Resonance Tomography Analysis

Deep Learning research on Brain MRT scans is especially relevant to the presented work since they both work with three-dimensional medical images acquired through magnetic resonance imaging. Publications range from various disorder classification systems to segmentation-, detection- and enhancement-tasks as well as disease progression prediction and others [1]. Image classification has most notably been applied to the classification of disorders, particularly Alzheimer's disease which represents a considerable amount of relevant research. Early contributions utilised *Deep Belief Networks* (DBN) based on *Restricted Boltzmann Machine* (RBM) layers [59] and stacked *Autoencoders* (AE) to classify disorders. The following publications adopted the use of Deep Learning approaches based on three-dimensional convolutions to improve benchmark scores [60, 61] and recent contributions utilise *ResNets* [62, 33] among other advanced and deep architectures.

2.3.2 Chest X-Ray and Computed Tomography Imaging

Classification systems have also been adopted in chest X-rays an *Computed Tomography* (CT) scans. A considerable amount of research has been conducted with two-dimensional input images, however an adaption to three dimensions is feasible for many of the approaches. Accurate estimation of lung cancer probability is an important challenge and the detection, characterisation and segmentation of lung nodules [63] has received a considerable amount of research attention [1]. Furthermore, chest images have been utilised for the prediction of Pulmonary Tubercolosis [64], Pneumonia [10], Interstitial lung disease [65, 66] and Pathology detection [67]. Similarly to other fields of research, CNNs have been increasingly adopted and top performing submissions to relevant competitions all include convolutional networks [1]. Deep CNNs and novel approaches have been successfully leveraged with implementations of *GoogleLeNet* [35, 67, 68], *AlexNet* [23, 68] and *DenseNet* [5] which exceeded human radiologist performance in classifying Pneumonia from X-Ray scans [10]. Very recently, Ardila et al. [11] have published their successful approach to lung cancer detection from CT scans consisting of multiple CNNs for various sub-tasks. Cancer risk prediction was performed based on three-dimensional convolutional inception networks [69, 37] and outperformed human expert classifications by significant margins.

2.3.3 Mammography

Mammography analysis for breast cancer prediction and research has been one of the earliest contributions to medical image analysis using machine learning models [70]. The main applications of CNNs regarding mammography are detection, segmentation and classification of lesions. Other imaging techniques such as ultra-sound, tomosynthesis and shear wave elastography have also been studied with regard to breast cancer analysis but mammography has received the most attention due to its wide application in medicine [1]. Successful image recognition models could be utilised on Mammography data [71, 72] as well as combinations of CNNs with traditional computer aided diagnosis models [73] and were able to surpass human radiologist performance as measured by area under the receiver operating characteristic [74].

2.3.4 Pathology

There is a large amount of research dealing with pathological microscopy image data sets regarding the detection, segmentation and classification of nuclei, segmentation of large organs as well as disease classification at lesion level [1]. Most related to the presented work are convolutional networks which have been utilised to classify nuclei and metastases concerning various types of cancer [75]. An implementation of *GoogleLeNet* [35] achieved 92.5% AUROC of the *Receiver*

Operating Characteristics (ROC) curve at the prediction of metastatic breast cancer and used the trained network to produce gradient heat-maps. Through augmentation by a medical expert, the performance could be further increased [76]. Similarly, Liu et al. [77] adopted the *Inception (V3)* [37] architecture on high resolution images and was able to improve previous AUROC benchmarks to 97% in breast cancer classification. CNNs applied to the classification of skin cancer from lesion images were also able to exceed dermatologist-level accuracy benchmarks [78]. The AUROC metric is discussed in more detail in section 3.3.2.

2.3.5 Diabetes Related Research Applications of CNNs

Diabetic Retinopathy

An implementation of Deep Learning approaches for the detection of Diabetic Retinopathy in retinal fundus images has received attention recently [4]. Their work used a CNN initialised with weights pre-trained on the *ImageNet* [69] data set and otherwise standard implementation to detect different sub-types of Diabetic Retinopathy. A performance benchmark was provided by trained professional graders and the deep learning model showed promising results, coming close to human expert performance and even beating it in certain metrics. Another interesting contribution used retinal images to predict various features related to cardiovascular risk factors [79] such as age, gender, blood pressure and smoking statues which were previously not thought to be quantifiable solely from retinal photographs. Gradient heat maps were generated to visualise network inference which are shown in Figure 2.5.

Deep Learning on Full-Body MRT Images

To the best of our knowledge, this work is the first to apply Deep Learning techniques on full-body MRT scans with regard to diabetes features.

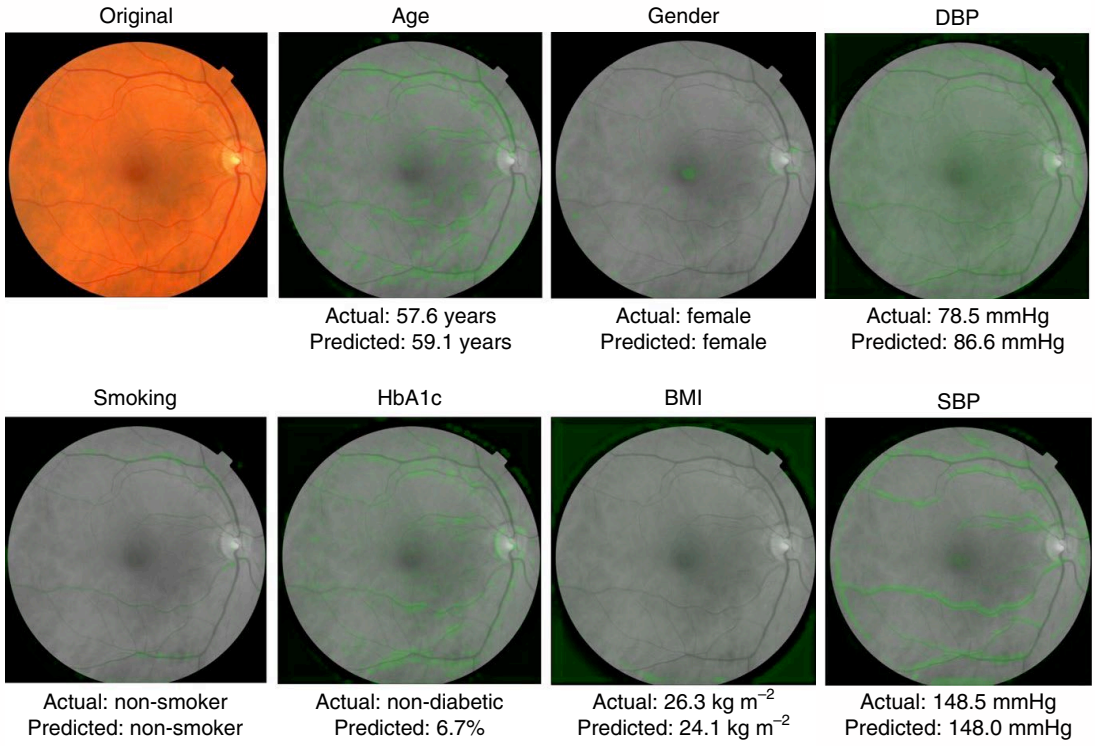

Figure 2.5: Gradient Maps of Retinal Images. Image taken from [79]. The top left image depicted an example of the original input. The remaining images show the respective attention-maps for several target labels. Predictions and labels are denoted below each attention image. The model was capable of predicting features that were previously not thought to be present in the eye, *e.g.* gender [79].

Chapter 3

Methods and Implementation

Our approach consisted of three main steps: First, we provided an overview over the data and pre-processing steps. Afterwards, we discussed the network architecture, implementation details and hyperparameters. Finally, we summarised network training as well as metrics and model selection and, lastly, outlined all post-processing steps involved such as visualisation, clusterings and gradient heat map generation.

3.1 Data and Preprocessing

The data used for this project has been generated and collected by the University Clinic of Tübingen. It consisted of full body MRT scans (Figure 3.1) with respective additional features.

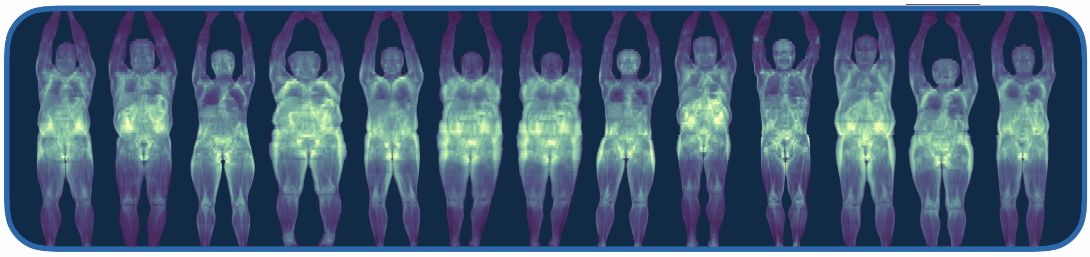

Figure 3.1: Examples of MRT scans from our Dataset. We averaged the images along the z-axis to generate two-dimensional images. After the described preprocessing steps, each scan had the dimensions $[85 \times 110 \times 135]$. The images were stretched along the body height axis, as this axis had a lower resolution than the remaining two.

3.1.1 Data

MRT voxel arrays were provided in the *Digital Imaging and Communications in Medicine* (DICOM) file format. The dataset consisted of 2555 full-body scans of 1080 patients, as some had been scanned multiple times. In total, there were 36 feature columns available. Four of these were binary labels: One for gender and the remaining three for different diabetes definitions: *diabetes* (D_α), *prediabetes* (D_β) and a third definition that consisted of diabetes positives as well as *Impaired Fasting Glucose* (IFG) and *Impaired Glucose Tolerance* (IGT) patients. We referred to the third diabetes label as *diabetes IFG+IGT* (D_γ).

Furthermore, there were 32 continuous features (some were binned, *e.g.* age was provided in years as an integer). In addition to the four binary features, we used age [years], *Body Mass Index* (BMI) [$\mathrm{kg} \cdot \mathrm{m}^{-2}$], *Insulin Sensitivity* (IS) [1] and (HbA1c) [%] as target labels for the network. HbA1c measurements are used for common diabetes diagnoses with our used label *diabetes* (D_α) having been defined as HbA1c$>$ 6.5% and HbA1c$>$ 5.7% for *prediabetes* (D_β), respectively.

Women were slightly over-represented with roughly 61% of available MRT scans. Table 3.1 outlines the distributions of the remaining target features. Most notably, D_γ and particularly D_α were considerably unbalanced. We did not use any of the other features for the deep learning model, however some of them were used for visualisations.

	D_α	D_β	D_γ	Age	BMI	IS	HbA1c
Male	2.8%	42.2%	18.4%	51±13.6	29±4.7	14±9.4	5.3±1.4
Female	1.9%	46.1%	12.9%	48±12.9	29±6.0	14±8.8	5.5±0.8
Total	2%	45%	15%	49±13.3	29±5.5	14±9.1	5.4±1.1

Table 3.1: Summary of Target Feature Distributions. We provided percentage ratios of positives for the binary labels. Regression targets were represented as 'mean ± standard deviation'. Preprocessing, as discussed in 3.1.2, had been applied to the data prior to the computation of the presented metrics.

The dataset had been collected as part of a prediabetes study, hence the distribution of our data did not represent the general population. We provided a comparison of BMI to the general German public in Table 3.2. Evidently there was a significant bias towards more overweight samples in our dataset. Furthermore, due to the setting of the study, the ratio of diabetes positives was smaller than

	German Population	Our Dataset
BMI $\leqslant 25$	47.3%	22.5%
$25 <$ BMI $\leqslant 30$	36.4%	36.6%
BMI > 30	16.3%	40.8%
$\varnothing$ BMI $[\frac{kg}{m^2}]$	26.0	29.3

Table 3.2: BMI Comparison to German Population. We compared the BMI distribution of our dataset to statistics from the German population in 2017 [80]. Our data was considerably biased towards overweight samples.

for the population, which is estimated to be approximately 10% [16]. Benchmark statistics for the two remaining diabetes definitions were unavailable. However, Chinese and North American studies have reported prediabetes prevalence of 36% [81] and 38% [82], respectively, thus considerably lower than in our dataset.

3.1.2 Preprocessing

Shape Normalisation

MRT scans were acquired by generating image slices along the patient's horizontal plane. Hence the body was represented by stacked slices of dimension $[150 \times 250]$. Unlike the slice dimension, the number of slices varied according to body height. The most frequent number of slices was 95, thus all scans with different heights were linearly interpolated along the vertical body axis to produce arrays of shape $[95 \times 150 \times 250]$. The voxel grid resolution of the two horizontal axes (considering a standing person) was considerably higher than the resolution along the body height axis, with negligible differences in z-axis scaling due to aforementioned interpolation. We did not correct the lower resolution on the height-axis using further interpolations, however, we down-sampled the standardised voxel grids for computational efficiency to their final dimension of $[85 \times 110 \times 135]$.

Pixel Value Normalisation

All three-dimensional pixels, *i.e.* voxels which did not belong to the body were identified using value distributions and set to zero. We standardised body voxel values to have a mean of zero and a standard deviation of one, truncated and subsequently shifted the distribution to strictly positive values to keep the distinction from the image background, *i.e.* surrounding air. Figure 3.2 provided the normalised density function for a subset of the MRT scans before the mean- and variance-adjustment.

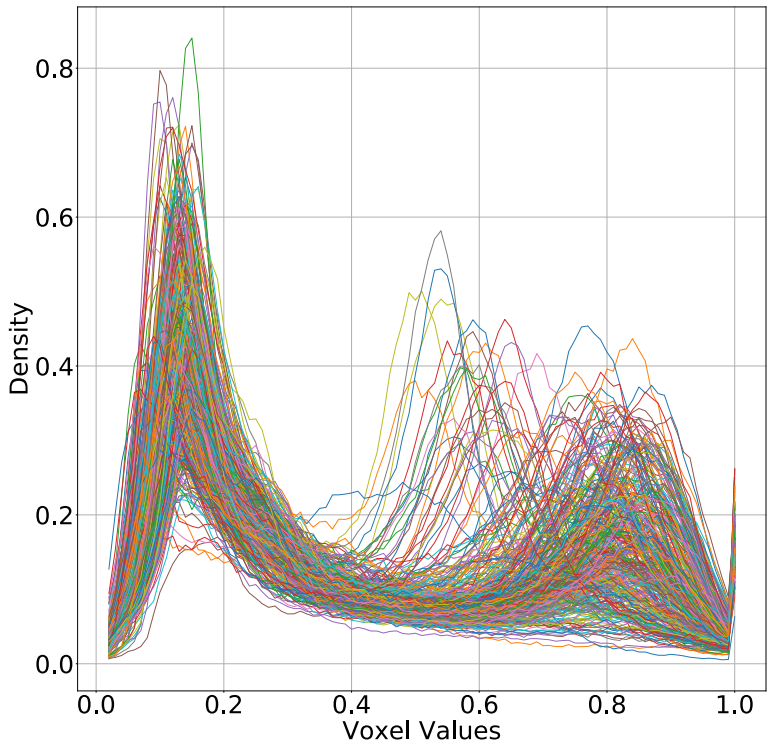

Figure 3.2: Normalised Voxel Value Distribution. We plotted the normalised voxel value density function of a subset of our dataset for all non-zero values. For the network input node, we subtracted the means from the distribution, divided it by its variance and finally shifted the voxel distribution to strictly postive values.

Feature Processing

There was a total number of 36 features available, eight of which were used as target variables. Samples with missing target labels were excluded and continuous variables were normalised to $[0, 1]$. Outliers were removed and missing values which were not used for prediction tasks were set to zero.

Data Partitioning

The dataset was stratified three-fold using BMI, Insulin sensitivity and prediabetes as stratification features. Samples were split into 70% training data and two sets of each 15% used for validation and testing, respectively. Our stratification algorithm required that each multivariate 'bucket' contains more than one sample, thus a certain number of additional outliers was removed, resulting in a total number of 2371 samples available for training and analysis.

Augmentation

To increase model robustness, the input arrays were augmented in several ways. Additional zero-padding was added to each dimension, increasing the size of the three-dimensional image array. To augment the training samples, padding was executed randomly, thus shifting the bodies inside their bounding box along all three dimensions. Furthermore a series of rotations was randomly performed on each input array as well as addition of Gaussian noise to all body voxels. For testing and validation, the body arrays were centered and no rotation or noise was applied.

3.2 Deep Learning

3.2.1 Model Architecture

Densely Connected Convolutional Layers

We built the network in accordance to the DenseNet architecture [5]. The three-dimensional voxel grids were fed to the input layer in mini-batches of eight. An initial batch normalisation and convolutional layer with a kernel size of five and relatively low number of activation maps (*e.g.* eight) was used as input layer. Subsequently, the feature map were down-sampled using a pooling layer to improve computation efficiency and its output was fed to the first dense block. Alternating dense blocks and transition layers were sequentially added, to process the input.

Following the final transition layer, the activation maps were flattened to a one-dimensional array and passed to three sequential densely connected layers with dropout. The output of the dense layers had the dimension 1×128 and was referred to as *embedding layer*. The embedding layer was used for the prediction of the desired target labels as well as input to both, the t-SNE embedding visualisation, as well as the unsupervised clustering analysis. For the prediction of the output nodes, subsequent dense layers were added to the embedding layer. A schematic of the entire model was provided in Figure 3.3.

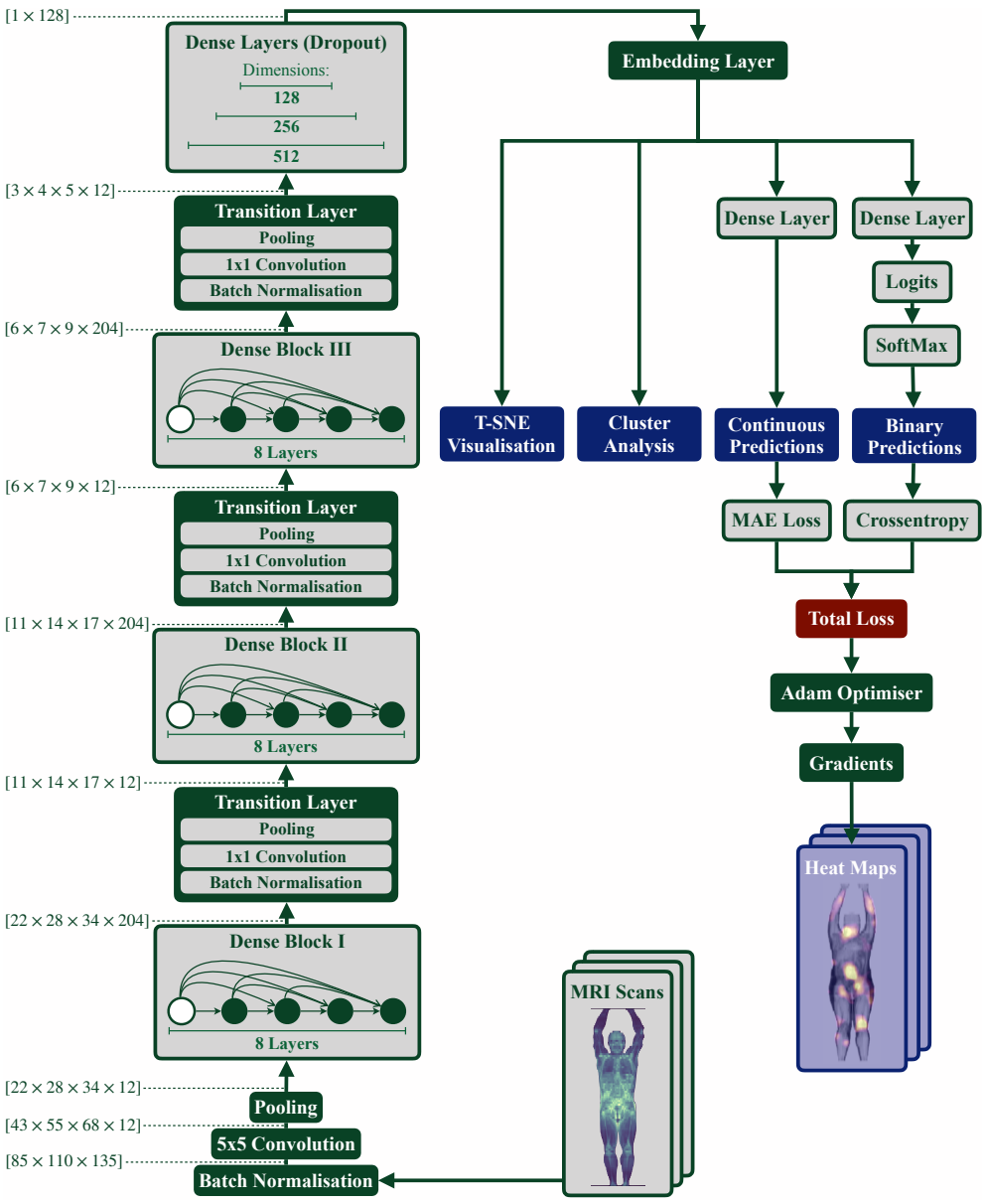

Figure 3.3: Schematic of used Model Architecture. The input to the network consisted of mini-batches of three-dimensional MRI scans. The input was passed to a batch normalisation layer as well as an initial convolution and pooling operation. Afterwards, three dense blocks were built in series with transition layers following each dense block. Finally, we used a series of densely connected layers with dropout to map the output of the final transition layer to the embedding layer of dimension 1×128 (Dimensions between network layers were denoted on the left). The embedding layer was used as input for postprocessing and predictions. We optimised the sum of all prediction losses to optimise the network and used the trained model to generate gradient maps.

Binary Classification

There were three binary classifications for different definitions of diabetes and an additional, fourth one for gender prediction. For each of them, a fully connected layer predicted two outputs $\boldsymbol{z}_i$ from the embedding layer for positive/ negative and male/ female, respectively. A subsequent *softmax* layer [17], defined as

$$\text{softmax}(\boldsymbol{z})_i = \frac{e^{\boldsymbol{z}_i}}{\sum_{j=1}^{K} e^{\boldsymbol{z}_j}} \tag{3.1}$$

transformed its inputs $\boldsymbol{z}_i$ into $\text{softmax}(\boldsymbol{z})_i$, representing the probabilities $P(0|\boldsymbol{x}_i)$ and $P(1|\boldsymbol{x}_i)$ for the implemented binary classifications with $\boldsymbol{x}_i$ being the corresponding MRT scan input. The loss function was chosen to penalise the divergence between predicted- and true class distributions. To this end, *crossentropy* [17] was used, defined by

$$H(p,q) = H(p) + D_{KL}(p||q) = -\sum_{x\in\mathcal{X}} p(\boldsymbol{x}) \log q(\boldsymbol{x};\boldsymbol{\theta}) \tag{3.2}$$

for a fixed label distribution $p(\boldsymbol{x})$ and the respective *softmax* output predictions $q(\boldsymbol{x};\boldsymbol{\theta})$ which are a function of the input $\boldsymbol{x}$ as well as the network weights $\boldsymbol{\theta}$. $D_{KL}(p||q)$ represented the *Kullback-Leibler divergence* (KL divergence) [17]

$$D_{KL}(p||q) = \mathbf{E}_{x\sim P}\left[\log \frac{P(x)}{Q(x)}\right] = \mathbf{E}_{x\sim P}[\log P(x) - \log Q(x)] \tag{3.3}$$

which, for two probability distributions p and q over the same random variable, quantifies their probabilistic distance. For $p = q$, the KL divergence equals zero and increases for diverging distributions. Minimising the cross entropy $H(p,q)$ with respect to q is equivalent to a minimisation of the KL divergence as $H(p)$ is a fixed term. We computed the crossentropy for each prediction and averaged over all considered samples to produce the cross entropy loss

$$L_{CE}(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{N}\sum_{n=1}^{N} H(p_n(\boldsymbol{x}), q_n(\boldsymbol{x};\boldsymbol{\theta})) \tag{3.4}$$

which we differentiated with respect to the network weights $\boldsymbol{\theta}$ to update the weights using back propagation.

Regression

All of the other target features are represented by continuous variables and, similarly to the classification sub-networks, their predictions were generated through fully-connected layers going from the embedding layer to a single output node. For the regression output layers, we used a sigmoid activations since the regres-

sion labels were normalised. Given the predictions, *i.e.* sigmoid outputs $z_i(\boldsymbol{x};\boldsymbol{\theta})$ and corresponding labels $y_i(\boldsymbol{x})$, we computed the *mean absolute error* (MAE) as

$$L_{MAE}(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N}\sqrt{[z_i(\boldsymbol{x};\boldsymbol{\theta}) - y_i(\boldsymbol{x})]^2} \tag{3.5}$$

Loss

For the classifications, Cross Entropy was used as loss function using the output of the softmax layers. Regressions were penalised using their mean absolute error. The final loss was the sum of all losses. We passed the final loss to the optimiser for back propagation and weight updates.

Gradient Heat Maps

Predictions for the various target variables were directly represented by the output nodes. Differentiating these with respect to previous convolutional layers yields pixel-wise gradients. As previously discussed, the chosen approach to heat map generation was *Gradients×Input* [43], hence we computed the gradients of the individual outputs with respect to the image input. Differentiating resulted in target-specific gradients of the same dimension as the input scans. Further analysis is provided later.

Hyperparameters

We used a growth factor $k_{\text{Growth}} = 32$. The initial convolution layer, prior to the first dense block had a kernel size of $[5 \times 5 \times 5]$ and generated between twelve activation maps. We chose to evaulate the model with three dense blocks and three subsequent fully connected layers, downsampling the flattened representation to 512, 256 and 128, respectively. With the sole exception of the final regression output layers, all activation functions throughout the network were *Exponential Linear Units* (ELU) [26]. We chose the initialisation, proposed by He et al. [83], for all weights of the model.

Adam [84] was used as optimiser with an initial learning rate of 10^{-4}. All other hyper parameters of the optimiser were kept to their *Tensorflow* [85] implementation defaults. The learning rate is adapted during training through a *Tensorflow* variable and has a cyclic, exponentially decay.

3.3 Training and Validation

3.3.1 Training

Network training was performed on a Nvidia Tesla V100-PCIE 32GB GPU, using the *Compute Unified Device Architecture* (CUDA) framework [86]. Computational resources were provided by the *Max Planck Institute for Intelligent Systems*. The network was trained for a maximum of 250 epochs with a batch size of eight, due to the considerable memory requirements of our three-dimensional voxel grids. The network converged after approximately two to three days, depending on network depth, *i.e.* number of trainable parameters as well as batch size and other hyper parameters.

3.3.2 Validation Metrics

Regression

Mean Average Error (MAE) was chosen as performance metric for the regressions. For predictions z_i and labels y_i, the MAE score was defined as

$$\text{MAE} = \frac{1}{N}\sum_{i=0}^{N}\sqrt{(z_i - y_i)^2} \tag{3.6}$$

Classification

We used the following definitions of accuracy, precision, recall, false-positive rate (FPR) and F1-Score. Metrics were functions of the numbers of true-positives (TP), false-positives (FP), true-negatives (TN) and false-negatives (FN).

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.7}$$

$$\text{precision} = \frac{TP}{TP + FP} \tag{3.8}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{3.9}$$

$$\text{FPR} = \frac{FP}{TP + FN} \tag{3.10}$$

$$\text{F1-Score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{3.11}$$

The output for the binary classifications contained two probabilities for the two possible classes. We used a probability threshold as classification rule. Threshold

value choice drastically impacts the characteristics of the classification model. The *Receiver Operating Characteristics* (ROC) curve plots recall against FPR for various thresholds and the area under the ROC curve (AUROC) represents a measure of separability of the two classes. A model capable of perfectly separating the classes has an AUROC of 1 while random guesses result in a diagonal ROC curve and an area of 0.5. We used the AUROC score as key metric for the classifications.

In addition to the three aforementioned binary diabetes definitions, we evaluated the classifier predictions on a fourth label D_ϕ, defined as $HbA1c \geqslant 6.0\%$ ($\sim 18\%$ of the dataset). This label has not been used for training and is evaluated using the combined predicted probabilities of the trained classifiers.

3.3.3 Model Selection

We frequently evaluated the network's performance and selected the model according to the highest diabetes (D_α) AUROC score on the validation set.

3.4 Postprocessing

The output of the model consisted of a set of predictions for each sample in addition to the respective gradient maps as well as its embedding space representation.

We used the gradient maps to compute target specific heat maps, using the *Gradients* $*$ *Input* method, proposed by Shrikumar et al. [43]. The individual output nodes were differentiated with respect to the input to produce three-dimensional feature-specific gradient maps. For visualisation, the gradient maps were post-processed using Gaussian filters in addition to contrast enhancements. Furthermore, for the classification nodes, only the output node corresponding to the correct label were considered. In other words, for a patient with label *female*, only the gradient that increased the *female* probability prediction was used for visualisation. All positive gradients were considered for the regression tasks.

We used a t-SNE embedding for dimensionality reduction to visualise the optimised, high-dimensional presentation of the MRT scans. Colored scatter plots were generated for visual inspection of the representation, according to different features. In addition to the t-SNE visualisation, we used the embedding layer representation of the dataset for an unsupervised k-means clustering and six centroids. The choice of number of centroid was arbitrary and chosen as a sensible choice to divide our dataset. With the exception of the number of centroids, we utilised the scikit-learn implementation of the k-means algorithm with standard parameters.

Chapter 4

Results

We reported the results of our analyses in the following chapter. First, an overview of the training progress was provided for all predictions. Subsequently, we summarised the performance of the final model using the previously discussed metrics, particularly for the classification tasks. Finally, we showed results of the unsupervised analysis of the embedding space. t-SNE embeddings were used to depict a two-dimensional representation of the embedding space and colour indexing provides an overview regarding the distribution of various features. A k-means clustering was used to partition the samples in the embedding space and the resulting distributions are depicted.

4.1 Performance Metrics

The performance was evaluated on three independent sets, acquired through stratification as described in Section 3.1.2. We first summarised the training progress, using the AUROC and MAE metric on the training set over all trained epochs. Afterwards we analysed the model in more detail. Metrics were generally computed on the test set with the exception of diabetes (D_α) and prediabetes (D_β). Due to the critically low number of positives for these labels, we chose to concatenate the datasets for testing and validation to compute the classification performance metrics. Finally we provide an overview of the three binary diabetes label classifications, showing the ROC curve in addition to precision-recall and F1-recall plots, also computed on the concatenated dataset.

4.1.1 Training

Training metrics were provided in Figure 4.1 for the classifications and regressions, respectively. We evaluated the model ten times per epoch on the validation set to track performances. The vertical, dotted line in both plots, as well as the circled dot in the left plot, represented the highest achieved diabetes AUROC scores on the validation set, *i.e.* the chosen model. The opaque plots showed the original output, additionally we plotted a smoothed version of the training progress.

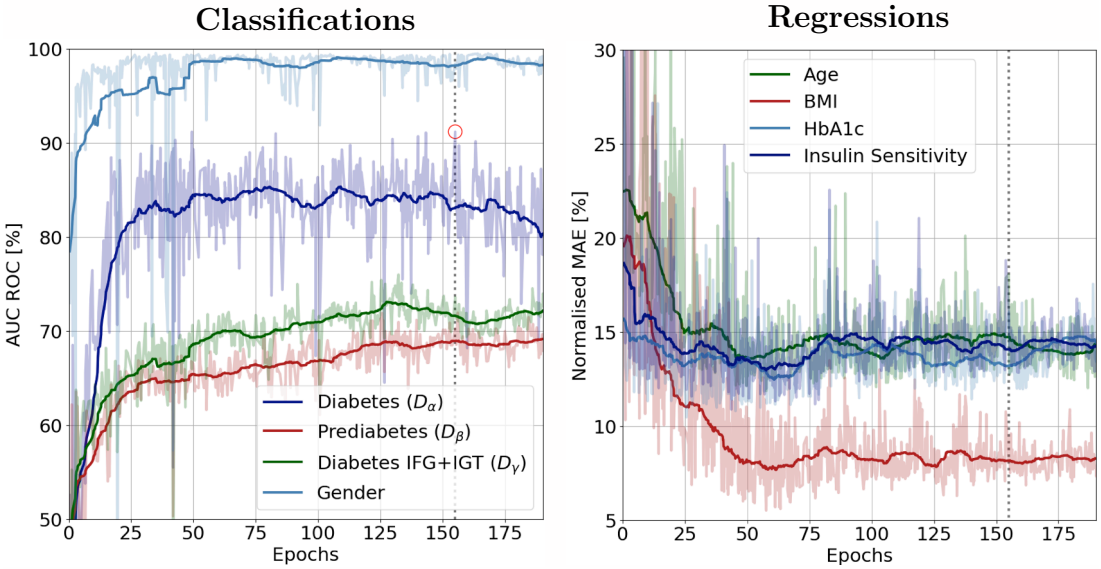

Figure 4.1: Training Metrics and Model Selection. We frequently evaluated the model's performance on the validation set during run-time and model selection was based on the best achieved diabetes AUROC as indicated by the dotted vertical line in both plots. The circled point in the left plot indicated the highest achieved diabetes ROC on the validation set.

Gender prediction converged to $\sim 99\%$ AUROC within the first ~ 25 epochs on the training set. All other labels tended to take considerably longer to converge and individual performances varied with different network parameters. While gender prediction seemed to be easily feasible for the network, the smoothed AUROC scores for the diabetes labels mostly peaked at $\sim 85\%$ for diabetes (D_α and $\sim 70\%$ for the two remaining binary labels. The considerably better performance of D_α is partially due to the significant imbalance of the label. Considering the regression tasks, all of the predictive performances vary slightly with different network parameters, however they generally converge to $\sim 5-15\%$ mean average error (MAE) on the normalised training labels before starting to overfit. Across all experiments, BMI tended to be the continuous feature with the lowest MAE.

4.1.2 Classification

There was a Gender label as well as three different binary diabetes definitions used to train the network. Additionally, we used a forth diabetes label, D_ϕ, for validation which hat not been used during training. D_ϕ was defined as $HbA1c \geqslant 6\%$. Figure 4.2 shows the ROC plots for the four initial binary labels. Gender and prediabetes D_β performances are computed on the test set. Due to highly unbalanced ratios and consequently very low numbers of positives, we concatenated the validation and test set to compute ROC scores for diabetes D_α as well as diabetes IFG+IGT D_γ. The dashed lines represent gender-specific ROC curves for the three diabetes labels, used for training.

There was a notable performance difference depending on gender, in particular for diabetes IFG+IGT (D_γ). Notably, the predictive performance of our model was consistently better for female samples. All of the diabetes classifications achieved AUROC scores considerably beyond 50%, indicating successfully learnt correlations. The prediction of Gender was accomplished very successfully with an AUROC score of $> 99\%$. Separation of genders was also well visible in the embedding visualisation, as well as the cluster analysis, provided later in this chapter.

Figure 4.3 summarised the classification performance for all diabetes-related labels using the concatenated version of the validation- and test set. The dashed lines indicate the performances of a benchmark acquired with a *Support Vector Machines Classifier*, trained on the training set using Insulin sensitivity and BMI as input features. F1-Score and a precision-recall plot were provided in addition to the ROC scores. F1-Scores were also plotted against the respective recall.

Both F1, as well as precision-recall were affected by the imbalance of the labels, resulting in relatively poor metrics. The performance of the diabetes labels strongly correlated with their prevalence in our dataset. However, the respective benchmarks were considerably improved by the network predictions for all diabetes classifications and performance metrics.

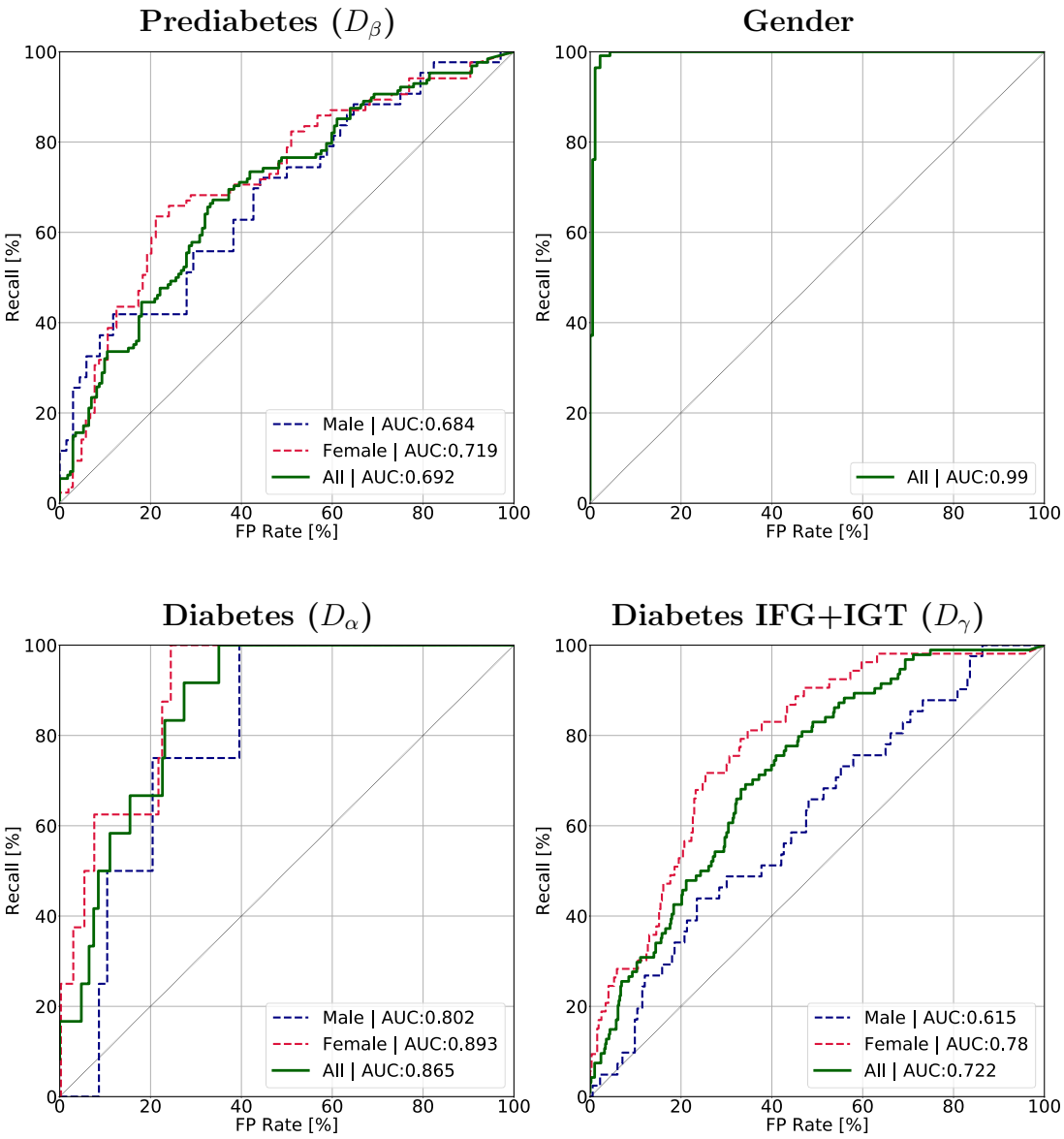

Figure 4.2: Classification Receiver Operating Characteristics. We computed prediabetes and gender classification performances exclusively on the independent test set. Due to insufficient numbers of true-positives, we combined the validation- and test sets to generate ROC curves for the two remaining diabetes labels. For the diabetes labels, dashed lines indicated the gender specific performances. Notably, the classifiation performance was better for female samples on all target labels. This discrepancy is particularly noticeable for the D_α label. Gender classification was performed successfully with an AUROC of approximately 99.5%.

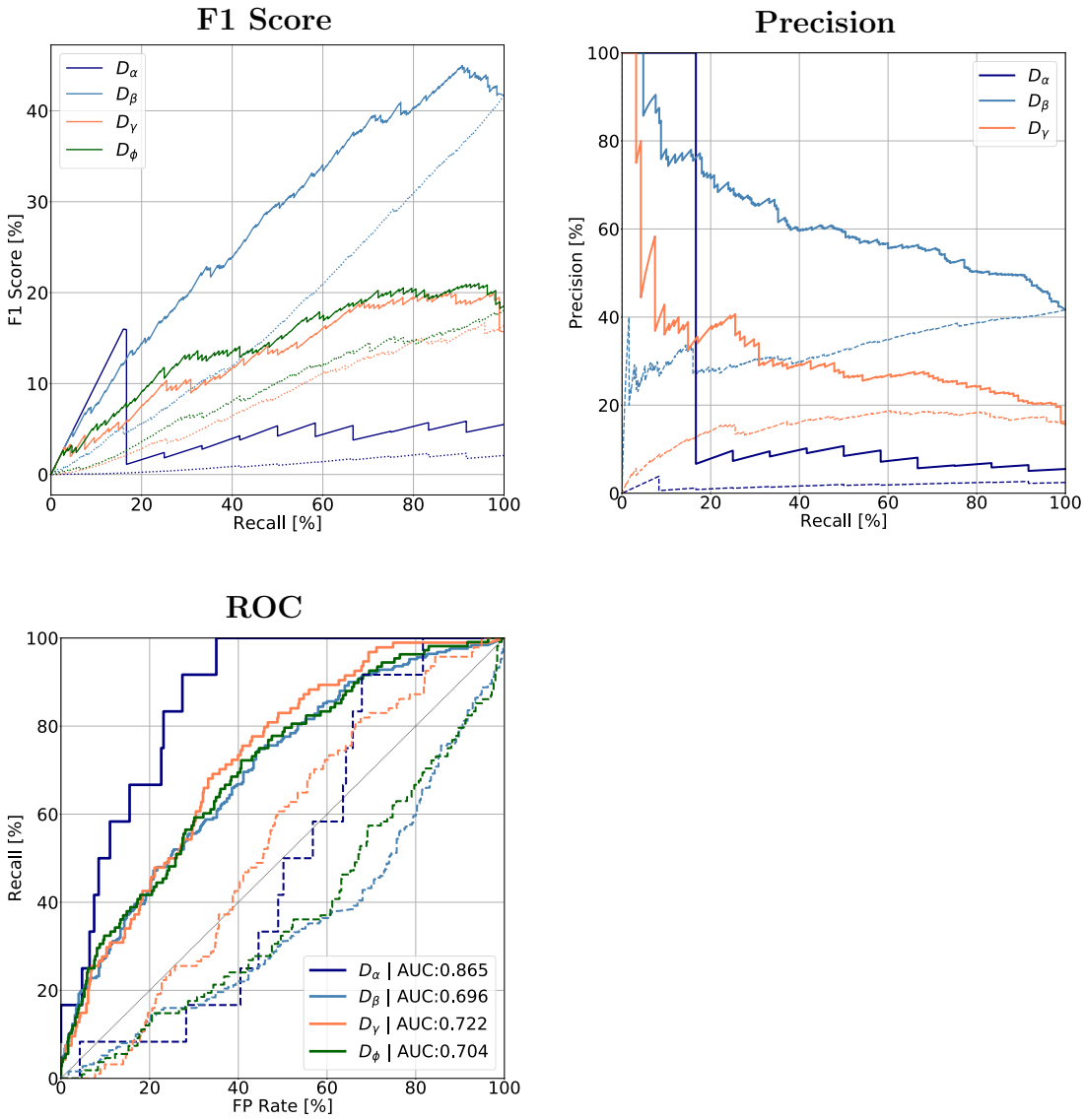

Figure 4.3: Diabetes Classification Performance Overview. We provided an overview over the diabetes classifications. To this end, we computed the F1-score as a function of recall as well as precision-recall and ROC plots. Label imbalance significantly affected performance metrics, resulting in relatively high AUROC scores, yet poor precision and F1 scores.

4.1.3 Summary and Comparison

We summarised the performance of model for varying hyperparameters in Table 4.1. Additionally, we provided benchmark AUROC scores, generated with a support vector classifier, optimised on the training set. The first row of Table 4.1 represented the model that has been used for clusterings and the embedding computation. A normalised MAE of 0.17 for age is equivalent so $\pm \sim 10$ years average error. Similarly, a normalised BMI MAE of 0.07 represents an average error of $\pm \sim 2$ and 0.13 normalised HbA1c MAE equals $\pm \sim 0.4\%$. Lastly, the Insulin sensitivity error of 0.26, *i.e.* an average error of $\pm \sim 10.2$, which represented the weakest regression performance in the analysed model.

Model		**Classification** [AUROC]				**Regression** [MAE]			
		D_α	D_β	D_γ	Gender	Age	BMI	Ins.S.	HbA1c
$\frac{\text{Layers}}{\text{DenseBlock}}$	$[8, 8, 8]$	87%	68%	72%	99%	0.17	0.07	0.26	0.13
k_{Growth}	32								
1^{st} Conv	$10 \times [3, 3, 3]$								
$\frac{\text{Layers}}{\text{DenseBlock}}$	$[8, 12, 16]$	73%	68%	71%	99%	0.17	0.23	0.18	0.23
k_{Growth}	32								
1^{st} Conv	$8 \times [3, 3, 3]$								
$\frac{\text{Layers}}{\text{DenseBlock}}$	$[8, 8, 8]$	78%	64%	72%	99%	0.18	0.09	0.13	0.14
k_{Growth}	24								
1^{st} Conv	$12 \times [3, 3, 3]$								
$\frac{\text{Layers}}{\text{DenseBlock}}$	$[10, 18]$	86%	71%	73%	99%	0.2	0.12	0.15	0.22
k_{Growth}	24								
1^{st} Conv	$8 \times [3, 3, 3]$								
$\frac{\text{Layers}}{\text{DenseBlock}}$	$[8, 10, 12]$	80%	70%	72%	99%	0.14	0.09	0.13	0.14
k_{Growth}	24								
1^{st} Conv	$10 \times [3, 3, 3]$								
Support Vector Classifier Benchmark Performance		49%	41%	66%					

Table 4.1: Model Comparison. Summary of performance metrics for models with varying hyperparameters. D_α and D_β are evaluated on the concatenated test- and validation sets. The remaining values are computed on the independent test set.

4.2 Embedding Visualisation

The embedding layer of dimension [1 × 128] was used for visualisation. To this end, t-SNE was utilised for dimensionality reduction to two dimensions. The resulting representation has been plotted in the scatter plots of Figure 4.4 and coloured according to different features. For gender, we used one colour for each. For the other provided plots we partitioned the samples into their bottom and top half for male and female patients, respectively.

The top-left scatter plot displayed the distribution of total adipose tissue. This label had not been used for network training. The remaining three features were explicitly used for network optimisation. Most notably, the separation of genders worked very well and resulted in two distinct clusters with almost complete gender-exclusivity (bottom left image).

The remaining features, apart from gender, revealed more or less distinct patterns in the two-dimensional representation, depending on the considered feature. We displayed Insulin sensitivity, total adipose tissue and prediabetes as example distributions in Figure 4.4.

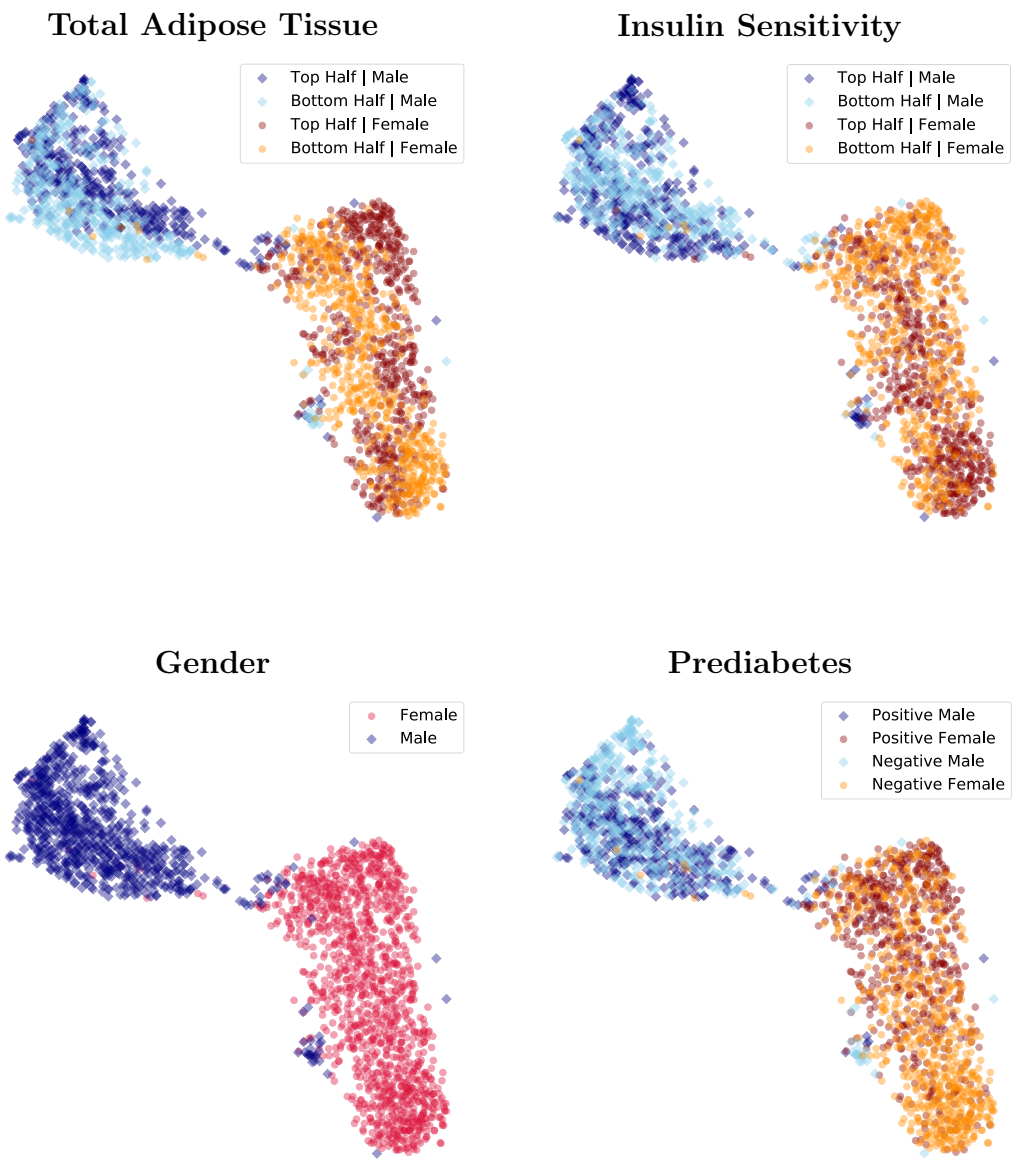

Figure 4.4: t-SNE Embedding Visualisation. The embedding layer representation of the entire dataset was used to generate the t-SNE plots. Colours were chosen according individual features. For the continuous features we partitioned the samples into their respective lower and upper half for each gender. The separation of genders is well visible in all of the plots. The remaining features demonstrated visible patterns in the two-dimensional representation.

4.3 Cluster Analysis

Figure 4.5 showed the distribution of binary features over the individual clusters. The distribution of the entire dataset was provided on the top right of Figure 4.5 and the distribution of samples over the six clusters was shown on the top left. The three diabetes-related features are represented by nested pie-charts with the smallest positives-ratio (D_α) in the centre and the highest positives-ratio (D_β) on the outer-most pie chart.

The result of the clustering was strongly dependent on gender with clusters generally being mostly female or mostly male. We found three, almost exclusively female clusters (2, 3, 5), where clusters 2 and 5 have fairly similar distributions while cluster 3 exhibits considerably higher ratios for all three labels. The prevalently male clusters where less gender-exclusive than their female counterparts, though they also showed high diabetes percentage clusters (4 and 6) as well as a 'healthier' cluster (1).

We provided violin plots for the cluster distribution of continuous features in Figure 4.6. BMI, age, fat bio-impedance and visceral adipose tissue were chosen as examples. The full dataset distribution is shown on the top line of each plot in orange. P-values for a two-sided t-test were computed to validate the statistical significance of the found clusters. All of the depicted distributions were significantly different from the full dataset with confidence levels $\geqslant 95\%$.

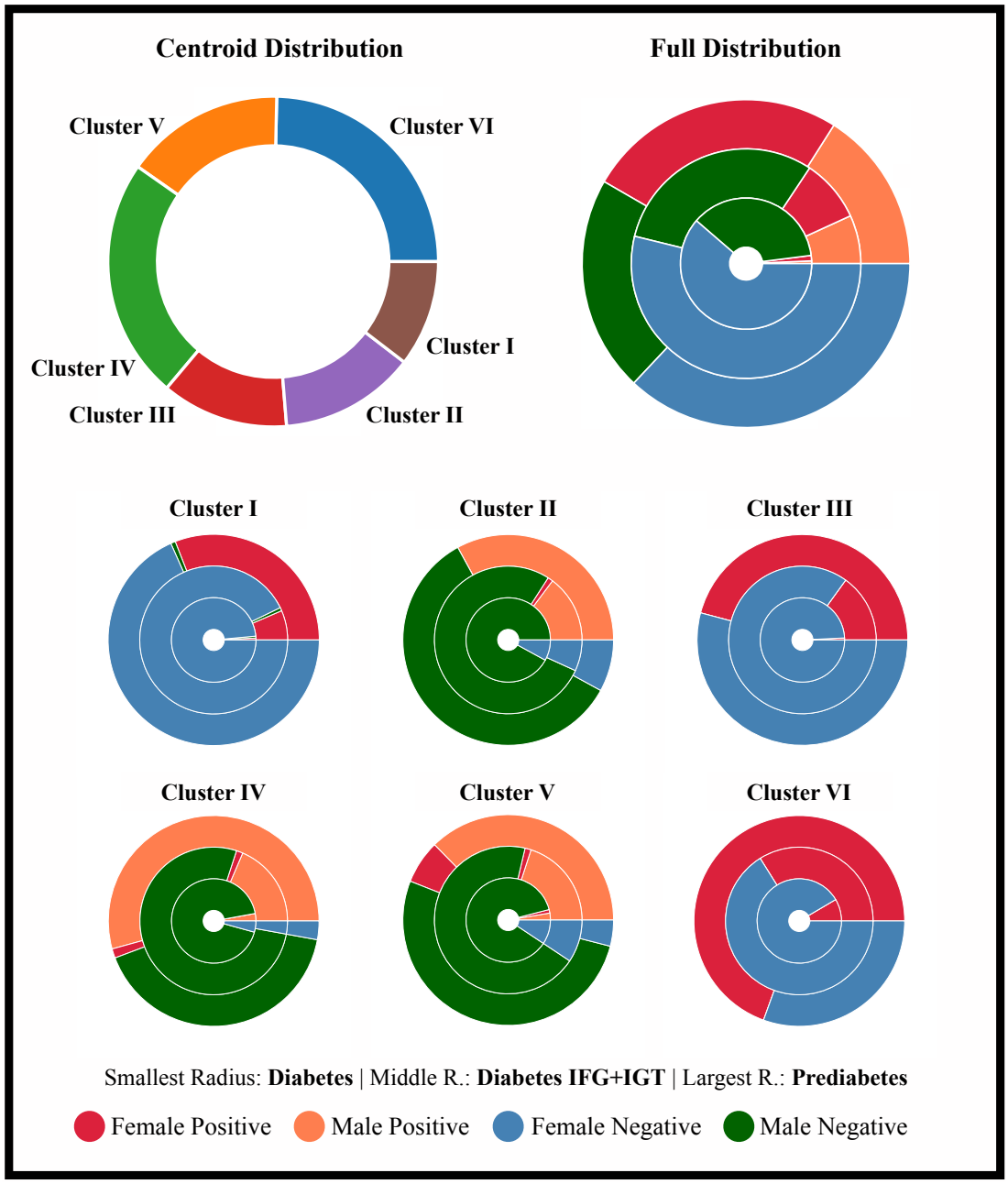

Figure 4.5: Summary of Cluster Distribution. Binary feature distributions were represented by nested pie charts. The distribution of the full dataset was shown on the top right and the distribution over the centroids was depicted on the top left. The pie charts represent the distributions of diabetes (D_α), prediabetes (D_β), diabetes IFG+IGT (D_γ).

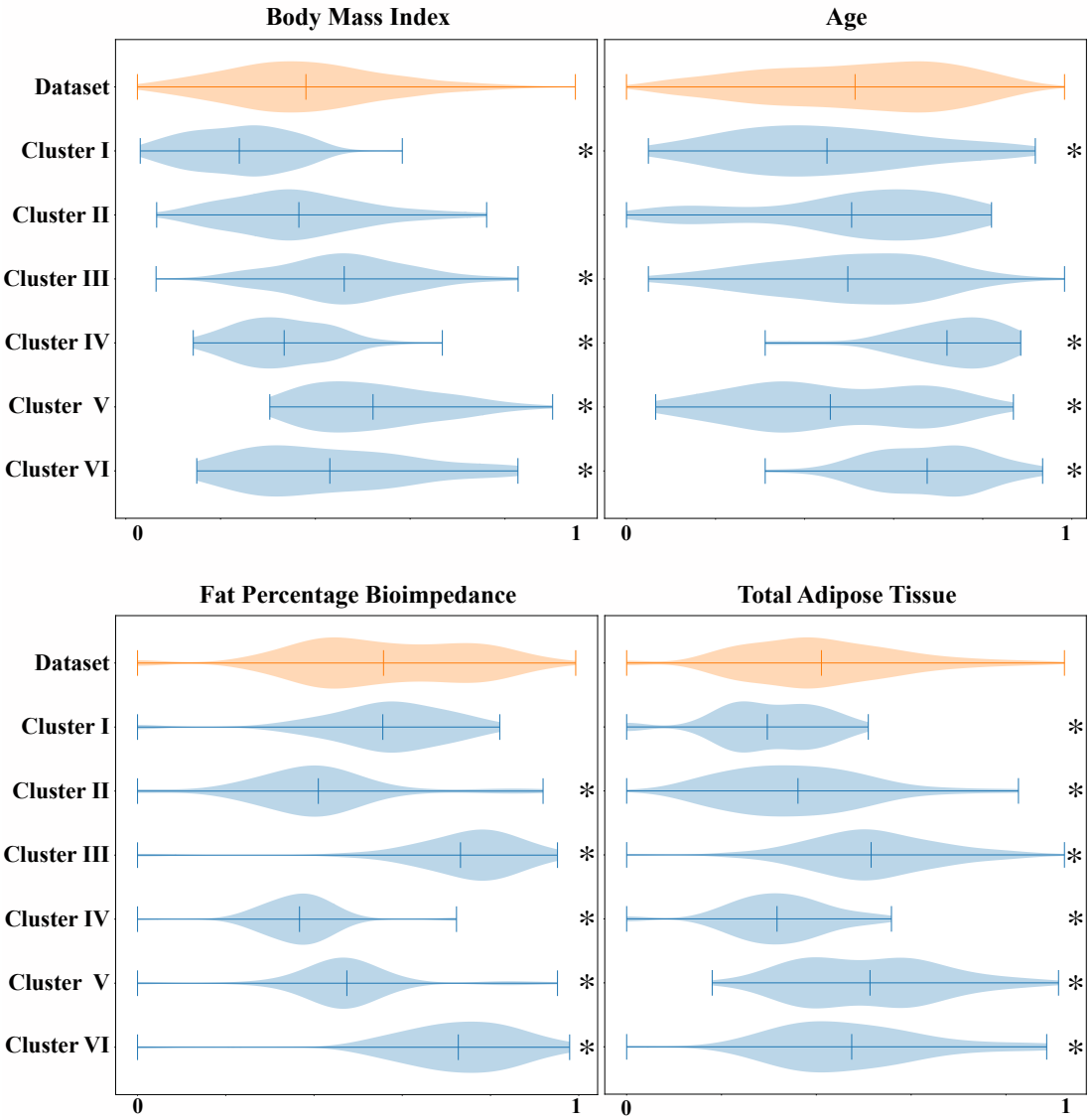

Figure 4.6: Continuous Feature Cluster Distributions. The top plot of each figure represents the entire dataset. The means of the individual clusters were analysed using a t-test and cluster distributions with respective p-values below 5% were indicated by an asterisk on the right side of the plots.

4.4 Target Specific Gradient Maps

We computed attention heat maps to acquire information about the reasoning behind predictions and to provide visualisations for further analyses. Comparison plots of heat maps for 50 random, prediabetes-positive samples for diabetes and Insulin Sensitivity are provided in Figure 4.7. An example for all predictions and corresponding heat maps for an arbitrary sample were provided in Figure 4.8.

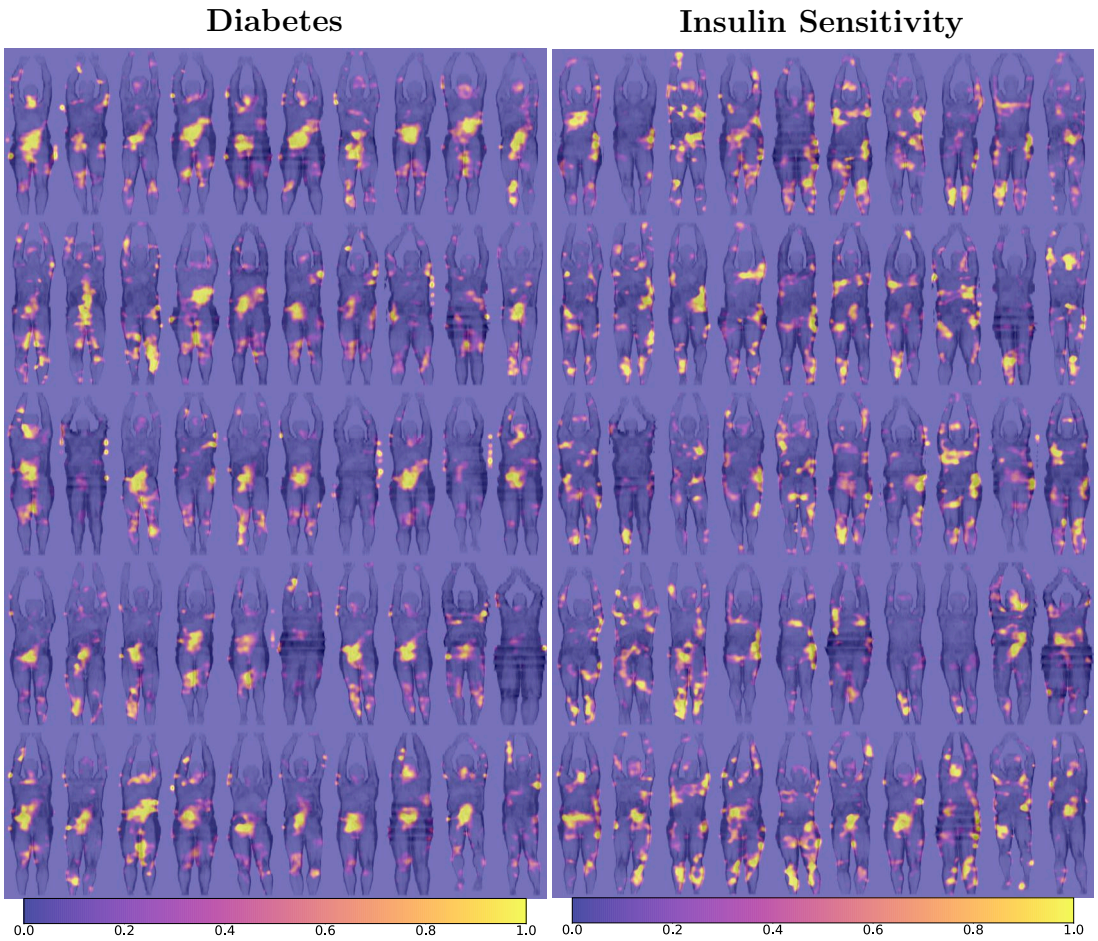

Figure 4.7: Gradient Map Comparison. We provided the gradient maps for diabetes and Insulin Sensitivity, computed for 50, randomly selected, prediabetes-positive samples. The body scans, as well as the gradient maps, were averaged along the z-axis to generate a two-dimensional representations.

4.4.1 Human Expert Gradient Classification

For evaluation and verification of the highlighted ares in the attention heat maps, we provided a random list of 100 attention maps (similar to Figure 4.8) for each feature to six human medical experts. The human professionals were asked to manually classify the most highlighted area in the attention maps according to Table 4.2, which summarises the results of the experiment.

Table 4.2: Classification of anatomical gradient locations by medical experts

	Age	BMI	Gender	Diabetes
Lower abdomen (visceral)	**66%**	**27%**	**57%**	**94%**
Upper abdomen right (visceral)	**31%**	**43%**	**20%**	**42%**
Upper abdomen left (visceral)	**16%**	**57%**	**7%**	**12%**
Thighs/lower abdominal	**56%**	**55%**	**42%**	**35%**
Legs (upper)	**67%**	**68%**	**77%**	**72%**
Legs (lower)	**33%**	**22%**	**61%**	**32%**
Neck	**43%**	**26%**	**11%**	**47%**
Mediastinum	**2%**	**12%**	**18%**	**9%**
Upper thorax/breasts	**13%**	**30%**	**77%**	**8%**
Arms	**43%**	**68%**	**81%**	**49%**
Head	**7%**	**2%**	**17%**	**11%**

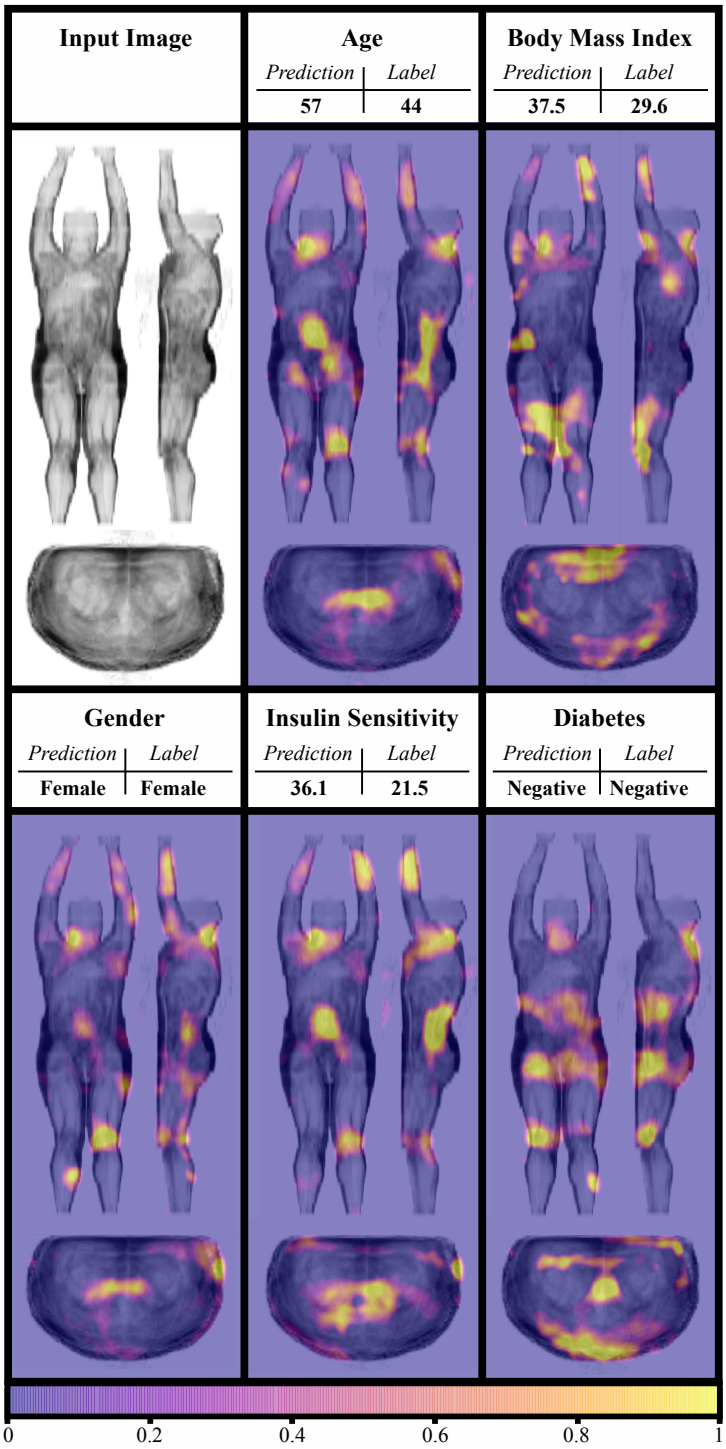

Figure 4.8: Gradient Heat Maps Example. The three-dimensional gradient arrays are averaged over each axis individually to generate three views. The top left image showed the original input. The remaining images show the heat maps for several target labels. For each predicted label, we provided the network prediction as well as the corresponding label. For visualisation, we applied a gaussian filter in addition to contrast enhacement to the heat maps.

Chapter 5

Discussion

We utilised a state-of-the-art CNN architecture, the DenseNet model [5], to predict diabetes and related features from full-body MRT scans. Our results provide a proof of concept, as well as performance benchmarks for future work. The AUROC score for the diagnosis of diabetes (D_α) reached $\sim 0.87\%$ while scores of ~ 0.68 and ~ 0.72 were achieved for the classification of the remaining labels, D_β and D_γ, respectively. Our model was able to surpass the benchmark classification performances for every metric and diabetes label. However, the precision and F1-score suffered due to the considerable label imbalance, particularly for D_α. Gender prediction was performed almost perfectly by the network.

The remaining, continuous predictions of the network varied slightly in performance, depending on network hyperparameters. BMI- and Insulin sensitivity prediction tended to have the lowest and largest MAE, respectively. Depending on the investigated feature, the resolution of the utilised scans posed a significant challenge for prediction, as only a certain level of detail was available. BMI prediction is arguably less affected by this circumstance, than features, such as, HbA1c or Insulin sensitivity.

While the network was evidently able to find corresponding correlations, further work needs to be conducted to improve predictive performance. The t-SNE embedding representation revealed visible patterns for several of the features and demonstrated the clear separation of the genders. The same holds for the k-means cluster analysis, which computed very gender-dependent cluster distributions. The vast majority of the computed cluster feature distribution proved to be statistically different from the full sample distribution.

From a medical perspective, the definition and diagnosis of diabetes is not trivial. Contrary to several other diseases, infections or injuries, a natural binary classification does not exist and the utilised methods for the diagnosis vary. One common test procedure is to measure the HbA1c level over an extended period of time and classify the patient accordingly [87], as described earlier. This method offers the advantage that no fasting or prior preparation is required. However, diabetes diagnosis solely based on HbA1c measurements have been found to significantly underestimate diabetes prevalence with a reported recall of 26.93%, when compared to diagnoses based on oral glucose tolerance testing [87].

Hence there lies an interest in novel, accurate procedures to diagnose diabetes or compliment existing test methods. MRT-based diagnosis may provide a viable option. Though limited capacity and considerable cost of MRT imaging present challenges for the clinical application of our presented approach at this point. In the future, however, full-body MRT scan analysis could potentially have the significant advantage of providing an analysis and diagnosis for several health risks and diseases, provided that respective data becomes available.

In conclusion we achieved an initial proof of concept for the diagnosis of diabetes and related features from full-body MRT scans using deep learning methods. The model's classification performance for all evaluated diabetes definitions surpassed benchmark scores and demonstrated the potential of the approach. Furthermore we provided a thorough analysis of the high-dimensional representation of the MRT scans, as well as label-specific heat maps for further analyses.

5.1 Future Work

Further research needs to conducted on the topic to both validate our results as well as to improve the model's performance. We provide a summary of suggested future work on the project.

The dataset utilised in this work provided a well-suited starting point for the diagnosis of diabetes and related features from full-body MRT scans. However, several limitations existed. The sample size of 2555 scans, despite being sufficient for our work, was rather on the lower bound for the application of deep learning models. Furthermore, the distribution of samples should be adjusted, such that it resembles the population distribution more closely and contains an increased number of positives, particularly for the D_α diabetes label. Finally, an improved resolution of the scans could enable the model to analyse parts of the body which, in our dataset, have been below the voxel size and, thus, unrecognisable.

Concerning our methods, we have trained a state-of-the art CNN image recognition model to analyse the entire three-dimensional scan to predict diabetes-related features. While the CNN approach to the task presented a reasonable starting point and standard method for comparable tasks in literature, there were other options which could be explored in future work. Possible, conceptionally different approaches were, among others, analysis of individual anatomical body parts or a attention-based networks [88].

Additional options regarding our used model, which have not yet been explored, include the consideration of the fact that more than one scan could belong to an individual as well as consideration of different MRT scanner models and used field strengths. Both of these variables have not been considered in this work.

Particularly if the dataset was extended or a larger one became available, an increased depth and width of the network might lead to further performance improvements. Due to the three-dimensional image inputs, however, an increase of model size is accompanied by considerable increases in computational requirements and training on a single 32GB GPU, as we have done, becomes unfeasible. The same challenge arises for resolution increases, as suggested above.

Finally, as previously discussed, several options existed for the visualisation of network attention. We chose an intuitive and easily adaptable approach, however, occlusion-models or more advanced gradient-methods might produce improved results and should be explored in future work.

Bibliography

[1] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, "Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning," *Nature Biotechnology*, vol. 33, no. 8, p. 831, 2015.

[2] S. K. Zhou, H. Greenspan, and D. Shen, *Deep learning for medical image analysis*. Academic Press, 2017.

[3] M. Cicero, A. Bilbily, E. Colak, T. Dowdell, B. Gray, K. Perampaladas, and J. Barfett, "Training and validating a deep convolutional neural network for computer-aided detection and classification of abnormalities on frontal chest radiographs," *Investigative Radiology*, vol. 52, no. 5, pp. 281–287, 2017.

[4] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros *et al.*, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *Journal of the American Medical Association*, vol. 316, no. 22, pp. 2402–2410, 2016.

[5] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[6] E. Cambria and B. White, "Jumping nlp curves: A review of natural language processing research," *IEEE Computational Intelligence Magazine*, vol. 9, no. 2, pp. 48–57, 2014.

[7] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *IEEE Computational Intelligence magazine*, vol. 13, no. 3, pp. 55–75, 2018.

[8] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[10] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya *et al.*, "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning," *arXiv preprint arXiv:1711.05225*, 2017.

[11] D. Ardila, A. P. Kiraly, S. Bharadwaj, B. Choi, J. J. Reicher, L. Peng, D. Tse, M. Etemadi, W. Ye, G. Corrado *et al.*, "End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography," *Nature medicine*, p. 1, 2019.

[12] C. L. Chen, A. Mahjoubfar, L.-C. Tai, I. K. Blaby, A. Huang, K. R. Niazi, and B. Jalali, "Deep learning in label-free cell classification," *Scientific Reports*, vol. 6, p. 21471, 2016.

[13] C. D. Naylor, "On the prospects for a (deep) learning health care system," *Journal of the American Medical Association*, vol. 320, no. 11, pp. 1099–1100, 2018.

[14] S. Liu, S. Liu, W. Cai, S. Pujol, R. Kikinis, and D. Feng, "Early diagnosis of Alzheimer's disease with deep learning," in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2014, pp. 1015–1018.

[15] A. D. Association *et al.*, "Diagnosis and classification of diabetes mellitus," *Diabetes Care*, vol. 33, no. Supplement 1, pp. S62–S69, 2010.

[16] "Deutscher gesundheitsbericht diabetes 2018," https://www.diabetesde.org/system/files/documents/gesundheitsbericht_2018.pdf, accessed: 2019-06-14.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT press, 2016.

[18] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological Review*, vol. 65, no. 6, p. 386, 1958.

[19] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[20] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 2017.

[21] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[22] S.-C. Lo, S.-L. Lou, J.-S. Lin, M. T. Freedman, M. V. Chien, and S. K. Mun, "Artificial convolution neural network techniques and applications for lung nodule detection," *IEEE Transactions on Medical Imaging*, vol. 14, no. 4, pp. 711–718, 1995.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[25] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, 2011, pp. 315–323.

[26] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[27] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[29] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[30] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.

[31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.

[32] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, "How does batch normalization help optimization?" in *Advances in Neural Information Processing Systems*, 2018, pp. 2483–2493.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[34] C. Szegedy, S. Ioffe, and V. Vanhoucke, "Inception-v4, inception-resnet and the impact of residual connections on learning," *Computing Research Repository*, vol. abs/1602.07261, 2016.

[35] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Computing Research Repository*, vol. abs/1409.4842, 2014.

[36] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2818–2826.

[38] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," in *Advances in Neural Information Processing systems*, 2015, pp. 2377–2385.

[39] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[40] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in *European Conference on Computer Vision*. Springer, 2016, pp. 646–661.

[41] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Artificial Intelligence and Statistics*, 2015, pp. 562–570.

[42] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, p. e0130140, 2015.

[43] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje, "Not just a black box: Learning important features through propagating activation differences," *arXiv preprint arXiv:1605.01713*, 2016.

[44] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," *arXiv preprint arXiv:1506.06579*, 2015.

[45] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3319–3328.

[46] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*. Springer, 2014, pp. 818–833.

[47] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Towards better understanding of gradient-based attribution methods for deep neural networks," *arXiv preprint arXiv:1711.06104*, 2017.

[48] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3145–3153.

[49] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[50] M. F. De Oliveira and H. Levkowitz, "From visual data exploration to visual data mining: a survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 9, no. 3, pp. 378–394, 2003.

[51] J. Tang, J. Liu, M. Zhang, and Q. Mei, "Visualizing large-scale and high-dimensional data," in *Proceedings of the 25th international conference on world wide web*. International World Wide Web Conferences Steering Committee, 2016, pp. 287–297.

[52] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[53] S. T. Roweis, L. K. Saul, and G. E. Hinton, "Global coordination of local linear models," in *Advances in Neural Information Processing Systems*, 2002, pp. 889–896.

[54] L. K. Saul and S. T. Roweis, "An introduction to locally linear embedding," *Unpublished. Available at: http://www. cs. toronto. edu/˜ roweis/lle/publications. html*, 2000.

[55] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Nnformation Processing Systems*, 2002, pp. 585–591.

[56] E. Amid and M. K. Warmuth, "A more globally accurate dimensionality reduction method using triplets," *arXiv preprint arXiv:1803.00854*, 2018.

[57] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[58] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[59] T. Brosch, R. Tam, A. D. N. Initiative *et al.*, "Manifold learning of brain MRIs by deep learning," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2013, pp. 633–640.

[60] E. Hosseini-Asl, R. Keynton, and A. El-Baz, "Alzheimer's disease diagnostics by adaptation of 3d convolutional network," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 126–130.

[61] A. Payan and G. Montana, "Predicting alzheimer's disease: a neuroimaging study with 3d convolutional neural networks," *arXiv preprint arXiv:1502.02506*, 2015.

[62] S. Korolev, A. Safiullin, M. Belyaev, and Y. Dodonova, "Residual and plain convolutional neural networks for 3d brain MRI classification," in *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, 2017, pp. 835–838.

[63] X. Huang, J. Shan, and V. Vaidya, "Lung nodule detection in CT using 3d convolutional neural networks," in *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, 2017, pp. 379–383.

[64] P. Lakhani and B. Sundaram, "Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks," *Radiology*, vol. 284, no. 2, pp. 574–582, 2017.

[65] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[66] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung pattern classification for interstitial lung diseases using a deep convolutional neural network," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1207–1216, 2016.

[67] Y. Bar, I. Diamant, L. Wolf, S. Lieberman, E. Konen, and H. Greenspan, "Chest pathology detection using deep learning with non-medical training," in *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2015, pp. 294–297.

[68] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[69] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, 2009, pp. 248–255.

[70] B. Sahiner, H.-P. Chan, N. Petrick, D. Wei, M. A. Helvie, D. D. Adler, and M. M. Goodsitt, "Classification of mass and normal breast tissue: a convolution neural network classifier with spatial domain and texture images," *IEEE Transactions on Medical Imaging*, vol. 15, no. 5, pp. 598–610, 1996.

[71] D. Ribli, A. Horváth, Z. Unger, P. Pollner, and I. Csabai, "Detecting and classifying lesions in mammograms with deep learning," *Scientific Reports*, vol. 8, no. 1, p. 4165, 2018.

[72] T. Kooi, G. Litjens, B. Van Ginneken, A. Gubern-Mérida, C. I. Sánchez, R. Mann, A. den Heeten, and N. Karssemeijer, "Large scale deep learning for computer aided detection of mammographic lesions," *Medical Image Analysis*, vol. 35, pp. 303–312, 2017.

[73] Y. Wang, M. Heidari, S. Mirniaharikandehei, J. Gong, W. Qian, Y. Qiu, and B. Zheng, "A hybrid deep learning approach to predict malignancy of breast lesions using mammograms," in *Medical Imaging 2018: Imaging Informatics for Healthcare, Research, and Applications*, vol. 10579. International Society for Optics and Photonics, 2018, p. 105790V.

[74] T. Kooi, A. Gubern-Merida, J.-J. Mordang, R. Mann, R. Pijnappel, K. Schuur, A. den Heeten, and N. Karssemeijer, "A comparison between a deep convolutional neural network and radiologists for classifying regions of interest in mammography," in *International Workshop on Breast Imaging*. Springer, 2016, pp. 51–56.

[75] A. Janowczyk and A. Madabhushi, "Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases," *Journal of Pathology Informatics*, vol. 7, 2016.

[76] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. H. Beck, "Deep learning for identifying metastatic breast cancer," *arXiv preprint arXiv:1606.05718*, 2016.

[77] Y. Liu, K. Gadepalli, M. Norouzi, G. E. Dahl, T. Kohlberger, A. Boyko, S. Venugopalan, A. Timofeev, P. Q. Nelson, G. S. Corrado *et al.*, "Detecting cancer metastases on gigapixel pathology images," *arXiv preprint arXiv:1703.02442*, 2017.

[78] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, p. 115, 2017.

[79] R. Poplin, A. V. Varadarajan, K. Blumer, Y. Liu, M. V. McConnell, G. S. Corrado, L. Peng, and D. R. Webster, "Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning," *Nature Biomedical Engineering*, vol. 2, no. 3, p. 158, 2018.

[80] "Gesundheitsberichterstattung des Bundes," http://www.gbe-bund.de/oowa921-install/servlet/oowa/aw92/dboowasys921.xwdevkit/xwd_init?gbe.isgbetol/xs_start_neu/&p_aid=i&p_aid=69998455&nummer=434&p_sprache=D&p_indsp=-&p_aid=41426509, accessed: 2019-06-14.

[81] L. Wang, P. Gao, M. Zhang, Z. Huang, D. Zhang, Q. Deng, Y. Li, Z. Zhao, X. Qin, D. Jin *et al.*, "Prevalence and ethnic pattern of diabetes and prediabetes in china in 2013," *Journal of the American Medical Association*, vol. 317, no. 24, pp. 2515–2523, 2017.

[82] A. Menke, S. Casagrande, L. Geiss, and C. C. Cowie, "Prevalence of and trends in diabetes among adults in the united states, 1988-2012," *Journal of the American Medical Association*, vol. 314, no. 10, pp. 1021–1029, 2015.

[83] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[84] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[85] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org.

[86] J. Nickolls, I. Buck, and M. Garland, "Scalable parallel programming," in *2008 IEEE Hot Chips 20 Symposium (HCS)*. IEEE, 2008, pp. 40–53.

[87] M. M. Chang Villacreses, W. Feng, R. Karnchanasorn, R. Samoa, and K. Chiu, "Sat-125 underestimation of the prevalence of diabetes and overestimation of the prevalence of glucose tolerance by using hemoglobin a1c criteria," *Journal of the Endocrine Society*, vol. 3, no. Supplement_1, pp. SAT–125, 2019.

[88] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3156–3164.