Loughborough University London

# Explainable Deep Learning: A Study on Understanding Learning Process of Convolutional Neural Networks with Information Theory

by

Yusuf Aslan

A thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Cyber Security and Big Data

at the

Loughborough University London

September 2020

## Abstract

The goal of this thesis was to understand the learning process of convolutional neural networks with information theoretic concepts. Although the use of Deep Learning is booming in many real-world tasks, their internal processes of how they draw the results are still uncertain. In this paper, an information theoretic approach is used to reveal the typical learning patterns of a convolutional neural network. For this purpose, training samples, true labels, and estimated labels are considered to be random variables. The mutual information and conditional entropy between these variables are then studied using our proposed method. The results of the numerical experiments conducted reveal that information theory is an excellent tool with which to explain convolutional neural networks. The first outcome of the results is that each layer has a different effect on learning. The layers that need to be added to a neural network to gain desired learning level can be determined with the help of information theoretic quantities. Secondly, the optimum number of training epochs and other parameters can be determined with information theory. It is inferred from the results that show the information theoretic quantities graph is parallel to the training accuracy graph. Overall, the experimentations show that information theoretic approach can be utilised to explain convolutional neural networks. This study and related future studies can be considered the foundation for Explainable Machine Learning studies.

# Table of Content

# CHAPTER 1 Introduction

## 1.1 Background

The field of artificial intelligence and machine learning have been developing rapidly over recent years. The success, especially in deep learning during the last decade, has been rapid with unpredictable achievements in international challenges. Deep learning algorithms have made remarkable progress on numerous machine learning tasks and dramatically improved the state-of-the-art in many functional areas ranging from visual image recognition to understanding languages from audio (Graves et al., 2013; Zhang and LeCun, 2015; Hinton et al., 2012; He et al. 2015; LeCun et al., 2015). As a result of this success, deep learning models have been used in various application areas such as criminal justice, medicine, and finance.

Despite their great success, there is still no comprehensive understanding of the optimisation process or the internal organisation of deep neural networks, and they are often criticised for being used as mysterious ”black boxes” (Alain and Bengio, 2016; Adadi and Berrada, 2018). Deep learning models usually contain millions of parameters and functions. People cannot understand this representation, and also cannot physically interpret the results of models. This lack of understanding can lead to a belief that the models are untrustworthy. Additionally, there is no way to know if the reasons behind the results are ill-formatted, biased, or even wrong, which can raise many ethical, financial, and legal issues. Studies on Explainable Machine Learning are currently attempting to provide explanation and solutions to these kinds of problems. Explainable Machine Learning models provide reasoning for the models' results.

## 1.2 Research Aim and Objectives

Convolutional neural networks are in high demand as models for a multitude of computer vision tasks. Although they are used to solve a variety of problems, the learning processes of convolutional neural networks are still not transparent. In recent years, many studies have been undertaken to explain these models. However, the theoretical understanding of convolutional neural networks is still insufficient.

With the motivation to make convolutional neural networks more intuitive, the aim of this thesis is stated as an attempt to understand the learning process of convolutional neural networks with information theory. This goal has been one of attempting to answer two fundamental questions: *What are the effects of hidden layers on learning of a convolutional neural network?* and *How does training affect the learning process of a convolutional neural network?.*

The research questions were investigated and studied in light of information theory. This idea was theoretically introduced and first proposed in Tishby's research on deep learning and information theory (Shwartz-Ziv and Tishby, 2017; Tishby and Zaslavsky, 2015 ). Later, some other researchers also utilised this theory to understand deep neural networks (Balda, Behboodi and Mathar, 2018). In the context of this dissertation, the previously proposed method for analysing deep neural network was adopted to investigate the learning process of convolutional neural networks.

## 1.3 Research Value

In this dissertation, a systematic method has been proposed to analyse the learning process of a convolutional neural network. By answering the research questions, the dissertation makes the following contributions:

1) By calculating mutual information of individual layers, their effect on the learning process can be inferred from these quantities.

2) By observing mutual information between input and output during training, the effects of the training on the learning process of the model can be examined. The results show that there is a parallelism between the information-theoretic approach and the mathematical approach when determining optimum training.

## 1.4 Thesis Outline

Beyond this introduction, this dissertation consists of six additional chapters.

In Chapter 2, an extensive literature review comprising three subsections will be given. Firstly, pieces of literature on big data and analytics will be provided. Some vital work into the understanding of deep learning will be described in the second section. Finally, the concept of explainable machine learning will be introduced, and some of the relevant literature will be discussed.

In Chapter 3, a description of fundamental information theory will be given to understand the results of this thesis. The reader is assumed to have prior basic probability theory knowledge, and no information theory is needed.

In Chapter 4, a detailed review and explanations of some of the more critical literature mentioned in Chapter 2 will be given. Previously proposed methods for analysing deep learning via information theory will be introduced and explained in this chapter.

In Chapter 5, the proposed method and experimental setups will be provided. Firstly, a brief introduction to convolutional neural networks will be given. Then, the steps to the proposed method required to understand the learning processes of convolutional neural networks via information theory will be explained.

In Chapter 6, the results of experimentation and further discussion will be given. Firstly, the results that show how hidden layers affects the learning process of a convolutional neural network will be described. Later, inferences and discussions about the results will be made in the same section. The effects of training on learning will be illustrated and discussed in the second section.

In Chapter 7, the work and results will be summarised in a series of concluding remarks, after which various ideas about possible future work will be given.

# CHAPTER 2 Literature Review

## 2.1 Big Data Analytics

The term “big data” and such technologies that make use of it or benefit from it have been in existence for many years. Since the world entered the so-called “digital age”, huge amounts of data are generated through various means and from different sources every day. The generated data add to the collection of digital data which is now referred to as Big Data (Chen and Lin, 2014; Zhou et al., 2020). Accordingly, there are several descriptions and definitions of Big Data in the literature. For example, it is described in one paper as follows:

“Big Data is the Information asset characterised by such a High Volume, Velocity and Variety to require specific Technology and Analytical Methods for its transformation into Value.” (De Mauro, Greco and Grimaldi, 2016)

From the definition, it is inferred that Big data is not just about the size of the data. There are also other essential attributes of data that matter. Hence, generally, it is described by three main characteristics depicted as “3V”, i.e., Volume, Velocity and Variety (Sagiroglu and Sinanc, 2013). The amount of data generated and collected is referred to as Volume. Velocity is the speed at which data is processed and collected. Variety refers to the diversity of different formats of data. Further, recent reports in the literature extend the above to the 5V model by adding two more features: Veracity and Value (Rao et al., 2018).

Big Data has no value whilst it remains in its core form; its potential can only be realised when it is used to guide decision-making processes. So, fast and efficient processes are needed to obtain meaningful insights from this data in a process called Big Data Analytics (Gandomi and Haider, 2015). These analytics can be applied to various kinds of dataset such as text, audio, and visual using different analysis techniques. According to the McKinsey report, several methods based on multiple fields such as statistics, computer science, applied mathematics and economics can be utilised for big data analytics. These methods include data mining, statistical learning, A/B testing, Natural Language Processing and Machine Learning (McKinsey, 2011). Statistical methods have historically been one of the most frequently used methods for data analysis, as is currently used to analyse big data. Other researchers have examined the use of statistical learning techniques for data analysis (Hastie, Tibshirani and Friedman, 2009). Also, Wu et al. (2014) and Feng and Zhu (2016) have noted that data mining is a well-known method for big data analysis for different fields. Moreover, the state-of-the-art Machine Learning algorithms which utilise statistical and artificial intelligence methods are used for big data analytics (Watson, 2019)

## 2.2 Understanding Deep Neural Network

Deep Learning, or so-called deep neural networks, is a subfield of machine learning inspired by the structure and function of brain neurons. The advantage of deep learning over the other types of machine learning is its scalable behaviour. Namely, the performance of deep learning models gets better as the amount of data used to train the model increases. The structure of deep neural networks consists of anything from several layers to millions of layers, which makes their mathematical explanation intractable. This nature of neural networks is thus somewhat “black box” in character (Castelvecchi, 2016).

Transparency which is aimed at a direct understanding of the model's learning process can be said to be the opposite of the black box approach. When the Neural Network structure is considered, input-output relations and model design can be expressed mathematically so that these properties can thus be defined as being transparent. However, layer parameters, the number of layers, and non-linear properties are generally determined by traditional methods and heuristics. Since there is no optimisation of these parameters in a mathematical sense, the associated transparency is somewhat limited. In addition to these, because of the selection of hyperparameters such as learning speed, the batch size is also intuitive and has no transparent algorithmic structure, so these networks are not reproducible. Thus, efforts are being made to make deep neural networks more understandable and transparent.

*Post-hoc* explainability techniques are generally applied to deep learning models to explain their decisions. This method aims to understand ow an already designed model (thus, it is also called post-modelling explainability) processes the information and gives the input (Roscher et al., 2020). These methods enhance the transparency of models that are not tractable for in terms of explainable models, such as with deep neural networks. Some basic approaches to achieving this goal include visual explanations, local explanations, and text explanations (Barredo Arrieta et al., 2020). Zeiler and Fergus (2013) have tried to explain convolutional neural networks with a novel visualisation technique that gives information about intermediate layers. Also, Bae, Moon and Kim (2019) have introduced a textual explanation deep learning model for self-driving cars to obtain safe autonomous devices.

Besides computer science and statistical methods, some other techniques and procedures can be adopted to explain deep learning models. Information theory of communications systems has recently become one of the most referenced methods. The work of Ziv and Tishby (2014), "Opening the Black Box of Deep Learning", as based on the Information Bottleneck Method (Tishby, Pereira and Bialek, 2000), has led to a focus on explaining neural networks via information-theoretic quantities such as mutual information. Other researchers have also investigated and discussed this approach (Saxe et al., 2018; Gabrié et al., 2019). Moreover, Yu and Principe (2019) and Yu et al. (2020) have introduced a new matrix-based Renyi's $\alpha$-entropy technique to analyse the information flow in stacked autoencoders and convolutional neural networks. Furthermore, Balda, Behboodi and Mathar (2018) have adopted the information-theoretic method and incorporated it with generalisation error and suggest learning process of neural networks.

## 2.3 Explainable Machine Learning

Machine Learning is a part of research into Artificial Intelligence that aims to give computers the ability to learn, and make and improve predictions based on data (Gilpin et al., 2019). There are several types of machine learning algorithms depending on their learning style, i.e., supervised learning, unsupervised learning, and semi-supervised learning. Nowadays, AI systems based on machine learning have been remarkably successful at various computer-related tasks (He et al., 2016), to understand natural language (Cho et al., 2014), and to play games such as Go (Silver et al., 2016). Most of machine learning models such as deep neural networks are too complicated for people to understand easily due to their non-intuitive and

opaque nature (Gunning and Aha, 2019). Hence, this lack of explainable of Machine Learning models acts as a barrier to the adoption of these models into such fields such as law, medicine, and transportation. For example, knowing why a car performed a particular action is vitally important when designing self-driving cars (Doshi-Velez and Kim, 2017). As machine learning methods started to be used to make important predictions at critical points, the demand for transparency from the stakeholders of AI began to increase (Preece et al., 2018). Thus, many researchers are now studying how to explain machine learning models. The aim of this research, which is generally called explainable machine learning, is to help people understand and trust machine learning models in an intuitive manner.

Explainable machine learning is such a new and broad topic for the research community that it is yet to be formally defined. Gilpin et al. (2019) define it as the "Science of understanding what a model did or might have done" and Murdoch et al. (2019) as the "Use of machine learning models for the extraction of relevant knowledge about domain relationships contained in data." There is a range of reasons why some form of an explanation of a machine learning model is desirable. Adadi and Berrada (2018) reported that justifying decisions, enhancing control, improving models, and discovering new knowledge are four fundamental reasons behind this desire. To achieving this, some basic questions must be answered: *How does the model work? Which inputs or features of the data are the most influential in determining an output?* and *What is the optimum mathematical representation of this model?* Explainable models can be divided into three categories according to the purpose of motivating the associated research. These are explainability, interpretability and transparency. The first two are mostly about making a model, its internal process, and its outputs intuitively humanly understandable. Transparency is about understanding the process of how the model or algorithm learns from the data (Lipton, 2018).

A model can be considered transparent if it is understandable by itself. The transparency of a model can be measured in two different ways. Easily understandable models can be explained by methods during design. However, the explanations of some models are not at first tractable. For example, the very first machine learning models based on probabilistic mappings such as Decision Trees, Logistic Regression and Clustering are convenient in terms of their explanations. Still, Deep Neural Networks are very hard to understanding (Barredo Arrieta et al., 2020), so recently many pieces of research have begun to consider the learning behaviour of neural networks to make them more transparent.

# CHAPTER 3 Fundamentals of Information Theory

Information theory is concerned with measuring information related to distributions as based on probability and statistics. It was initially proposed and developed by Claude Shannon for communication system design. This theory arises from the quest to determine how much information a signal contains.

In this chapter, some basic definitions and theorems of information theory that are needed to understand the results of this thesis are presented. The definitions, theorems, and formulae in the remainder of this chapter are mainly adopted from Shannon's famous paper "A Mathematical Theory of Communication" (Shannon, 1948) and the book "Elements of Information Theory" (Thomas and Cover, 1991).

## 3.1 Information Entropy

Information entropy or basically "entropy" is the measure of the uncertainty of a random variable that has a probability distribution. This quantity is described by the probability distribution of the random variable $p(x)$. Generally, entropy is a quantity that depicts how much information an event or random variable contains.

***Definition:*** Let X be a discrete random variable with a probability mass function

$p(x) = \Pr\{X = x\}, x \in X$. The entropy H(X) of a variable X is defined by;

$$H(X) = -\sum_{x \in X} p(x) \log p(x), \tag{3.1}$$

The unit of information entropy is measured in "bits" or "not" depends on the base of the logarithmic function being two or $e$ respectively.

## 3.2 Joint Entropy

In the case of two different random variables, the entropy of these values can be calculated in a similar manner to that of calculating entropy a random variable. This term, called joint entropy, gives the overall uncertainty of these two random variables.

***Definition:*** The joint entropy H(X,Y) of a pair of discrete random variables with a joint distribution $p(x, y)$ is defined as:

$$H(X, Y) = -\sum_{x,y} p(x, y) \log p(x, y). \tag{3.2}$$

## 3.3 Conditional Entropy

Conditional entropy is a measure of the amount of information required to determine the outcome of a random variable Y given the value of the random variable of X.

***Definition:*** The conditional entropy of Y given X, H(Y|X), is defined as;

$$H(Y|X) = -\sum_{x,y} p(x)H(Y|X = x), \qquad (3.3)$$

$$= -\sum_x p(x) \sum_y p(y|x) \log p(y|x) , \qquad (3.4)$$

$$= -\sum_{x,y} p(x,y) \log p(y|x). \qquad (3.5)$$

From the definitions of joint entropy and conditional entropy, it can be seen that the entropy of two random variables is the summation of the marginal entropy of one and the conditional entropy of the other. This specification is called the chain rule of information entropy, and the proposed theorem is explained below. For the proof of this theorem, the reader is referred to the sourcebook mentioned above.

***Theorem 3.1:*** $H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y).$ (3.6)

## 3.4 Kullback-Leibler Divergence

Kullback-Leibler divergence, also called relative entropy, is the measure of the distance between two different distributions over the same random variable. This quantity depicts how different these two distributions are.

***Definition:*** The Kullback-Leibler distance between two probability distributions $p(x)$ and $q(x)$ is defined as

$$D(p||q) = \sum_x p(x) log \frac{p(x)}{q(x)}. \qquad (3.7)$$

The relative entropy is always non-negative and is zero if the only p = q. Moreover, it is an asymmetric function, so it does not give an accurate measure of the distance between distributions. Thus, it is simpler to think of "relative entropy" rather than the distance between distributions.

## 3.5 Mutual Information

Mutual information describes the amount of information that one random variable contains about another.

***Definition:*** The mutual information of two random variables, $I(X;Y)$, is defined as

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}, \qquad (3.8)$$

This is the reduction in the uncertainty of one random variable due to the knowledge one has of the other. Thus, mutual information can be calculated by entropy quantities as

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X), \quad (3.9)$$

From equation 3.9, it can be seen that the notion of mutual information is symmetric. Thus, X gives as much information about Y as Y gives about X. The relationship between these quantities is exhibited in Figure 3.1. From this figure, it can be understood that mutual information is the intersection of the two random variables' information.

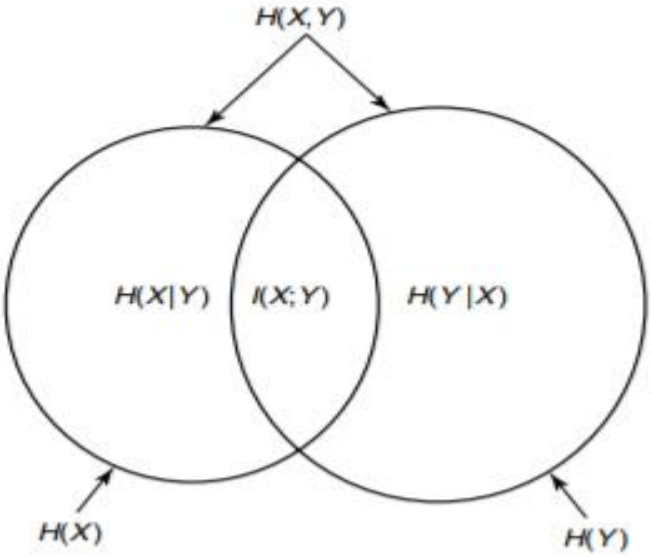


*Figure 3.1 Relationship between entropy and mutual information*

*(Source: Thomas and Cover, 1991)*

## 3.6 Data Processing Inequalities

Data processing inequality is an information-theoretic concept that states that the no physical processing of data can improve its information content. It can be said that the information that a variable contains cannot be increased by post-processing.

***Definition:*** Let random variables X, Y, Z form a Markov Chain in that order, as denoted X→Y→Z, and the conditional distribution of Z depends only on Y and is conditionally independent of X. In this setting X, Y and Z form a Markov chain. This resulted in a theorem that no processing of Y can increase the information Y contains about X.

This is formulated as

***Theorem 3.2***: $I(X;Y) \geq I(X;Z)$ $(Data - processing\ inequality)$

## 3.7 Fano's Inequality

Fano's inequality is an information-theoretic lemma that gives the relationship between categorisation error and average information lost in a channel. It is utilised to find a lower bound on the error probability of any decoder.

***Definition:*** Let X be a random variable with finite outcomes in $\theta$. Let $\hat{X} = g(Y)$ be the predicted value of X for a deterministic function. Then, Fano's inequality can be defined as

$$p_e \equiv p(\hat{X} \neq X) \geq \frac{H(X|Y) - 1}{\log|\theta|}, \qquad (3.10)$$

where $p_e$ is the generalisation error of function and $H(X|Y)$ is the conditional entropy.

# CHAPTER 4 Information Theory for Explaining Deep Learning

Deep learning models have recently begun to show excellent performance for real-world data challenges. This success has attracted people to apply these models to various tasks. However, basic questions such as the optimal architecture of the model, the number of required layers, or the number of neurons have not yet been adequately answered (Tishby and Zavlavsky, 2015). Thus, many researchers are also trying to understand the theory behind these models. Utilising Information Theory to explain the nature of deep neural networks has attracted the attention of many researchers in recent years.

The literature mentioned in section 2.3 has provided us with an intuitive knowledge of how information theory can be used to explain deep learning. Many new studies are being conducted on these necessary studies. In this section, reviews of the essential research that forms the basis of this thesis will be provided.

## 4.1 Information Bottleneck Principle and Deep Learning

The Information Bottleneck Principle was introduced by Tishby et al. (1999) to extract the relevant information that a random input variable X contains about random variable Y. This principle is intended to find the optimal representation of X that contains the maximum relevant information about Y, and that is compressed maximally by discarding all non-useful information; this process is used to find the best trade-off between accuracy and compression. For general compression process, the relevant part of X concerning Y is denoted by T. This compression has the assumed form of a Markov Chain $Y \to X \to T$ and minimises the mutual information $I(X;T)$ under the constraint on $I(T;Y)$ due to the data processing inequality defined by Theorem 3.2. The information bottleneck can be seen as a rate-distortion measure which is defined as

$$D_{IB} = E[d_{IB}(X,T)] = I(X;Y|T), \tag{4.1}$$

where D is the Kullback-Leibler divergence, which is defined in Section 3.6.

The idea above was adopted to analyse the learning behaviour of neural networks by Tishby and co-workers' recent papers (Tishby and Zavlavsky, 2015; Shwartz-Ziv and Tishby, 2017). In this context, deep learning is taken as a question that is representative of a learning problem. The goal of a supervised learning algorithm is to capture as much relevant information from the input variables about the output variables during training as possible. So, it can be said that the layers of the deep neural network form a Markov Chain, and the overall model can be seen as an encoder/decoder structure (see Figure 4.1). The Markov Chain feature is due to every hidden layer, only having access to the output of previous layers as its input. Hence, data processing inequalities can be used in neural network layers. This is defined in mathematical terms as

$$I(X;Y) \geq I(T_1;Y) \geq I(T_2;Y) \geq \cdots \geq I(T_k;Y) \geq I\left(\hat{Y};Y\right), \tag{4.2}$$

$$H(X) \geq I(X;T_1) \geq I(X;T_2 \geq \cdots \geq I(X;T_k) \geq I\left(X;\hat{Y}\right). \tag{4.3}$$

These formulations imply that the mutual information between the variables decreases when moving across the network towards the output layer.

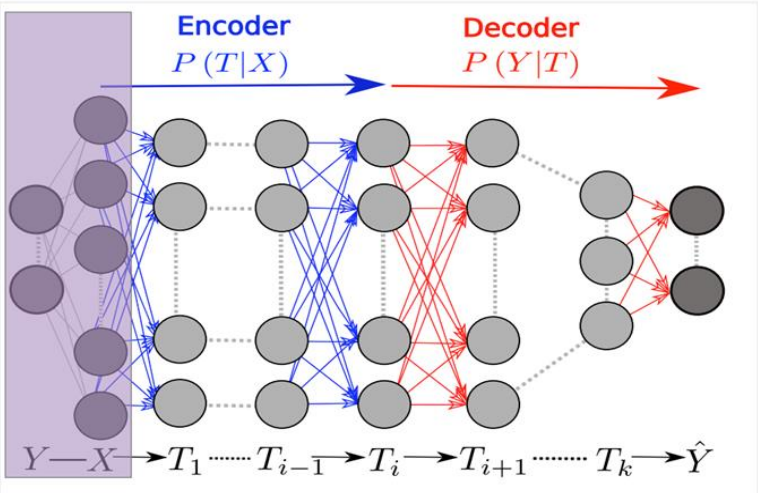


*Figure 4. 1 A Deep Neural Network Structure visualised as an encoder and decoder (Schwartz-Ziv and Tishby, 2015)*

Hence, they analysed deep neural networks by measuring the information quantities in each layer's representations, X, T, Y, where X is the input variable, Y is the label and T is the hidden layers' variables on the Information Plane (IP). The goal of the network then is to optimise the Information Bottleneck (IB) trade-off between compression and prediction, successively, for each layer. So, with the implementation of the IB method to deep neural networks, they optimally learn to extract the most efficient informative features with the most compact architecture (i.e., the optimum number of layers and units). Besides, they are supposed that train a model with the stochastic gradient descent optimisation through two distinct phases: the fitting phase, and the compression phase. Initially, the model gets into the fitting phase where $I(X;T)$ and $I(T;Y)$ increase together, along with the training iterations. Later, the model goes into the compression phase where both $I(X;T)$ and $I(T;Y)$ decrease. It is suggested in the paper that this compression phase is a good indicator of the excellent generalisability of deep neural networks.

## 4.2 An Information-Theoretic View of Learning of Artificial Neural Networks

In light of Tishby's paper, Balda et al. (2019) also attempted to use information-theoretic quantities to reveal typical learning patterns of neural networks in their recent study. They investigated the mutual information and conditional entropy of the input, output, and true labels. What makes mutual information so interesting is that, unlike correlation, it can pick up non-linear dependencies between variables. Given two random variables X and Y, it looks at the divergence of their probability distribution $p(x,y)$ from $p(x)p(y)$ to determine how dependent - or independent - they are.

Then, from Fano's inequality, an upper bound is derived for the conditional entropy of the estimated labels given the true ones in terms of the error probability. The upper bound is defined as follows:

***Definition:*** *For a neural network denoted as $g_\theta$ , and the output denoted as $\hat{y} = g_\theta(x)$ . The conditional entropy quantities upper bounded by the generalisation error $R(g_\theta)$.*

$$\max\{H(Y|\hat{Y}), H(\hat{Y}|Y)\} \leq \varphi(R(g_\theta)), \tag{4.4}$$

For experimentation, they used three different datasets: the MNIST handwritten image dataset, the Cifar-10 dataset, and the Spirals dataset. For each dataset, they used a different model with different hyperparameter settings. It can be seen from Figure 4.2 that regardless of the activation function or dataset used, the learning process consists of two phases. In the first phase, $I(X;\widehat{Y})$ and $H(\hat{Y}|Y)$ increase together. The increase means that the neural network learns mostly about input distributions. This behaviour continues to a specific value of mutual information, then the second phase starts (this stage is referred to as non-discriminative). In the second phase, $H(\hat{Y}|Y)$ is minimised through the epoch. This decrease indicates that the neural network learns information about true labels (this stage is referred to as the discriminative). It can also be seen from Figure 4.2 that conditional entropies approach the bounds given by Equation 4.4.

The figures show that in well-trained networks, conditional entropy and expected errors approach their theoretical limits. Hence, it is suggested that mutual information, conditional entropy and expected error can serve as a method for verifying the correct learning of deep neural networks.

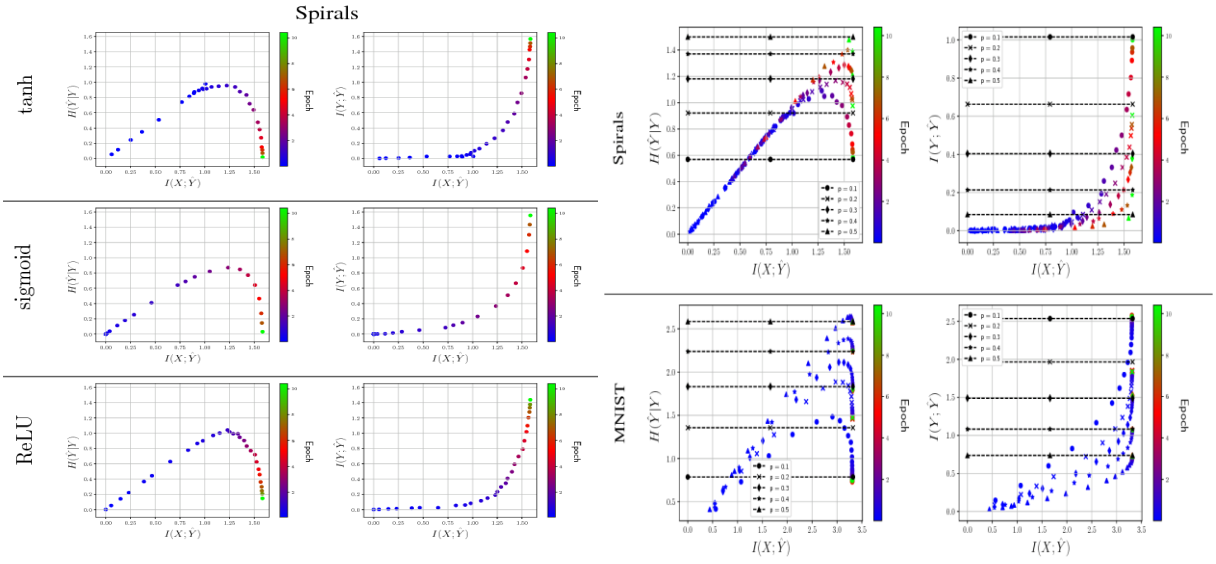


*Figure 4.2 Information theoretical quantities during training*

# CHAPTER 5 Understanding Convolutional Neural Networks via Information Theory

Convolutional neural networks (CNNs) have been some of the most influential innovations in the field of computer vision. In contrast, whilst there are efforts to understand deep learning, the studies working on convolutional neural networks are still insufficient. Within the scope of this thesis, the learning process of CNNs will be analysed for image classification tasks.

In this chapter, a brief introduction is first given for CNNs. Then, the methods followed during the experiments are explained. Last, the learning process for CNNs is investigated with information-theoretic quantities in the vision of the studies presented in the previous chapter.

## 5.1 Convolutional Neural Networks

CNNs are used to process data that has a grid-like topology. This model has been beneficial for working on two-dimensional image data. The model was introduced and proposed by LeCun and Bengio (1995). Goodfellow, Bengio, and Courville stated in their book that these models could be seen as a successful application of studies into the brain to those of machine learning implementations (2017). CNNs are particularly useful at finding patterns in images to recognise objects for computer vision tasks. CNNs are beneficial due to a number of breakthrough features for processing data: they eliminate the need for manual feature detection due to the ability to learn directly from an image.

Generally, CNNs are comprised of four principle layer types: the convolution layer, rectified linear unit layer, pooling layer, and fully connected layer. An example of a CNN architecture is illustrated in Figure 5.1.

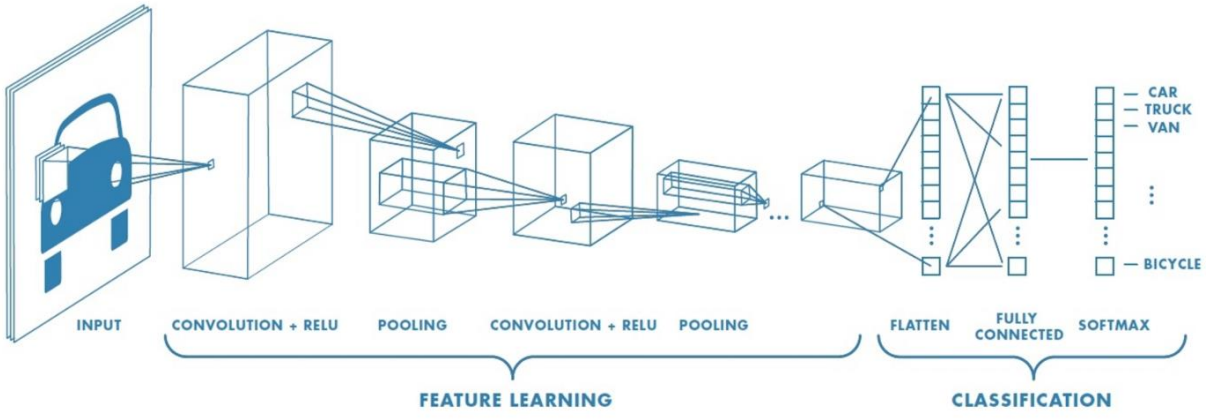


*Figure 5.1 Basic CNN architecture for image classification tasks (Mathworks, 2018)*

The first layer that takes images as input is the convolutional layer. The name comes from its operation name, “convolve”. This layer convolves the input with predefined kernels (filters) and creates feature maps of the input. Filters slide over the input matrix and every time

multiplication of image region and filter being output. An example of a convolution operation can be seen in Figure 5.2. ReLu layers are usually implemented on convolution layers that mapped negative values of feature maps to zero and make training faster and more effective. Also, the ReLu layer ensures the non-linearity of the network.

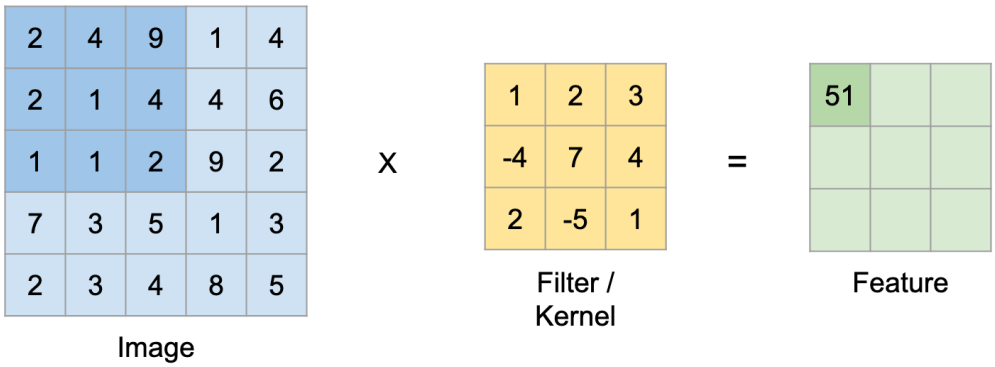


*Figure 5.2 Illustration of the convolution operation*

Pooling is a process of downsampling the feature map on CNN. The pooling layer is a new layer that is added right after the convolutional layers after ReLu (activation) has been applied. This layer extracts a particular value from a set of values. Two of the pooling layers' types are the most well-known: the Max Pooling Layer, which takes the maximum value of the predefined matrix, and the Average Pooling Layer, which takes the average of the defined matrix values. This layer is useful for improving computation time by reducing the size of the output matrix. An example of a pooling operation is shown in Figure 5.3.

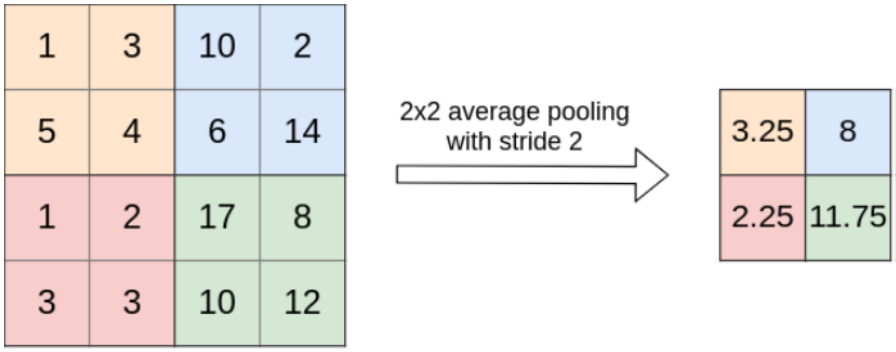


*Figure 5.3 An example of an average pooling layer operation*

In the end, fully connected layers are added to the network to make the classification. In this layer, all neurons are connected like other neural networks. The matrix-shaped outputs are flattened before the fully connected layer to make it convenient to feed the layer. This layer computes the class probability scores and gives N-dimensional vectors that belong to the N number of classes.

## 5.2 Experimentation Methodology

The Cifar-10 and MNIST datasets are have been selected and trained for two different CNN models in this thesis. The development of the test setup consists of five main stages:

1) The dataset to be used for analysis is loaded.

2) A convolutional neural network model is designed and compiled.

3) While the compiled model is being trained, the network weights are saved after every epoch.

4) The models selected for visualising are rerun and the outputs of hidden layers extracted and assigned to the predefined lists.

5) Finally, from these outputs, information quantities are calculated and plotted.

All experiments were carried out in the Keras Library with a Tensorflow backend. The high-level flowchart of the experimental process can be seen in Figure 5.4.

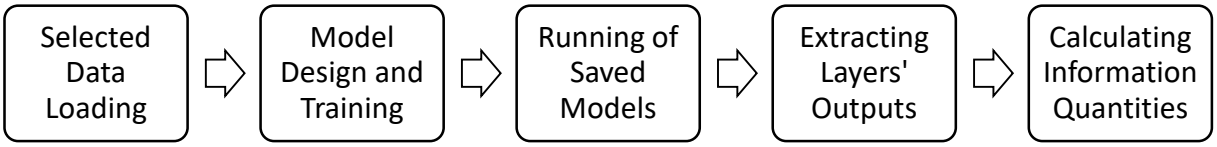


*Figure **Hata! Burada görünmesini istediğiniz metne 0 uygulamak için Giriş sekmesini kullanın.**4 Development pipeline of the proposed method*

## 5.3 Experimentation with MNIST Dataset

Firstly, the training process for the convolutional neural network for MNIST classification is analysed for the proposed testing method.

### 5.3.1: Dataset

The MNIST dataset is a modified subset of the National Institute of Standards and Technology database. This dataset contains 60,000 grey-scale images of handwritten digits represented by 28 x 28 pixels. The task is to classify a given picture of a handwritten digit into one of 10 classes representing integer values from 0 to 9, inclusive. Key specifications and examples of the dataset can be seen in Table 5.1 and Figure 5.5, respectively.

*Table 5.1 MNIST Dataset Specifications*

| | |
|---|---|
| Data Type | Image data |
| Size of each image | 28 x 28 image pixel values |
| Number of channels | One channel grey-scale |
| Number of classes | Ten classes |
| Number of the training set | 50,000 |

| | |
|---|---|
| Number of the test set | 10,000 |

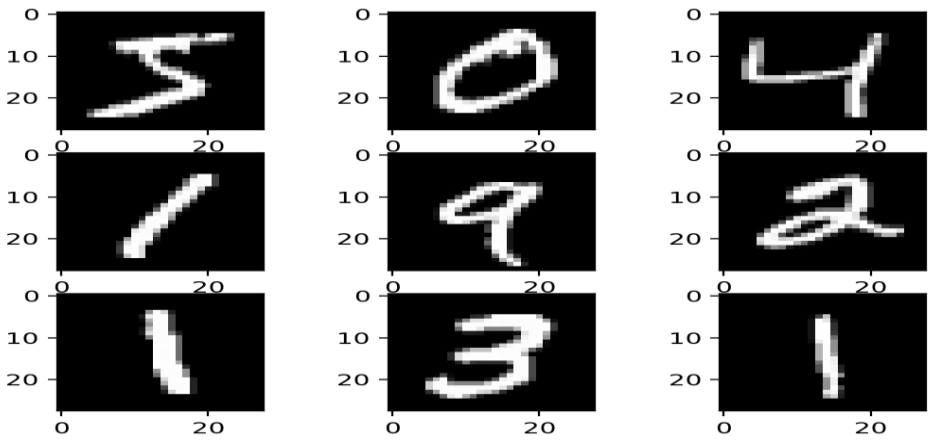

*Figure 5.5 Examples of digits in the MNIST dataset*

### 5.3.2: Model Design and Training

To classify the MNIST dataset, a nine-layer structure convolutional neural network model is designed. The model is based on a well-known VGG (Simonyan and Zisserman, 2014) model. This model was chosen due to its good performance with the Image Net challenge. The modular structure of the model makes it easy to implement. The architecture involves two convolution blocks and one fully connected block. The convolution block consists of two convolution layers with small 3 x 3 kernels followed by a Max Pooling layer. Also, each convolution layer was implemented with a ReLU (Nair and Hinton, 2010) activation function. After two convolution blocks, a fully connected layer is added to the model to make the classification. The classification is achieved via the softmax function. A summary of the designed model can be seen in Figure 5.6.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 28, 28, 32)        320
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 26, 26, 32)        9248
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 13, 13, 32)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 13, 13, 64)        18496
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 11, 11, 64)        36928
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 5, 5, 64)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 1600)              0
_________________________________________________________________
dense_1 (Dense)              (None, 512)               819712
_________________________________________________________________
dense_2 (Dense)              (None, 10)                5130
=================================================================
Total params: 889,834
Trainable params: 889,834
Non-trainable params: 0
```

*Figure 5.6 A summary of designed Model 1*

For optimisation, the “rmsprop” algorithm was selected with a categorical cross-entropy loss function to train the network. Then, the model was compiled and run for one hundred epochs. The model’s learning curves, as produced during the training period, are plotted in Figure 5.7. From the figure, it can be seen that the model converged around 15-20 epochs. After that, it starts to over fit the data. The accuracy and loss table of the best epoch of the trained model is given in Table 5.2. Each epoch’s weight is saved during the training.

*Table 5.2 Accuracy and loss values for Model 1’s best epoch*

| | Training Set | Validation Set |
|---|---|---|
| Accuracy | 0.9990 | 0.9936 |
| Loss | 0.0036 | 0.0268 |

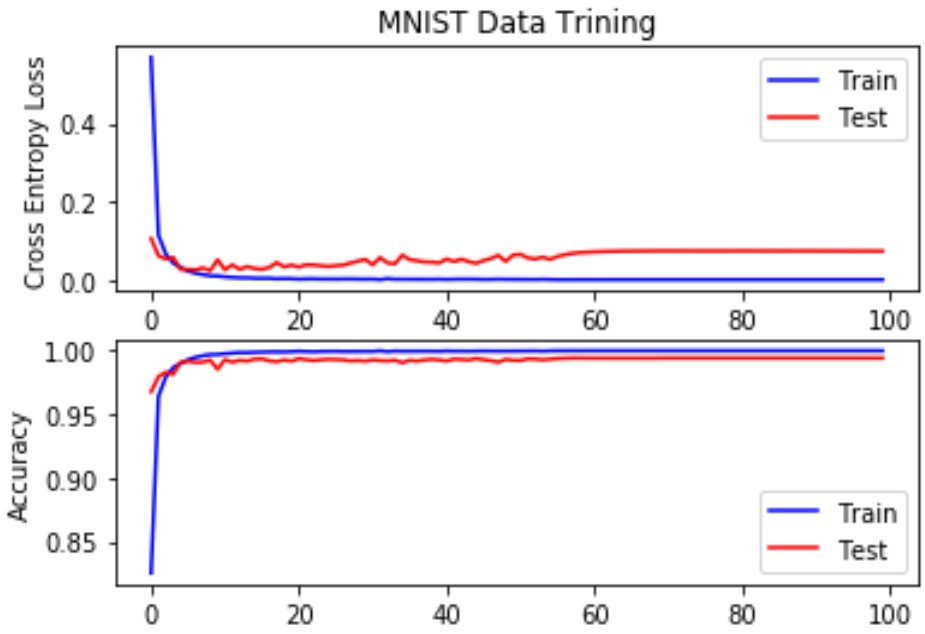


*Figure 5.7 Learning curves for designed Model 1*

## 5.4 Experimentation with the Cifar-10 Dataset

### 5.4.1: Dataset

Cifar-10 is another popular dataset that is commonly used for computer vision tasks. This dataset consists of 60,000 raw images chosen from 80 million small-sized photos. In the dataset, ten classes represent the contained objects. These objects are listed as aeroplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. Each image in the dataset represents one class of objects with a 32 x 32 image pixel matrix. In this experiment, 50,000 samples were used for training and 10,000 for testing. Some important features and examples of the dataset are shown in Table 5.3 and Figure 5.8, respectively.

*Table 5.3 CIFAR-10 Dataset Specifications*

| | |
|---|---|
| Data Type | Image data |
| Size of each image | 32 x 32 image pixel values |
| Number of channels | Three channels RGB |
| Number of classes | Ten classes |
| Number of the training set | 50.000 |
| Number of the test set | 10.000 |

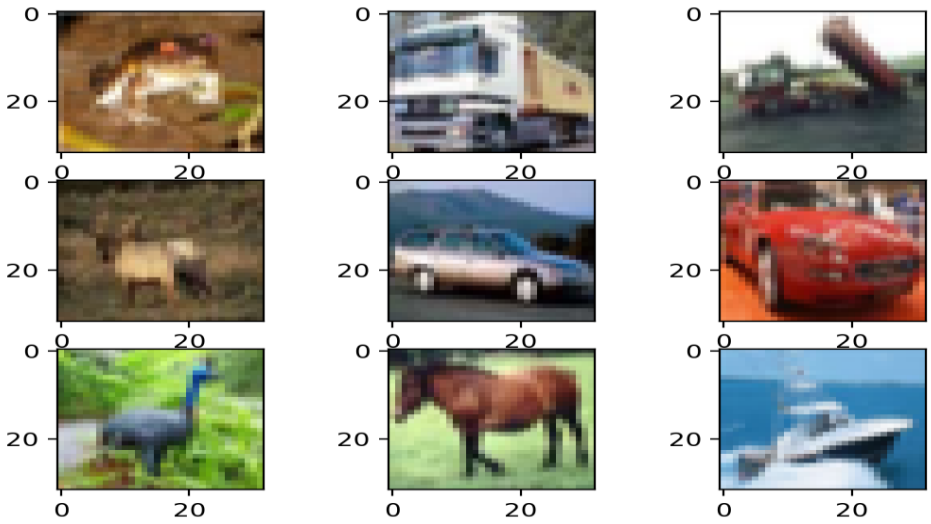

*Figure 5.8 Cifar-10 dataset examples*

### 5.4.2: Model Design and Training

A new model for training the Cifar-10 dataset was designed. This new model otherwise has a similar structure to the previous one but, by contrast, the ReLU activation functions are not implemented with a convolution layer but are added separately immediately after every convolutional layer. This has done due to realise the effects of non-linear activations on learning. The overall model is composed of two convolutional blocks, followed by a fully connected layers. The classification is performed in the fully connected layer via the softmax function.  A summary of the designed model can be seen in Figure 5.9.

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_5 (Conv2D)            (None, 32, 32, 32)        896
_________________________________________________________________
activation_5 (Activation)    (None, 32, 32, 32)        0
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 30, 30, 32)        9248
_________________________________________________________________
activation_6 (Activation)    (None, 30, 30, 32)        0
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 15, 15, 32)        0
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 15, 15, 64)        18496
_________________________________________________________________
activation_7 (Activation)    (None, 15, 15, 64)        0
_________________________________________________________________
conv2d_8 (Conv2D)            (None, 13, 13, 64)        36928
_________________________________________________________________
activation_8 (Activation)    (None, 13, 13, 64)        0
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 6, 6, 64)          0
_________________________________________________________________
dropout_2 (Dropout)          (None, 6, 6, 64)          0
_________________________________________________________________
flatten_2 (Flatten)          (None, 2304)              0
_________________________________________________________________
dense_2 (Dense)              (None, 10)                23050
=================================================================
Total params: 88,618
Trainable params: 88,618
Non-trainable params: 0
```

*Figure 5.9 Summary of designed Model 2*

The model optimised with the Stochastic gradient descent algorithm with learning rate $lr = 0.01$. SGD was selected due to its superior generalisation performance (Wilson, 2017). Also, the accuracy of this network was calculated using the categorical cross-entropy loss function due to it being beneficial in multiclass classification. The network was compiled and run for two hundred epochs. The learning curves of the model as produced during training can be seen in Figure 5.10. From the figure, it can be seen that the network had converged at around the 75$^{th}$ epoch; after that, it began to over fit the data. The accuracy and loss table of the best epoch for the trained model is given in Table 5.4.

*Table 5.4 Accuracy and the loss values of Model 2's best epoch*

| | Training Set | Validation Set |
|---|---|---|
| Accuracy | 0.5465 | 0.5568 |
| Loss | 1.28 | 1.2598 |

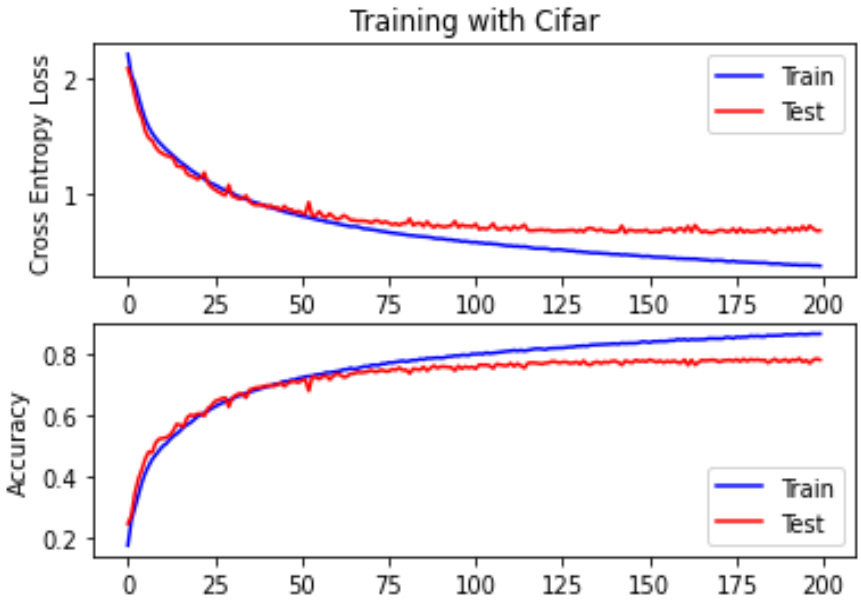


*Figure 5.10 Learning curves for designed Model 2*

## 5.5 Calculation of Information Quantities

In this thesis, information-theoretic quantities are used to understand the behaviour of convolutional neural networks. to calculate these quantities, all representations of the input samples and layers' outputs are considered to be random variables. Then, for each experiment, the mutual Information $I(X;T)$ and entropy $H(X)$ of the investigated layers were calculated empirically by Shannon's formulations.

The computing algorithm can be defined as:

Step 1) Representations, i.e., input image matrices or output tensors, the of investigated layer output and input samples, are converted to a one-dimensional array.

Step 2) Each unique value in the previously generated arrays and the frequencies of those values are counted. Then, from these counts, the occurrence probability of each value is calculated.

Step 3) From the probabilities obtained in Step 2, the Shannon's entropy and mutual information of variables is computed.

To make it intuitively understandable, the calculations of two samples in the MNIST and Cifar-10 datasets are shown as examples.

### 5.5.1 Information Quantities of MNIST Images

Firstly, the normalised histograms of the two randomly selected images from the MNIST dataset are plotted in Figures 5.11 and 5.12. The histograms show the probability of each pixel value belonging to the sample images. Because of the grey-scale nature of the dataset, it can be seen that pixel values are concentrated at 0 and 1. This indicates that in the MNIST images,

most bits are either brighter (represented by pixel value of 255) or darker (represented by a pixel value of 0).

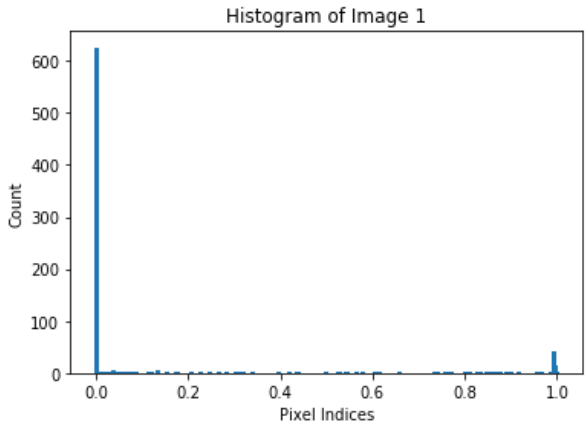


*Figure 5.11 The histogram of selected Image 1*

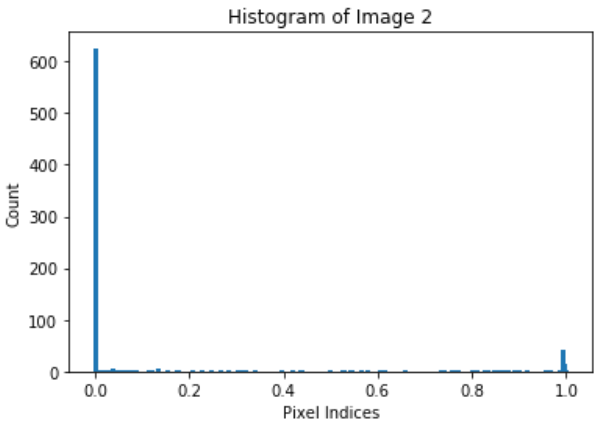


*Figure 5.12 The histogram of selected Image 2*

Then, from the probability values of the pixel information, theoretical quantities such as entropy, mutual information, and conditional entropy are calculated empirically. The results of these calculations are shown in Table 5.5.

*Table 5.5 Calculated information quantities for sample MNIST images*

| | |
|---|---|
| The entropy of Image 1, H(X): | 1.7383348383366406 |
| The entropy of Image 2, H(Y): | 1.5905442413028457 |
| Mutual Information of Images, I(X;Y): | 1.5400207334952398 |
| Conditional Entropy of Images, H(X\|Y) : | 0.19831410484140077 |
| Conditional Entropy of Images, H(Y\|X) : | 1.7888583461442464 |

It can be seen from histograms of the selected images that the probability distributions are approximately the same. Further, it can be inferred that the amount of uncertainty is relatively low due to the black and white structure. Thus, the amount of entropy was expected to be low. The fact that entropy values meet expectations gives some assurance about the appropriateness of the calculation method. Moreover, it can be seen that mutual information is the subtraction of conditional entropy from marginal entropy, which demonstrates the compatibility of the method used with Equation (3.9).

### 5.5.2 Information Quantities of Cifar-10 Images

The same procedure as above was again implemented for the Cifar-10 dataset. The histograms of two randomly selected sample images are shown in Figure 5.13 and Figure 5.14. From the figures, it can be seen that the images that the Cifar-10 dataset contains are somewhat different in nature to the MNIST samples. This is the result of the RGB channel representations of the images. RGB representation allows the use of all pixel values and allows the image to be represented with colours. Thus, the probability distribution of the pixel values is more invariant than the MNIST images.

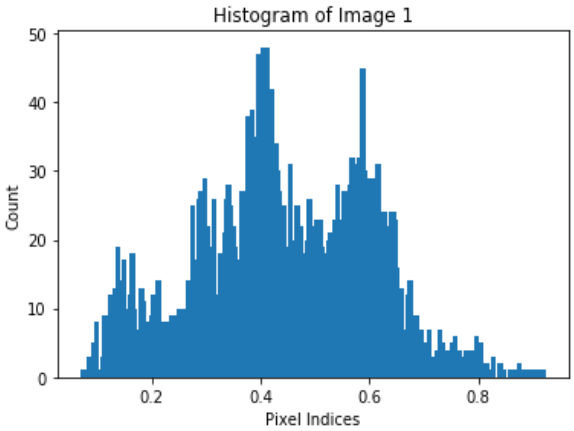


*Figure 5.23 Histogram of selected Image 1*

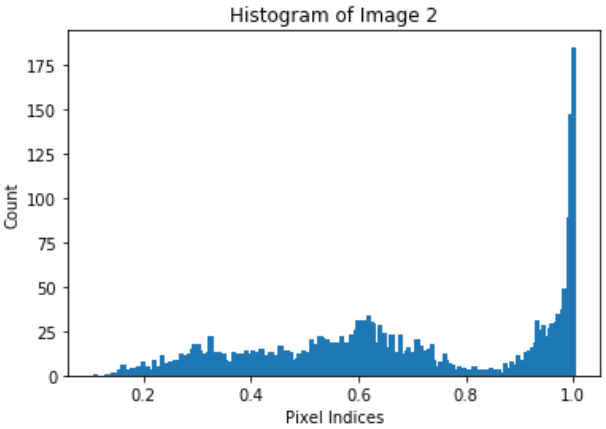


*Figure 5.14 Histogram of selected image 2*

In the second stage, the information-theoretic quantities of the selected two images from Cifar-10 dataset were calculated. These results can be seen in Table 5.6. The entropy values calculated are approximately the same for each of the images. The definition of entropy indicates that the entropy of a variable returns the minimum number of bits required to represent that variable. Generally, an 8-bit representation is used for images. The computed entropy values being approximately equal to 8 indicates the reliability of the empirical method used. In addition, the mathematical calculation of mutual information from entropy and conditional entropy also conform to Equation (3.9).

*Table 5.6 Calculated information quantities for the sample Cifar-10 images*

| | |
|---|---|
| The entropy of Image 1, H(X): | 7.250533790510763 |
| The entropy of Image 2, H(Y): | 7.184727136961712 |
| Mutual Information of Images, I(X;Y): | 6.958058843859936 |
| Conditional Entropy of Images, H(X\|Y) : | 0.2924749466508274 |
| Conditional Entropy of Images, H(Y\|X) : | 0.22666829310177583 |

# CHAPTER 6 Results and Discussion

This chapter contains the results of the proposed methodology experimentation conducted to answer the research questions:

Research Question 1: What is the effect of hidden layers on the learning of a convolutional neural network?

Research Question 2: How does training affect the learning behaviour of a convolutional neural network?

These research questions were answered through the experimentation. The learning behaviour of the convolutional neural networks was investigated via information theory during the training of the classification tasks. During the experimentation, mutual Information, $I(X;T)$, between the input and output of the layers was calculated. As mentioned in Chapter 5, the layers in the network were considered to be a single variable and the mutual information between each layer with the input and labels were calculated from those variables.

In the remainder of this chapter, the results and discussions will be given. In each of these sections, studies into each particular research question and the discussion of their results will be given.

## 6.1 Effect of Hidden Layers on Learning

To understand the effect of hidden layers on learning, two designed models were used with two different datasets, as explained in Chapter 5. The results obtained for the MNIST and Cifar-10 experimentation setups are illustrated in Figure 6.1-a and Figure 6.1-b, respectively. The mutual information, $I(X;T)$, between the input and the output of the layers are calculated and plotted in these figures. In the context of study, the mutual information between input and output was calculated by considering the theorem $I(X;T) = H(T)$, where the variable $T$ is the deterministic functional of X. This assumption was made because it is known that deep learning models are deterministic (Wu, 2014). With these graphs, one can monitor how the information-theoretic quantities change with the hidden layers. The compliance of the calculated and plotted quantities with the data processing inequality theorem, as introduced in Chapter 3, will be analysed and discussed.

One can interpret from the results that convolutional neural networks do not follow a monotonic learning process. The influence of individual layers and the overall layer behaviour varies from model to model. Each layer has an increasing or decreasing impact on mutual information. In both figures, it is clear that dropout and flattening of layers has no apparent effect on learning. It was thus inferred that most learning takes place in the fully connected layer, which actually fits the definition of a convolutional neural network. In the convolutional neural network, convolutional blocks are used to extract features from the image. The fully connected layers are added to make the classification. The experimental results also confirm this situation.

The convolutional neural network was investigated with regard to whether it satisfied data processing inequality or otherwise. In the case of a feedforward neural network, the Markovian structure and data processing inequalities across layers are generally accepted (Schwartz-Ziv and Tishby, 2017; Balda, Behboodi and Mathar, 2018). In a previous study, it was stated that this can also be seen in convolutional neural networks, despite the calculation limits (Yu et al.,2020). However, in tests performed with the proposed approach in this thesis, there was no DPI between the layers for convolutional neural networks. Hence, the investigation of data processing inequality theory for convolutional neural networks could be conducted in future related studies.

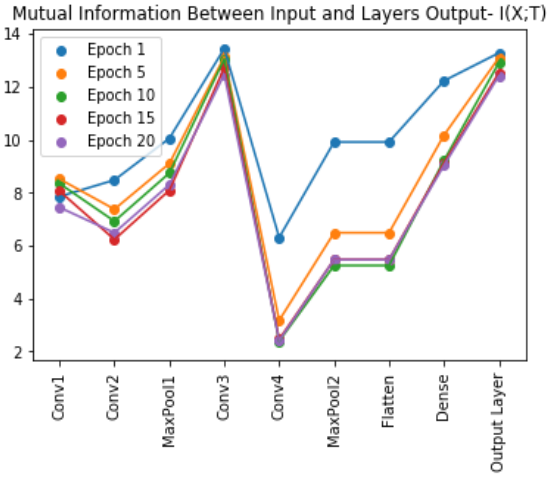


*Figure 6.1-a*

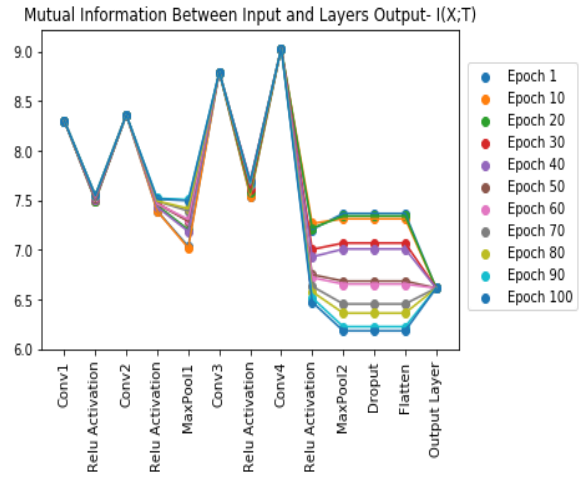


*Figure 6.1-b*

*Figure 6.1- Mutual information between input and output of layers, I(X;T). These two graphs reflect the results of the situation where 1000 MNIST images are sent to the network.*

*Figure 6.1-b Entropy and mutual Information along the layers are plotted in this figure. The figure is plotted for 1000 Cifar-10 images as the input size.*

## 6.2 Effect of Training on Learning Process

Mutual information quantities between the input and output $I(X;T)$ and between the output and true labels $I(T,Y)$ were observed during the training of the two models. These quantities were investigated to identify training steps with regard to the amount of mutual information. This observation will offer a degree of insight about learning of the convolutional neural network. The results of this experimentation are visualised in Figure 6.2 and Figure 6.3.

In Figure 6.2, it can be seen that the mutual information shows a decreasing trend during the training epochs. Although there is a slow decrease until the network converges, there is a sharp decrease at the onset of overfitting. The overfitting behaviour of the network can be seen from the accuracy graph of the network, which was presented in Chapter 5. It can be inferred that the change of an information-theoretic quantity is parallel to the learning process of the network. As a result, the optimum training epoch number can be determined by observing the information-theoretic quantities.

However, it can be seen from Figure 6.3 that mutual Information quantities, $I(X;T)\ and\ I(T;Y)$ do not change during the training for the Cifar-10 dataset and model. At first sight, this might be considered a calculation problem. The algorithms and results of every step are checked individually and detailed. The outputs and the entropy of the outputs are calculated for every epoch. It was then realised that the problem is not one of calculating mutual information. The results show that the network outputs have the same uncertainty during the entirety of the training. It was observed that although the probabilities of the class vector changed throughout the training, the amount of uncertainty (entropy) remained the same. This issue or problem is unclear, and one of the open cases that need to be investigated in future studies.

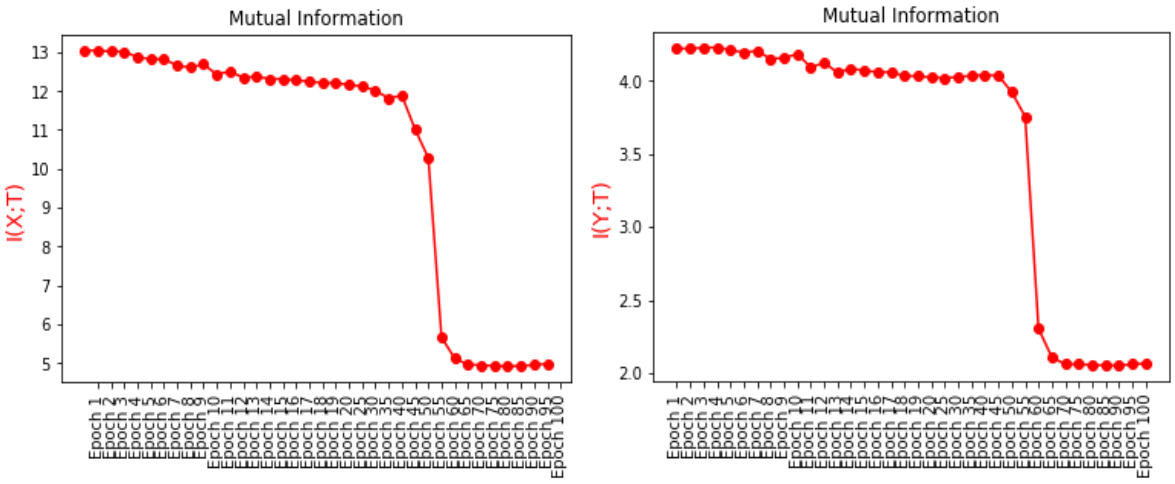


*Figure 6.2 Mutual information between input and output of model $I(X;T)$ and mutual information between the output of the model and true labels $I(Y;T)$ as plotted during the training epochs.*

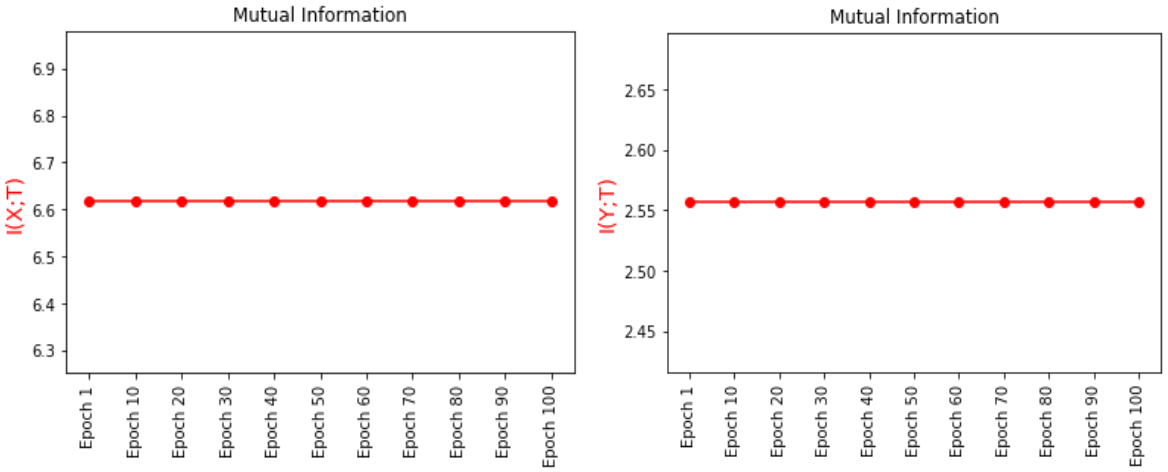


*Figure 6.3 The mutual information quantities I(X;T), and I(Y;T) are plotted for Cifar-10 images sent to the network.*

# CHAPTER 7 Conclusion and Future Work Recommendations

## 7.1 Introduction

The main aim of this dissertation was to understand the learning process of convolutional neural networks via information theory. During the research, information theoretic quantities were utilised to explain the CNNs according to the research questions. As a result of the study, the effects of hidden layers and training on learning could be stated. In the thesis, a new method for calculating information theoretic quantities of neural network parameters were proposed. During the study, numeric experiments were carried out using the suggested method.

The first research question posed in the Introduction was "*What are the effects of hidden layers on learning of a convolutional neural network?".* This question was investigated by observing mutual information between the input and layer outputs. The results showed that each individual layer has a different effect on learning. The layers that need to be added to a neural network to ensure the desired learning level can be determined with the help of information theoretic quantities.

The second research question considered was "*How does training affect the learning process of a convolutional neural network?".* The results of experimentation showed that the information theoretic quantities graph was parallel to the training accuracy graph. It can be thus interpreted that the optimum number of training epochs and other parameters can be determined with information theory. The results of all the experiments performed and the literature reviewed are signs that information theory in convolutional neural networks can be used.

## 7.2 Limitations and Recommendations

Although the results indicated the precise implementation of the thesis methodology, there are a number of limitations to the proposed method. First, all the information quantities mentioned in this paper are calculated by taking all variables in one-dimensional vectors, i.e., the input images or output of any layer is first converted to a single vector before entropy or mutual information of the variables calculated. Although this calculation is straightforward, it results in the spatial relationships in the image-related data being ignored. Therefore, the question remains as to the reliability of the information theoretic estimation that is feasible within a tensor structure. In future studies, developing a calculation method by considering this spatial relation will give more reliable results.

Other useful work that could be attempted can be listed as follows:

1) By giving different numbers of inputs to the training model, the effect of the number of training data on learning can be seen with the same approach.

2) By using different optimisation functions for the model during the learning process, the effect of the optimisation function on learning can be realised.

3) In addition, the accuracy of the proposed method can be determined by trying different estimation methods for mutual information calculation.

## 7.3 Summary

In conclusion, in this thesis, a new method was proposed and experimentation conducted to analyse the learning process of convolutional neural networks. To explain the features of CNNs' learning processes, quantities from Information Theory were utilised. During the experiments, two different datasets, MNIST and CiFAR-10, were trained with two different models in order to ensure the reliability of the methods used. Both model's results generally supported each other. The results of the numerical experiments revealed that information theory is an excellent tool with which to explain convolutional neural networks. Calculation limitations and related concerns are stated above and are given as recommendations for proper resolution in future studies.

# Bibliography


Adadi, A. and Berrada, M. (2018). Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access*, 6, pp.52138–52160.

Bae, I., Moon, J. and Kim, S. (2019). Driving Preference Metric-Aware Control for Self-Driving Vehicles. *International Journal of Intelligent Engineering and Systems*, 12(6), pp.157–166.

Balda, E.R., Behboodi, A. and Mathar, R. (2018). An Information Theoretic View on Learning of Artificial Neural Networks. *2018 12th International Conference on Signal Processing and Communication Systems (ICSPCS)*.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R. and Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, [online] 58, pp.82–115. Available at: https://arxiv.org/pdf/1910.10045.pdf [Accessed 24 Aug. 2020].

Castelvecchi, D. (2016). Can we open the black box of AI? *Nature*, [online] 538(7623), pp.20–23. Available at: https://www.nature.com/articles/doi:10.1038/538020a [Accessed 24 Aug. 2020].

Chen, X.-W. and Lin, X. (2014). Big Data Deep Learning: Challenges and Perspectives. *IEEE Access*, 2, pp.514–525.

Cho, K., Merrienboer, van, Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. [online] arXiv.org. Available at: https://arxiv.org/abs/1406.1078 [Accessed 24 Aug. 2020].

Cover, T.M. and Thomas, J.A. (1991). *Elements of information theory*. New York: Wiley, New York.

De Mauro, A., Greco, M. and Grimaldi, M. (2016). A formal definition of Big Data based on its essential features. *Library Review*, 65(3), pp.122–135.

Doshi-Velez, F. and Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning. *arXiv:1702.08608 [cs, stat]*. [online] Available at: https://arxiv.org/abs/1702.08608 [Accessed 24 Aug. 2020].

Feng, Z. and Zhu, Y. (2016). A Survey on Trajectory Data Mining: Techniques and Applications. *IEEE Access*, 4, pp.2056–2067.

Gabrié, M., Manoel, A., Luneau, C., Barbier, J., Macris, N., Krzakala, F. and Zdeborová, L. (2019). Entropy and mutual information in models of deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, [online] 2019(12), p.124014. Available at: https://arxiv.org/abs/1805.09785 [Accessed 3 Jul. 2020].

Gandomi, A. and Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, [online] 35(2), pp.137–144. Available at: https://www.sciencedirect.com/science/article/pii/S0268401214001066 [Accessed 6 Mar. 2019].

Gartner. (n.d.). *Definition of Big Data - Gartner Information Technology Glossary*. [online] Available at: http://www.gartner.com/it-glossary/big-data/ [Accessed 9 Sep. 2020].

Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M. and Kagal, L. (2019). Explaining Explanations: An Overview of Interpretability of Machine Learning. *arXiv:1806.00069 [cs, stat]*. [online] Available at: https://arxiv.org/abs/1806.00069 [Accessed 24 Aug. 2020].

Gunning, D. and Aha, D. (2019). DARPA's Explainable Artificial Intelligence (XAI) Program. *AI Magazine*, [online] 40(2), pp.44–58. Available at: https://www.darpa.mil/attachments/XAIProgramUpdate.pdf [Accessed 24 Aug. 2020].

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The elements of statistical learning, second edition : data mining, inference, and prediction*. New York: Springer.

He, K., Zhang, X., Ren, S. and Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lipton, Z.C. (2018). The Mythos of Model Interpretability. *Queue*, 16(3), pp.31–57.

Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. and Byers, A.H. (2011). *Big data: The next frontier for innovation, competition, and productivity | McKinsey*. [online] www.mckinsey.com. Available at: https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/big-data-the-next-frontier-for-innovation# [Accessed 26 Aug. 2020].

Mathworks (2018). *Introducing Deep Learning with MATLAB*. [online] Available at: https://uk.mathworks.com/content/dam/mathworks/ebook/gated/80879v00_Deep_Learning_ebook.pdf [Accessed 2 Sep. 2020].

Murdoch, W.J., Singh, C., Kumbier, K., Abbasi-Asl, R. and Yu, B. (2019). Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, [online] 116(44), pp.22071–22080. Available at: https://www.pnas.org/content/116/44/22071 [Accessed 22 Aug. 2020].

Preece, A., Harborne, D., Braines, D., Tomsett, R. and Chakraborty, S. (2018). *Stakeholders in Explainable AI*. [online] arXiv.org. Available at: https://arxiv.org/abs/1810.00184 [Accessed 24 Aug. 2020].

Rao, T.R., Mitra, P., Bhatt, R. and Goswami, A. (2018). The big data system, components, tools, and technologies: a survey. *Knowledge and Information Systems*, 60(3), pp.1165–1245.

Roscher, R., Bohn, B., Duarte, M.F. and Garcke, J. (2020). Explainable Machine Learning for Scientific Insights and Discoveries. *IEEE Access*, 8, pp.42200–42216.

Sagiroglu, S. and Sinanc, D. (2013). Big data: A review. *2013 International Conference on Collaboration Technologies and Systems (CTS)*. [online] Available at: https://ieeexplore.ieee.org/document/6567202/ [Accessed 26 Aug. 2020].

Saxe, A.M., Bansal, Y., Dapello, J., Advani, M., Kolchinsky, A., Tracey, B.D. and Cox, D.D. (2018). *On the Information Bottleneck Theory of Deep Learning*. [online] openreview.net. Available at: https://openreview.net/forum?id=ry_WPG-A- [Accessed 28 Jun. 2020].

Shannon, C.E. (1948). A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(4), pp.623–656.

Shannon, C.E. (1949). Communication Theory of Secrecy Systems*. *Bell System Technical Journal*, 28(4), pp.656–715.

Shwartz-Ziv, R. and Tishby, N. (2017). *Opening the Black Box of Deep Neural Networks via Information*. [online] arXiv.org. Available at: https://arxiv.org/abs/1703.00810 [Accessed 28 Jun. 2020].

Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T. and Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587), pp.484–489.

Tishby, N., Pereira, F.C. and Bialek, W. (2000). The information bottleneck method. *arXiv:physics/0004057*. [online] Available at: https://arxiv.org/abs/physics/0004057 [Accessed 28 Jun. 2020].

Tishby, N., Pereira, F.C. and Bialek, W. (2000). The information bottleneck method. *arXiv:physics/0004057*. [online] Available at: https://arxiv.org/abs/physics/0004057 [Accessed 28 Jun. 2020].

Watson, H.J. (2019). Update Tutorial: Big Data Analytics: Concepts, Technology, and Applications. *Communications of the Association for Information Systems*, pp.364–379.

Wu, X., Zhu, X., Wu, G.-Q. and Ding, W. (2014). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 26(1), pp.97–107.

Wu, Z. (2014). *Deep Learning Deterministic Neural Networks*. [online] Available at: http://3dvision.princeton.edu/courses/COS598/2014sp/slides/lecture05_cnn/lecture05_cnn.pdf [Accessed 6 Sep. 2020].

Yu, S. and Principe, J.C. (2019). Understanding Autoencoders with Information Theoretic Concepts. *arXiv:1804.00057 [cs, math, stat]*. [online] Available at: https://arxiv.org/abs/1804.00057 [Accessed 3 Jul. 2020].

Yu, S., Wickstrøm, K., Jenssen, R. and Principe, J.C. (2020). Understanding Convolutional

Neural Networks with Information Theory: An Initial Exploration. *arXiv:1804.06537 [cs, math, stat]*. [online] Available at: https://arxiv.org/abs/1804.06537 [Accessed 10 Jul. 2020].

Zeiler, M.D. and Fergus, R. (2013). *Visualising and Understanding Convolutional Networks*. [online] arXiv.org. Available at: https://arxiv.org/abs/1311.2901 [Accessed 18 Jun. 2020].

Zhou, L., Pan, S., Wang, J. and Vasilakos, A.V. (2017). Machine learning on big data: Opportunities and challenges. *Neurocomputing*, 237, pp.350–361.