4-2023

An Analysis of Heterogeneous Quantization Schemes for Neural Networks

Anusha Holavanahali
axh7259@rit.edu

An Analysis of Heterogeneous Quantization Schemes for Neural Networks

Anusha Holavanahali

An Analysis of Heterogeneous Quantization Schemes for Neural Networks

Anusha Holavanahali

April 2023

A Thesis Submitted
in Partial Fulfillment
of the Requirements for the Degree of
Master of Science
in
Computer Engineering

RIT | Kate Gleason College of Engineering

Department of Computer Engineering

An Analysis of Heterogeneous Quantization Schemes for Neural Networks

Anusha Holavanahali

Committee Approval:

Cory Merkel *Advisor* Date
Assistant Professor, Department of Computer Engineering

Amlan Ganguly Date
Professor and Head, Department of Computer Engineering

Dongfang Liu Date
Assistant Professor, Department of Computer Engineering

Acknowledgments

I express my thanks to my advisor Dr.Cory Merkel for guiding me through this research. This work was made possible through a grant from the Air Force Research Lab.

Abstract

Quantization of neural network models is becoming a necessary step in deploying artificial intelligence (AI) at the edge. The quantization process reduces the precision of model parameters, thereby lowering memory and computational costs. However, in doing so, this process also limits the model's representational capacity, which can alter both its performance on nominal inputs (clean accuracy) as well as its robustness to adversarial attacks (adversarial accuracy). Few researchers have explored these two metrics simultaneously in the context of quantized neural networks, leaving several open questions about the security and trustworthiness of AI algorithms implemented on edge devices.

This research explores the effects of different weight quantization schemes on both clean and adversarial accuracies of neural network models subjected to memory constraints. Two models—VGG-16 and a 3-layer multilayer perceptron (MLP)—were studied with the MNIST and CIFAR-10 image classification datasets. The weights of the models were quantized during training using the deterministic rounding technique. The models were either quantized homogeneously, with all weights quantized to the same precision, or heterogeneously, with weights quantized to different precisions. Several different bitwidths were used for homogeneous quantization, while several different probability mass function-based distributions of bitwidths were used for heterogeneous quantization. To the best of the author's knowledge, this is the first work to study adversarial robustness under homogeneous quantization based on different probability mass functions.

Results show that clean accuracy generally increases when quantized homogeneously at higher bitwidths. For the heterogeneously quantized VGG-16, the distributions that contain a higher quantity of low bitwidth weights have worse performance than those that did not. The heterogeneously quantized MLP performance, however, is generally consistent across distributions. Both models perform far better

on the MNIST dataset than the CIFAR-10. For the MNIST dataset, the VGG-16 model displayed higher levels of adversarial robustness when the quantization of the model contained a greater quantity of lower bitwidth weights. However, the adversarial robustness of the MLP decreases with larger attack strength for all bitwidths. Neither model shows convincing levels of adversarial robustness on the CIFAR-10 dataset. Overall, the results of this research show that both clean and adversarial accuracies have complex dependencies on the total capacity of weight memory and the distribution of precisions among individual weights.

Contents

Signature Sheet **i**

Acknowledgments **ii**

Abstract **iii**

Table of Contents **v**

List of Figures **vii**

List of Tables **1**

1 Introduction **2**

1.1 Objectives 3

2 Background and Related Work **5**

2.1 Quantization Methodologies 5

2.2 Quantization Factors 8

2.3 Quantization and Adversarial Robustness 11

3 Method **16**

3.1 Objective 1 16

3.2 Objective 2 21

4 Results **23**

4.1 Objective 1 23

4.1.1 Homogeneous Quantization of VGG-16 24

4.1.2 Heterogeneous Quantization of VGG-16 25

4.1.3 Homogeneous Quantization of MLP 29

4.1.4 Heterogeneous Quantization of MLP 30

4.2 Objective 2 33

4.2.1 VGG-16 and Adversarial Robustness 33

4.2.2 MLP and Adversarial Robustness 39

5 Discussion **46**

5.1 Homogeneous Quantization 46

5.2 Heterogeneous Quantization 47
5.3 Adversarial Robustness 48
5.4 Comparison to Established Results 49

6 Conclusion and Future Work **51**

List of Figures

1.1 Size of neural networks over time [1]. 2

2.1 Visual representation of uniform quantization [2]. 9
2.2 Visual representation of non-uniform quantization [2]. 9

3.1 VGG-16 architecture [3] 19
3.2 MLP architecture 19
3.3 Heterogeneous Distributions 21

4.1 Homogeneous quantization of VGG-16 on MNIST. 24
4.2 Homogeneous quantization of VGG-16 on CIFAR-10. 25
4.3 Heterogeneous quantization of VGG-16 with bitwidth range 3-6 on MNIST. 26
4.4 Heterogeneous quantization of VGG-16 with bit range 6-9 on MNIST. 26
4.5 Heterogeneous quantization of VGG-16 with bitwidth range 3-6 on CIFAR-10. 27
4.6 Heterogeneous quantization of VGG-16 with bitwidth range 6-9 on CIFAR-10. 28
4.7 Homogeneous quantization of MLP on MNIST. 29
4.8 Homogeneous quantization of MLP on CIFAR-10. 30
4.9 Heterogeneous quantization of MLP with bitwidth range 3-6 on MNIST. 31
4.10 Heterogeneous quantization of MLP with bitwidth range 3-6 on MNIST. 31
4.11 Heterogeneous quantization of MLP with bitwidth range 3-6 on CIFAR-10 32
4.12 Heterogeneous Quantization of MLP with bit range 6-9 on CIFAR-10. 33
4.13 Adversarial robustness of homogeneously quantized VGG-16 on MNIST. 34
4.14 Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 3-6 on MNIST. 35
4.15 Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 6-9 on MNIST. 36
4.16 Adversarial robustness of homogeneously quantized VGG-16 on CIFAR-10. 37
4.17 Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 3-6 on CIFAR-10. 38

4.18 Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 6-9 on CIFAR-10. 38
4.19 Adversarial robustness of homogeneously quantized MLP on MNIST. 39
4.20 Adversarial robustness of heterogeneously quantized MLP with bitwidth range 3-6 on MNIST. 40
4.21 Adversarial robustness of heterogeneously quantized MLP with bitwidth range 6-9 on MNIST. 41
4.22 Adversarial robustness of homogeneously quantized MLP on CIFAR-10. 42
4.23 Adversarial robustness of heterogeneously quantized MLP with bitwidth range 3-6 on CIFAR-10. 43
4.24 Adversarial robustness of heterogeneously quantized MLP with bit range 6-9 on CIFAR-10. 44
4.25 Adversarial robustness of heterogeneously quantized MLP with bitwidth range 1-4 on MLP. 45

List of Tables

3.1 Homogeneous quantization bitwidths 20
3.2 Heterogeneous Distributions for bits 3,4,5,6 with total memory 5p, where p is the number of weights in the model 20
3.3 Heterogeneous Distributions for bits 6,7,8,9 with total memory 8p . . 20

5.1 Accuracy of Homogeneously Quantized Models 46

Chapter 1

Introduction

As neural networks are being utilized for more intricate tasks with expectations for consistently high levels of performance, the complexity and size of these networks continues to grow. High performing neural networks regularly contain hundreds of billions of parameters and utilize gigabytes of data. Figure 1.1 shows the growth of neural networks in terms of number of parameters over time.

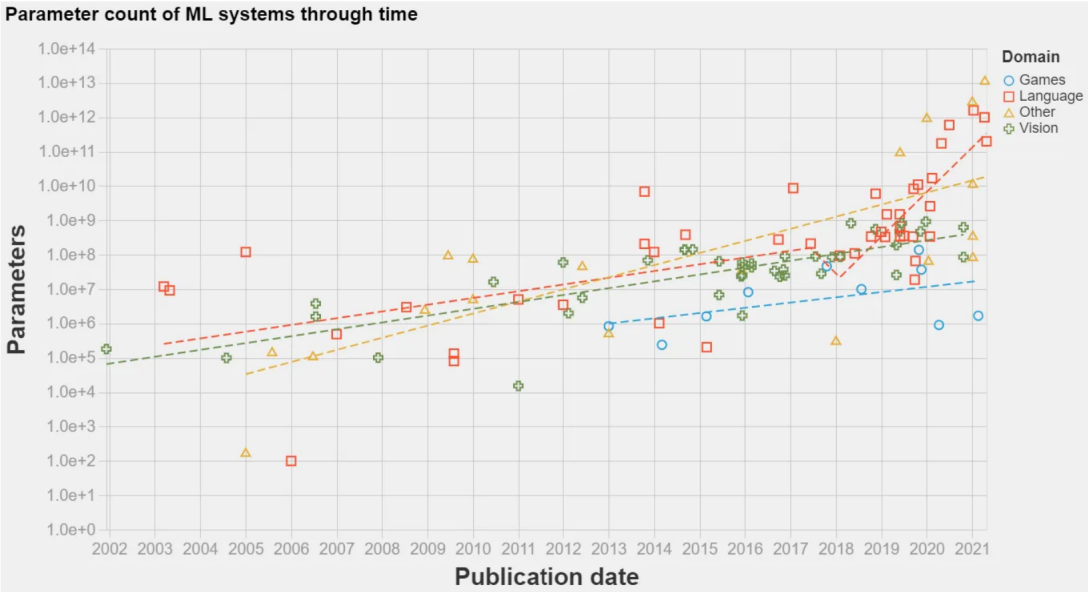

Figure 1.1: Size of neural networks over time [1].

As seen in the figure above, the number of parameters in neural networks and machine learning models has been steadily on the rise from 2002 -2021. In earlier years, models contained around 10000 parameters with large models consisting of

10 million parameters. Over time with the largest models having between 10 and 100 billion parameters. The computations required by these networks can be quite numerous which can cause compatibility issues when it comes to implementation on hardware.

Google (TPU), NVIDIA (Tensor Core), and AMD (CDNA2) have all developed hardware that is specifically optimized for machine learning processes. The advantages of these hardwares are that they are able to perform the intense computations required by machine learning operations with considerable speed and efficiency, however, they consume a substantial amount of energy and memory making them not generally accessible [4]. This means they cannot be easily used on for computing on the edge and thus limits the implementation of neural networks on the edge. This is where quantization comes into play.

Quantization is a set of techniques that can be used to reduce the precision and therefore size of neural networks. By reducing the size of the networks, less memory and energy is required for the hardware. Once neural networks are implemented on hardware, they should be adversarially robust. Adversarial robustness is the ability of the neural network to not succumb to purposeful manipulation and maintain high performance. The effect that limiting the precision of neural network parameters through the process quantization has on adversarial robustness must be investigated.

1.1 Objectives

The overall objective of this research is to study various schemes for quantization that would enable previously resource intensive neural networks to be implemented on devices that have resource constraints. However, a common side effect of quantization is reduced network performance. Thus, this research will attempt to mitigate that problem by ensuring the quantized network maintains both its network performance and adversarial robustness.

There are are two objectives to be accomplished by this research. These objectives are as follows:

1. Create and analyze the results of a variety of widely usable heterogenous quantization schemes that enable machine learning models to perform to high levels of accuracy and be robust against adversarial attacks

2. Compare the results of experiments conducted utilizing the quantization schemes with established results from comparable studies to validate the uniqueness and effectiveness of the research

Chapter 2

Background and Related Work

2.1 Quantization Methodologies

Quantization of neural network parameters (weights, activations, and gradients) can be implemented using several different techniques. These techniques are often split into 2 different categories: deterministic and stochastic. Deterministic quantization provides a direct mapping between the real parameter value and the quantized value, while stochastic quantization relies on discrete distributions from which the quantized values are drawn [5].

There are three different methodologies used to implement deterministic quantization. These methods are rounding, vector quantization, and quantization as optimization [5]. Rounding utilizes a function that turns a set of continuous values into discrete ones. An example of a rounding function established by Courbariaux et al. in 2015 [6] is shown below.

$$x^b = Sign(x) = \begin{cases} +1, & \text{if } x \geq 0 \\ -1, & \text{otherwise} \end{cases} \tag{2.1}$$

In the above function, x^b is a binarized version of the variable which is output depending on x which represents the variable's input value. However, while this function works for the forward pass which is when calculations are made moving forward

through the network, it cannot be applied during the backward pass which is when calculations are made moving back through the network. The need for the additional function derives from the fact that error cannot be back-propogated through the sign function due to the value of the gradient being 0 at most places. This additional function developed by Hinton et al. in 2012 [7] is called a "straight through estimator" and is shown below where E is the loss function.

$$\frac{\delta E}{\delta x} = \frac{\delta E}{\delta x^b} I_{|x|\leq 1} \tag{2.2}$$

The straight through estimator displayed in equation 2.2 provides an estimation for the gradient of a stochastic network neuron. The development and applications of rounding functions have been expanded to include the transformation of other forms of numbers such as floating point to fixed point [8] and real to k-bit integers [9].

Another method, vector quantization, combines model parameter values into subgroups and then uses the arithmetic mean of each subgroup as the substitution for the actual values. An established approach to vector quantization is k-means clustering. The general idea behind k-means clustering is to cluster the weights into groups and replace the weights with their centeroid during evaluation [5].

$$min \sum_{i}^{m} \sum_{n}^{j} \sum_{l}^{k} ||w_{ij} - c_k||_2^2 \tag{2.3}$$

Equation 2.3 demonstrates how this would be performed for a weight matrix $W \in \mathbb{R}^{\mathrm{m}\times\mathrm{n}}$ and where c_k is the centeroid. Product quantization [10] is an expansion on vector quantization in which the weight matrix is split up into submatrices and quantization is applied to each submatrix. Another expansion, residual quantization, quantizes the vectors such that they are grouped into a number or clusters and then the residuals are recursively quantized [10]. Product quantization and k-means clustering tend to be effective as quantization techniques, leading to highly compressed

models with minimal classification accuracy loss while residual quantization does not work well [10].

The third deterministic quantization approach is to view quantization as an optimization problem. Many authors including Rastegari et al. [11], H.Li et al. [12], and Zhu et al. [13] have developed optimization problems that incorporate various model parameters. These optimization problems are then solved to find the best values for the parameters.

In addition to deterministic quantization, a number of stochastic quantization techniques have been proposed to reduce model sizes. The first of these techniques is called random rounding. Random rounding is very similar to the deterministic technique of rounding except that the quantized value of the model parameter is assigned probabilistically. An example developed by Courbariaux et al. [6] in 2015 is shown in equation 2.4.

$$x^b = \begin{cases} +1, & \text{with probability } p = \sigma(x) \\ -1, & \text{with probability 1- p} \end{cases} \tag{2.4}$$

Where $\sigma(x)$ is equal to the hard sigmoid function defined as follows:

$$\sigma(x) = clip(\frac{x+1}{2}, 0, 1) = max(0, min(1, \frac{x+1}{2})) \tag{2.5}$$

In equation 2.4, the underlying principle is that if x is a positive value, then there is a high probability that it will quantize to 1, otherwise it will be quantized to -1. Other researchers have expanded on random rounding by developing different quantization equations and sampling different probability distributions in order to assign values. Random rounding can be used for cases past the binary as well [5].

The second stochastic quantization technique is called probabilistic quantization. In this technique, the weights of the model are assumed to be discretely distributed

and then the parameters are learned and updated based on a selected probability distribution [5].

Associated with each of these quantization techniques are unique challenges. Rounding operations have the potential to cause decreases in performance of the model as the values are rounded and lose precision. Vector quantization which utilizes computation of k-means clustering tends to be even more computationally expensive than rounding[5]. Quantization as an optimization problem suffers from the same increased complexity and computational resource challenges as vector quantization. While random rounding can actually help improve computational efficiency, the estimation required for the gradients can lead to unstable loss functions. Finally, probabilistic quantization is highly sensitivity to chosen probability distributions and neural network architectures [5].

It is also important to note that quantization can either take place during the training process (quantization-aware training) or after the model is fully trained (post-training quantization). Quantization-aware training often yields higher levels of accuracy than post-training quantization due to the fact that the model undergoes re-training. The quantization operation has the potential to inject perturbation into model parameters that have been already been trained. This can cause the model to move away from its previous point of convergence. In order to circumvent this, quantization-aware training quantizes the model parameters after each time the gradient is updated in addition to performing the usual forward and backward pass [2]. Quantization aware training is also more compatible with on-chip training [14].

2.2 Quantization Factors

Division of the values in the real domain, division of values in the quantized domain, precision of model parameters, and range of the quantization domain are the various factors that must be considered when constructing a quantization scheme.

Uniform quantization is when the values in the real domain and the values in the quantized domain are divided into equal groups. This can be seen in Figure 2.1.

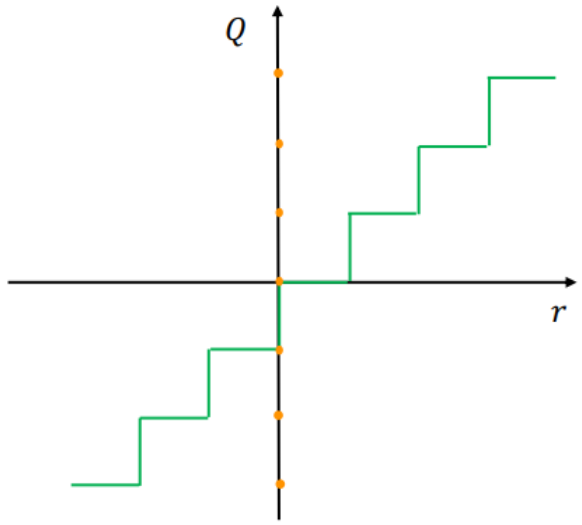

Figure 2.1: Visual representation of uniform quantization [2].

In the figure, r is the real domain while Q is the quantized domain. It can be seen that the quantized values map discretely to the real model values (yellow bullets) and both are divided into evenly spaced segments. Conversely, in non-uniform quantization the quantization domain and real value domains may be split unevenly. This is displayed in Figure 2.2.

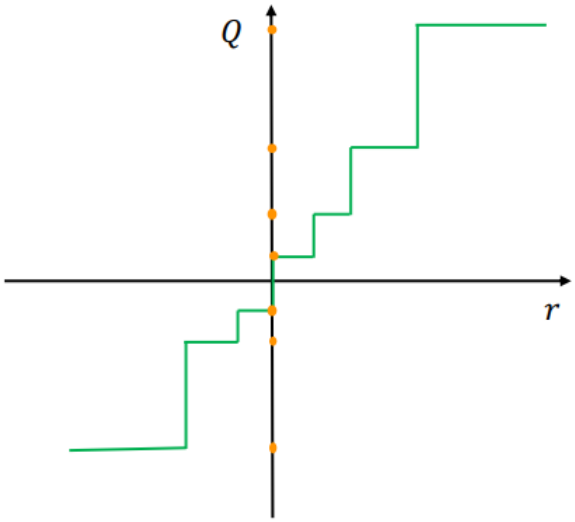

Figure 2.2: Visual representation of non-uniform quantization [2].

Again, r is the real domain while Q is the quantized domain. It can be seen in the figure that the real values are not split into even groups and the quantized values are not spaced evenly.

The difference in precision of the model parameters is what defines the difference between homogeneous quantization versus heterogeneous quantization. In a homogeneous quantization scheme, all model parameters are quantized to the same precision.Homogeneous quantization has been shown to be effective at reducing model size while still maintaining accuracy. However, it does not provide much flexibility when manipulating model parameters and can potentially lead to undesired results if important values end up being reduced to a very low precision [15].

However, in a heterogeneous quantization scheme, model parameters are quantized to different precisions. There are many different ways that a heterogeneous quantization scheme can be implemented. One way is when the selection of quantization levels for a network in a heterogeneous scheme is based on the distribution of the model parameters. This ensures that more important model parameters retain higher levels of precision. Baskin et al. [16] explored a method for bell-shaped distribution. They add noise to the model during the training process in order to emulate a "k-quantile quantization" method with balanced (equal probability mass) bins. The two datasets utilized were CIFAR-10 (60000 color images) [17] and ImageNet-1k (color images with 1000 different classes) [18]. In conjunction they used several different neural network topologies: MobileNet [19] and ResNet-18/34/50 [20]. Their findings showed that they could reach a maximum of 74.84% when they used the ResNet-50 architecture trained on the ImageNet dataset with an 8x decrease in model size as compared to the baseline.

Another heterogeneous quantization method has been developed by Y.Li et al. [21] which they term "additive powers of two." Their method builds on the existing "Powers-of-two" method implemented by Miyashita et al. [22] and Zhou et al. [23].

The Powers-of-two method bounds the quantization level values to either 0 or powers of 2. The additive powers of two method expands upon this by making each quantization level value a sum of a select number of powers-of-two terms. In order to conduct an evaluation of the method, Y. Li et al. quantized ResNet-20 and ResNet-56 models and used them in conjunction with the CIFAR-10 dataset. Results showed that they were able to achieve accuracy values as high as 92.3% with ResNet-20 and 94% with ResNet-56 when the models were quantized to 4 bits.

Heterogeneous quantization can also be based on a pre-determined rule or transformation. This means the quantization levels differ from one another and increase according to the rule that is applied to them. An example of this is Miyashita et al. [22] utilizing a non-uniform logarithmic distribution as the basis for their quantization in order to lower bitwidth precision while obtaining a negligible classification loss. To perform an evaluation, they use pre-trained AlexNet [24] and VGG-16 [25] architectures in the ILSVRC-2012 (a challenge based on the ImageNet dataset) and perform post-training quantization. Their findings showed that when using a base of 2 for the logarithm calculations, they were able to achieve accuracy values of 70.6% for AlexNet and 83.5% for VGG-16. However, with a base of $\sqrt{2}$, the accuracy increased to 75.1% for AlexNet and 89% for VGG-16.

The location of layers in the model can also have an effect on how sensitive the layers are to the effects of quantization as stated by Rastegari et al. [11]. They found that the first and last layers of a network are especially sensitive to quantization and often are not quantized in order to decrease information loss.

2.3 Quantization and Adversarial Robustness

Adversarial robustness is the measure of how a machine learning model behaves when subjected to conditions that are intended to cause poor performance. These conditions often occur at testing time in the form of structured “adversarial examples”

which are special inputs designed to cause poor performance in a neural network [26]. White-box and black-box are the two classifications for adversarial examples. White-box attacks assume the attacker has full access to the model details such as structure and parameters; while in a black-box attack the attacker does not have access to the model architecture or parameters [27].

While quantization does provide benefits in terms of reduced resource consumption, it is also imperative to ensure that quantized models maintain high performance when subjected to adversarial attacks. Adversarial attacks are used to generate the aforementioned adversarial examples. By injecting noise that is undetectable to humans into machine learning inputs, adversarial attacks are able to cause misclassification. Bernhard et al. investigate this in their 2019 study entitled "Impact of Low-bitwidth Quantization on the Adversarial Robustness for Embedded Neural Networks" [27]. They trained a model comprised of convolutional, dense, batch normalization, and activation layers in a quantized aware manner. The chosen datasets were CIFAR-10 and SVHN [28]. The model was subjected to five different attacks. Three were white-box attacks (FGSM [29], BIM [30], and CWL2 [31]) and two were black-box (SPSA [32] and ZOO [33]). The results showed that the models in which weights and activations were quantized were more robust against the FGSM (accuracy as high as 66% for CIFAR-10 and 78% for SVHN), BIM (accuracy as high as 66% for CIFAR-10 and 79% for SVHN), and ZOO attacks (maintaining accuracy values as high as 83% for CIFAR-10 and 94% for SVHN). These models, however, did not perform well against the SPSA and CWL2 attacks. However, the performance of models where only the weights but not activations were quantized, was poor against all types of attacks.

Gui et al. [34] also conducted research into the adversarial robustness of quantized models in their 2019 study entitled "Model Compression with Adversarial Robustness: A Unified Optimization Framework". They attempt to develop an "Adversarially

Trained Model Compression" framework for CNN networks that combines pruning, heterogeneous quantization of weights, and low-rank factorization in order to maintain adversarially robust models. They used the LeNet architecture in conjunction with the MNIST dataset, the ResNet34 architecture in conjunction with the CIFAR-10 dataset, the ResNet34 architecture in conjunction with the CIFAR-100 architecture, and the WideResNet [35] architecture in conjunction with the SVHN dataset. They subject their models to PGD [36], FGSM, and WRM [34] attacks. Their results do show that the compressed model (at 8 bits) is able to perform decently well against the attacks. The model maintains accuracy levels of 75% against PGD (perturbation of 2), 40% against PGD (perturbation of 8), 65% against FGSM, and 80% against WRM.

Gorsline et al. [14] found that the adversarial robustness of a weight quantized neural network can depend on the attack strength. A multilayer perceptron network with 100 hidden nodes was trained and tested on the MNIST dataset and 2 spiral classification problem. For the MNIST dataset, the results showed that when the attack strength of the FGSM attack increased, an increased quantization level (higher precision) only improved model performance to a certain extent. After a certain attack strength, an increase in precision actually causes a decrease in model performance. For the 2 spiral classification problem, as the attack strength increases, the performance of the model drops to close to 0% but then starts to increase again due to the low dimensionality of the dataset.

In the 2019 paper "Defensive Quantization: When Efficiency meets Robustness", Lin et al. [37] attempt to improve the low robustness of quantized models by adjusting the Lipschitz constant of the network during quantization. They stipulate that error can actually be amplified in models by the quantization operation. By controlling the Lipschitz constant, they are able to maintain a small magnitude for the noise in all layers. A Wide ResNet was used with the CIFAR-10 and SVHN datasets. The

results showed that for the Wide ResNet tested with CIFAR-10 and subjected to an FGSM attack, controlling the Lipschitz constant was able to attain a performance accuracy of 51.8% (at a quantization precision of 5 bits). When combined with other adversarial defense techniques (such as adversarial training and feature squeezing), controlling the Lipschitz constant was able to improve model performance against both white-box and black-box attacks.

Tsai et al.[38] also approach the issue of the low robustness of quantized models. They do this by analysing the impact of simultaneous model input and weight perturbation and then attempting to apply new regularization functions to make the model more robust. Neural networks composed of four dense layers and one ReLU layer were trained and tested on the MNIST dataset. The results show that models that were only trained with one type of perturbation did not perform that well. However, the model that was trained with both adversarial training and the new proposed regularizer function performed 24% better than the model that was just adversarially trained. Proving the effectiveness of considering non-singular peturbations and the new regularizer function.

There are several different approaches and factors when it comes to determining an adversarially robust and high performance scheme for quantization. The methodology that will be considered in this paper will focus on an implementation that uses uniform spacing between quantization levels but heterogeneous precision for model individual model weights. As discussed above, heterogeneous quantization provides a level of flexibility and customization not afforded by homogeneous quantization which can enable the size of models to be greatly reduced while maintaining high performance. Heterogeneous quantization has also been show to be robust against adversarial attacks. In this study, an analysis of the most optimal way to distribute precision among neural network parameters when the total number of bits is limited is conducted. The novel contribution of this research is both this approach to quanti-

zation as well as the analysis of its effects on performance and adversarial robustness of the neural network.

Chapter 3

Method

3.1 Objective 1

The goal of this objective is to construct various novel heterogeneous quantization schemes and analyze their performance in terms of average accuracy. The following method was utilized in order to achieve the objective of constructing and investigating various adversarially robust schemes of heterogeneous quantization that can be implemented on devices with resource constraints. The novelty of the proposed method lies in its unique selection of quantization approaches. A deterministic rounding method for heterogeneous quantization, of individual model weights, in which the values of the quantization levels selected from a probability mass function based distribution has not previously been explored. The incorporation of adversarial attacks to ensure adversarial robustness and quantization-aware training to enable on-chip training are also unique components. Heterogeneous quantization of individual model weights is worthy of exploration because it provides the flexibility of reducing model size while maintaining high performance due to the combination of weight bitwidths. The adversarial robustness of the model also benefits from the variance in weight bitwidths.

Quantization technique and model parameters undergoing quantization were selected first. Due to the factors mentioned in Chapter 1, it is evident that the most

straightforward and efficient technique for quantization-aware training is rounding. In this work, only the quantization of model weights will be considered. The equation (3.1) shows the implementation of the rounding function that was used to quantize each weight in this work.

$$x' = \frac{2}{N-1} * round[(N-1) * \frac{clip(x, -1, 1) + 1}{2}] - 1 \tag{3.1}$$

The variable N represents the number of quantization levels. $log_2 N$ is equal to the number of bits the input value is quantized to. The range of quantization is between -1 and 1.

The heterogeneous selection of quantization levels was based on a probability mass function as shown in equation (3.2).

$$np.random.choice([a, b, c, d], size = total{-}number{-}parameters, p = [a\%, b\%, c\%, d\%]) \tag{3.2}$$

Weights were quantized at levels that were randomly assigned from the created distribution. The number of elements in the distribution was the same as the number of weights in the model. The values within the distribution represented the bits in the range that was used to quantize each of the weights (a, b, c, d). Each distribution was subject to 2 constraints.

1. The distribution was created by designating probabilities for each bitwidth in the range (a%, b%, c%, d%). The total of these probabilities must add up to 1. Psuedocode implementation shown in Algorithm 1.

2. The sum of all values in the distribution must be less than or equal to a selected total value. The selected total value of bits of the distribution was determined by multiplying the middle number from the range by the total number of weights

in the model. The selected total value was a representation of the total size of memory on a theoretical hardware chip so the number of bits in the model had to be less than this value. Psuedocode implementation shown in Algorithm 2.

The quantization values were then shuffled and an appropriately shaped tensor with bitwidths from the distribution was passed to each layer so the weights of that layer could be quantized to the values that were in the passed tensor using the quantization function displayed in equation (3.1). For homogeneous quantization, the same method was used with all the bitwidths in the tensor being the same.The models were constructed of customized layers based on Keras dense and 2D convolutional layers which included the distribution for quantization as an input parameter and implemented the quantization function for each weight in the layer. The models were trained used the Adam optimizer with a learning rate of $1 * 10^{-5}$. The loss was calculated using categorical cross entropy and performance was evaluated as the average accuracy across 5 tests.

Algorithm 1 Determining sum of values in the heterogeneous distribution and ensuring they are less than the total memory value

```
p = probability mass function based distribution
for i in range(length of p) do
    if count + p[i] > total memory value then
        p[i] = total memory value - count
    end if
end for
count = count + p[i]
zeros = number of elements that are 0 in p
if zeros ≠ 0 thenexit()
end if
```

Algorithm 2 Assigning quantization values to weights for a given layer

```
shuffle(p)
weight distribution = p[number of parameters in layer]
weight distribution = reshape(p, shape of layer)
model.add(layer(weight distribution))
```

VGG-16, Figure 3.1, and MLP, Figure 3.2, models were used in conjunction with the MNIST and CIFAR-10 datasets. The MNIST dataset is comprised of 60000 28 pixels x 28 pixels grayscale images of handwritten digits from 0-9. The CIFAR-10 dataset is comprised of 60000 32 pixels by 32 pixels colored images from 10 different classes. The average accuracy of the models when quantized at several different homogeneous bitwidths and heterogeneous distributions was evaluated. The MLP architecture was chosen due to the fact that it is not complex and could provide reliable baseline results.

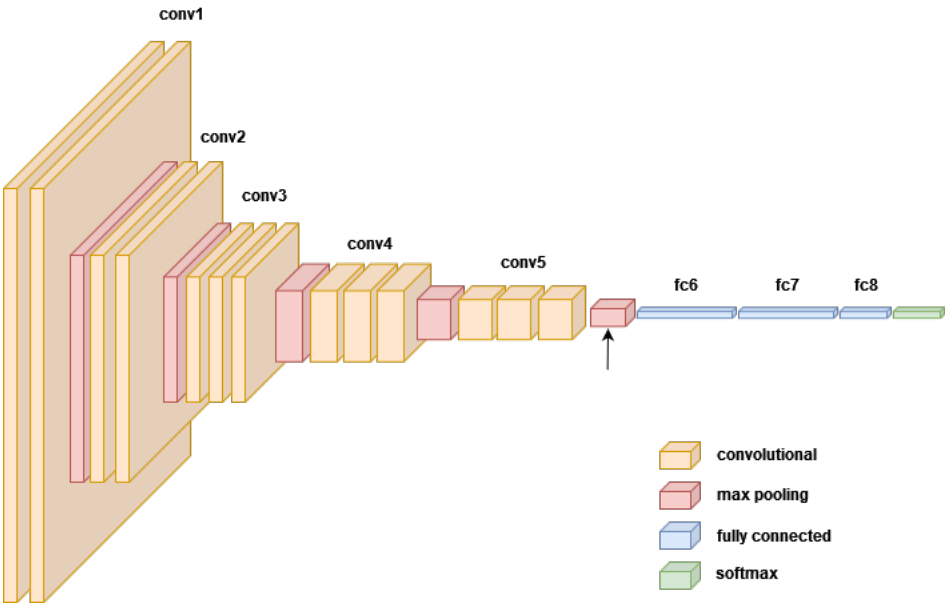

Figure 3.1: VGG-16 architecture [3]

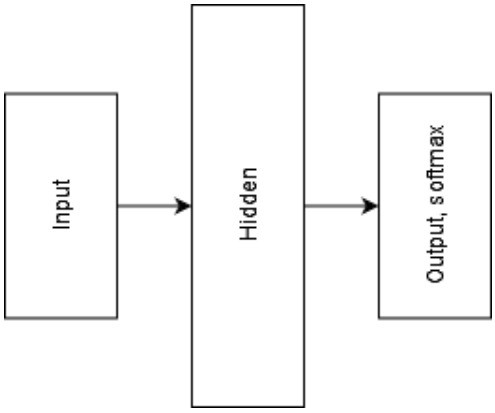

Figure 3.2: MLP architecture

The quantization schemes of the models for the experiments that were run are listed below.

Homogeneous Quantization:

Bitwidth
1 bit
2 bits
3 bits
4 bits
5 bits
8 bits
32 bits

Table 3.1: Homogeneous quantization bitwidths

Heterogeneous quantization with distributions listed in order of decreasing variance:

Distribution	[3 bits, 4 bits, 5 bits, 6 bits]
1	[60%, 0%, 0%, 40%]
2	[40%, 0%, 30%, 30%]
3	[25%, 25%, 25, 25%]
4	[10%, 35%, 35%, 20%]
5	[15%, 35%, 35%, 15%]
6	[10%, 40%, 35%, 15%]
7	[10%, 35%, 40%, 15%]

Table 3.2: Heterogeneous Distributions for bits 3,4,5,6 with total memory 5p, where p is the number of weights in the model

Distribution	[6 bits, 7 bits, 8 bits, 9 bits]
1	[60%, 0%, 0%, 40%]
2	[40%, 0%, 30%, 30%]
3	[25%, 25%, 25, 25%]
4	[10%, 35%, 35%, 20%]
5	[15%, 35%, 35%, 15%]
6	[10%, 40%, 35%, 15%]
7	[10%, 35%, 40%, 15%[

Table 3.3: Heterogeneous Distributions for bits 6,7,8,9 with total memory 8p

Figure 3.1 shows the relative amounts of each bitwidth present in the seven heterogeneous distributions.

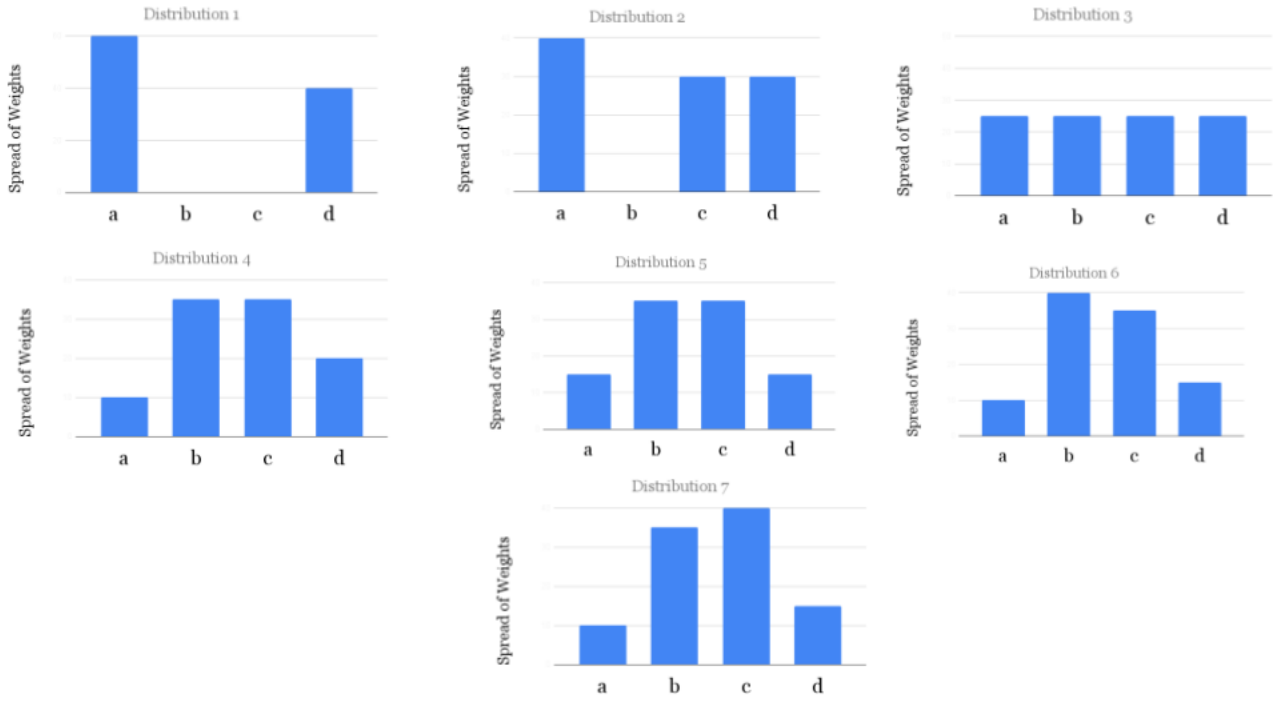

Figure 3.3: Heterogeneous Distributions

3.2 Objective 2

Adversarial robustness against an FGSM attack was evaluated for both the VGG-16 and MLP models on both the MNIST and CIFAR-10 datasets under the different quantization schemes. The attack strengths ranged between 0 and 0.1. The adversarial robustness was evaluated at all values of homogeneous quantization and heterogeneous quantization distributions 1,3,5, and 7 for both models. These distributions were selected as a representative sample due to the differences in their variances. Distribution 1 had the greatest variance and distribution 7 had the lowest variance. Distributions 3 and 5 were in the middle with the variance of 3 being greater than the variance of 5. Investigating how to effectively distribute a limited number of bits among the parameters of a neural network from the context of maintaining

adversarial robustness is the novelty of this research.

The results of these experiments were compared against the results from other established studies in order to show improved performance in terms of the metric of accuracy. Due to the similarities in motivation and methods, results from the heterogeneously quantized model created by Coelho Jr. et al. [39] in "Automatic deep heterogeneous quantization of Deep Neural Networks for ultra low-area, low-latency inference on the edge at particle colliders" was used as a comparator. The homogeneously quantized CNN utilized by Xu et al. in "Deep Neural Network Compression with Single and Multiple Level Quantization" [40] and Zhou et al. in "Dorefa-Net: Training Low Bitwidth Convolutional Nueral Networks with Low Bitwidth Gradients [23] were also used for comparison.

A comparison was also made to the results from "On the Adversarial Robustness of Quantized Neural Networks" by Gorsline et al. [14] and "Impact of Low-bitwidth Quantization on the Adversarial Robustness for Embedded Neural Networks" by Bernhard et al. [27]. Both these papers employ similar architectures and quantization bitwidths to ones studied in this paper.

Chapter 4

Results

4.1 Objective 1

Two models, a VGG-16 and MLP, were quantized both heterogeneously and homogeneously and trained and evaluated on the MNIST and CIFAR-10 datasets. TensorFlow 2 was used and each model was trained for 500 epochs. Each data point was obtained by training and evaluating the model five times and then finding the average and standard deviation of the evaluation accuracies.

4.1.1 Homogeneous Quantization of VGG-16

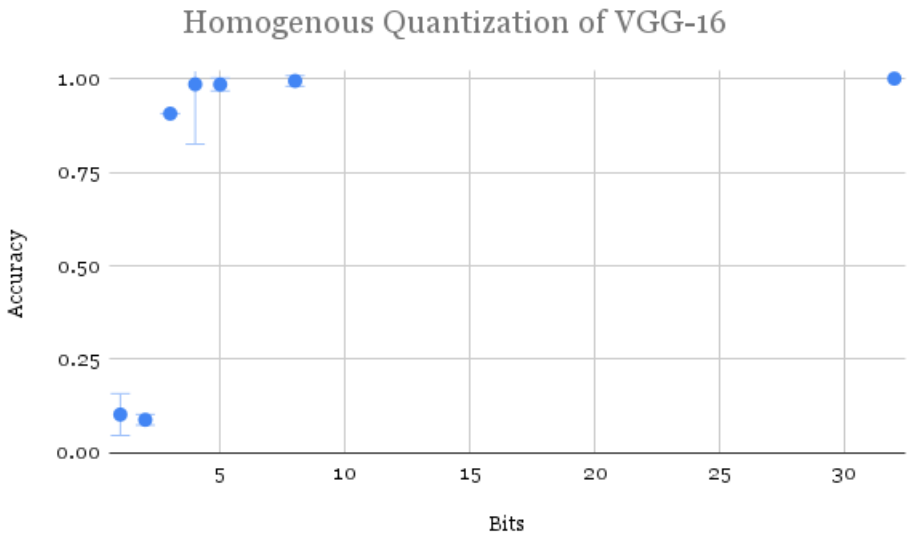

Figure 4.1: Homogeneous quantization of VGG-16 on MNIST.

As seen in the Figure 4.1 the VGG-16 model was quantized homogeneously at the values of 1 bit, 2 bits, 3 bits, 4 bits, 5 bits, 8 bits, and 32 bits and then trained and evaluated on the MNIST dataset. Quantization of the model to 1 bit and 2 bits resulted in poor performance with average accuracies close to random chance. However, once the quantization was increased to 3 bits, the model performance jumped up to an average accuracy around 90%. As the quantization values increased to 4 bits, 8 bits, and 32 bits, it can be seen that the model maintains a high average accuracy. This test using homogeneous quantization was repeated with the CIFAR-10 dataset.

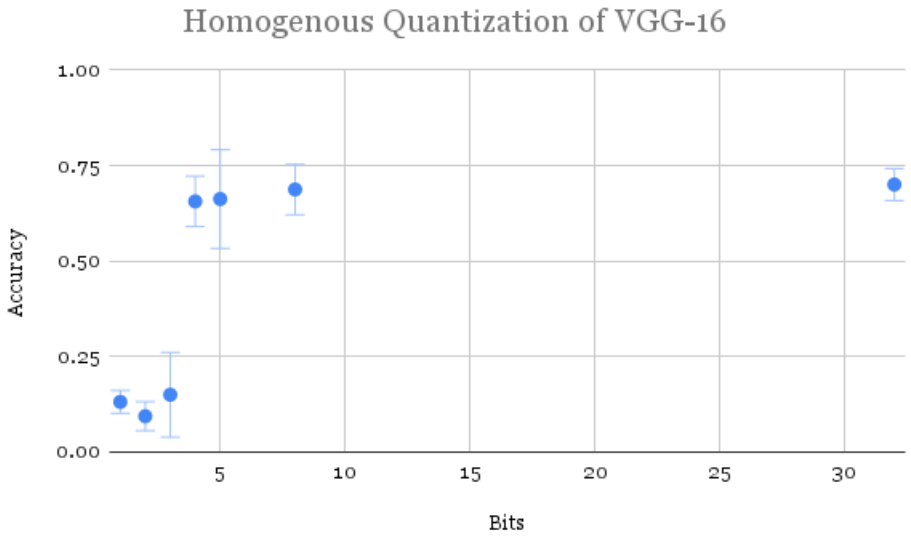

Figure 4.2: Homogeneous quantization of VGG-16 on CIFAR-10.

It can be seen in Figure 4.2 that the quantization values of 1,2, and 3 bits resulted in poor performance. When the quantization value was raised to 4 bits, the average model accuracy increased to a value of approximately 65% and continued to show a trend of increasing performance at 5,8,and 32 bits.

4.1.2 Heterogeneous Quantization of VGG-16

The VGG-16 model was quantized heterogeneously using different distributions of quantization levels while staying under the total memory limit. The distributions were labeled with numerals ranging from 1-7 in order of decreasing variance.

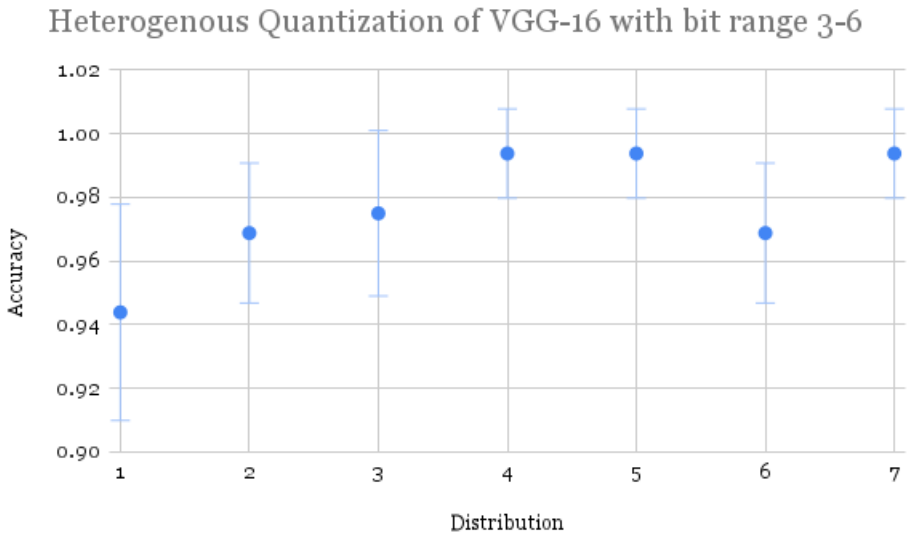

Figure 4.3: Heterogeneous quantization of VGG-16 with bitwidth range 3-6 on MNIST.

As shown in Figure 4.3, the model maintained high average accuracies across distributions with values ranging from $\approx$ 95% to $\approx$ 99% when quantized with a bitwidth range of 3-6 and evaluated on the MNIST dataset. A slight trend of increase in performance can be seen across distributions 1, 2, 3, and 4 before the accuracy fluctuates at distributions 5, 6, and 7.

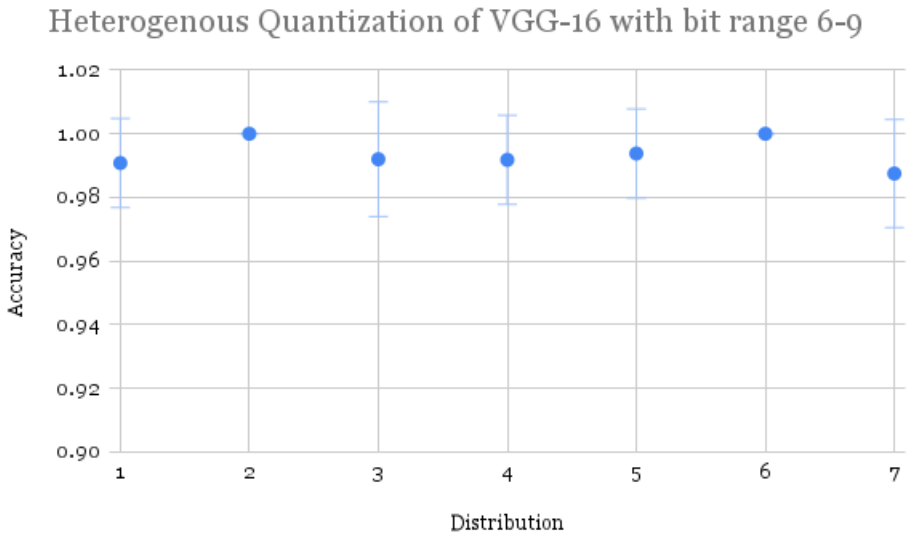

Figure 4.4: Heterogeneous quantization of VGG-16 with bit range 6-9 on MNIST.

In Figure 4.4, the model was quantized at the same distributions but this time the bitwidth range was increased to 6-9 bits. Again the model performed well across all distributions with the average accuracies near 99%. The accuracies slightly increase and decrease across distributions with no visible trend.

ANOVA and t-tests were conducted on the results for both sets of bitwidth ranges to determine 1) if there was statistical significance (p-value of less than .05) within the results of a given bitwidth range and 2) if there was statistical significance (p-value of less than .05) within the results of a given distribution across the bitwidth ranges. The outcomes from these tests indicated that the differences in the means from within the 3-6 bitwidth range were statistically significant while those from within the 6-9 bitwidth range were not. Additionally, the results from distributions 1,2,and 6 were statistically significant across the two bitwidth ranges while the results from distributions 3,4,5,and 7 were not.

The performance of the model was evaluated again using the same distributions on the CIFAR-10 dataset.

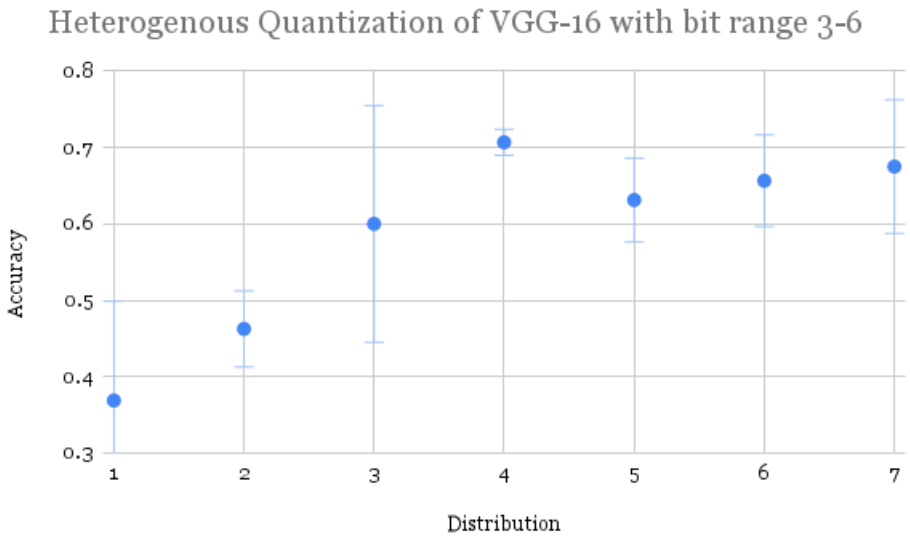

Figure 4.5: Heterogeneous quantization of VGG-16 with bitwidth range 3-6 on CIFAR-10.

It can be seen in Figure 4.5 that the performance of the model on the CIFAR-10

dataset started low at distribution 1 and increased across distributions 2, 3, and 4. The peak performance is at distribution 4 with an average accuracy value of .706. The average accuracy decreased from this value slightly at distribution 5 before increasing again across distributions 6 and 7. The range of the bitwidths was increased to 6-9 and the same tests with the same distributions were repeated.

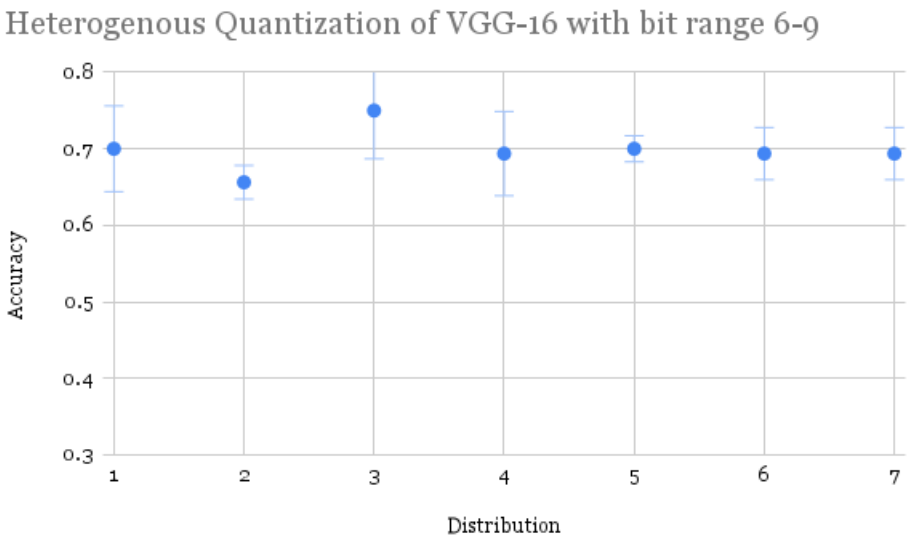

Figure 4.6: Heterogeneous quantization of VGG-16 with bitwidth range 6-9 on CIFAR-10.

As seen in Figure 4.6, the average accuracy of the model with the increased bitwidth range remained fairly consistent across distributions at around .7 . Distribution 3 attained the best performance out of all the distributions with a slightly higher average accuracy while distribution 2 attained the worst with a slightly lower average accuracy.

ANOVA and t-tests were performed again on the results for the tests utilizing the VGG-16 model and CIFAR-10. The outcomes from these tests indicated that the differences in the means from within the 3-6 bitwidth range and 6-9 bitwidth range were both statistically significant. In addition, the differences in the means from distributions 1,2,3, and 5 were statistically significant across the two bitwidth ranges while distributions 4,6, and 7 were not.

4.1.3 Homogeneous Quantization of MLP

A MLP model was quantized homogeneously to bit levels 1, 2, 3, 4, 5, 8,and 32 and trained and evaluated on the MNIST dataset.

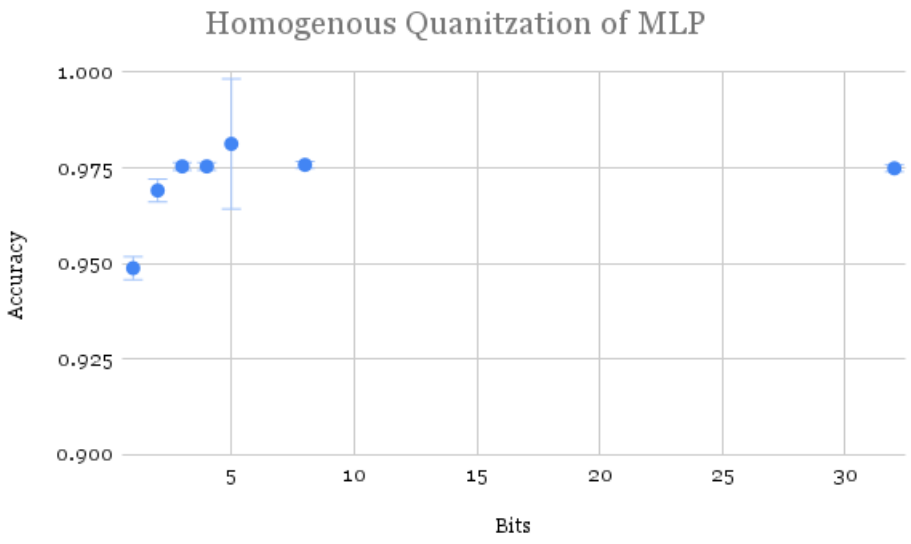

Figure 4.7: Homogeneous quantization of MLP on MNIST.

Displayed in Figure 4.7, the performance of the homogeneously quantized model is consistently high when at any of the bit values. Even at the lowest performance at a quantization of 1 bit, the model still achieved an average accuracy of .949. As the quantization value increases, the performance of the model continued to increase as well. The model was then trained and evaluated on the CIFAR-10 datasets at the aforementioned quantization values.

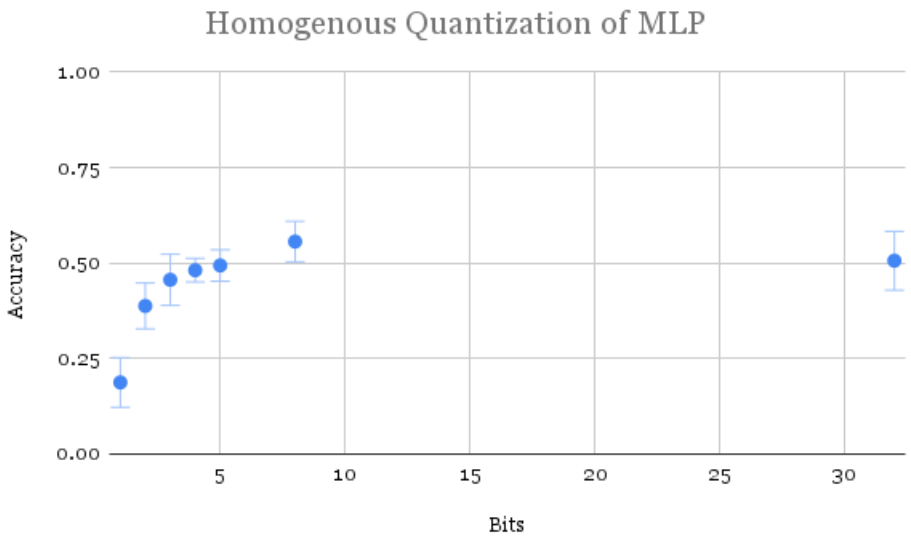

Figure 4.8: Homogeneous quantization of MLP on CIFAR-10.

The performance of the model on the CIFAR-10 dataset was generally low regardless of the quantization value. Even at a quantization value of 32 bits, the average accuracy of the model was about 50%. A trend of increasing average accuracy with increasing quantization values is shown with the lowest average accuracy being around 19% at a quantization value of 1 bit and increasing from there.

4.1.4 Heterogeneous Quantization of MLP

The MLP model was quantized heterogeneously at the same distributions as the VGG-16. The results of the quantized MLP with a bitwidth range of 3-6 when trained and evaluated on the MNIST dataset are displayed in Figure 4.9.

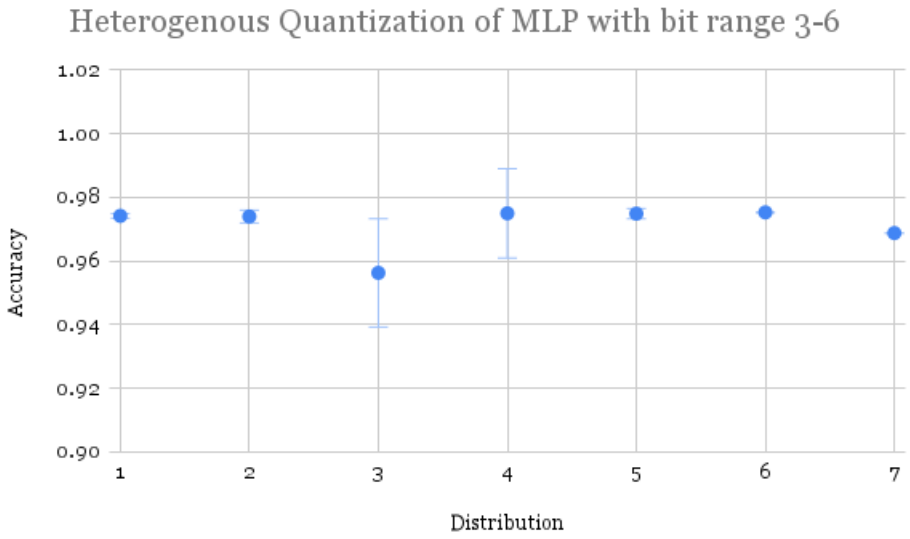

Figure 4.9: Heterogeneous quantization of MLP with bitwidth range 3-6 on MNIST.

The overall performance of the model was consistently high across distributions ranging from an average accuracy $\approx 96\%$ at distribution 3 to an average accuracy of $\approx 98\%$ at distribution 6. The bitwidth range was increased to 6-9 and the quantized model was re-trained and evaluated. The results are displayed in Figure 4.10.

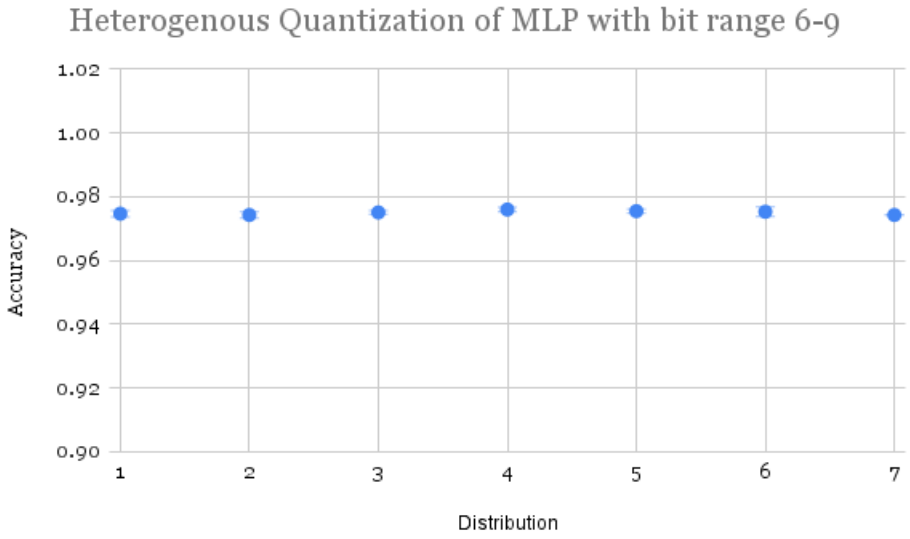

Figure 4.10: Heterogeneous quantization of MLP with bitwidth range 3-6 on MNIST.

The overall performance of the model was high across distributions, as seen in

Figure 4.10 with mean average accuracies consistently between ≈ 97% to ≈ 98% No visible trend is displayed.

The results from an ANOVA/t-test conducted on the results from the tests with the MLP and MNIST showed that the difference in means from within the 3-6 bitwidth range were statistically significant (p-value of less than .05) while those from within the 6-9 bitwidth range were not. Additionally, the only distributions for which the differences were statistically significant across the two bitwidth ranges were distribution 3 and 7. The tests were repeated using the CIFAR-10 dataset.

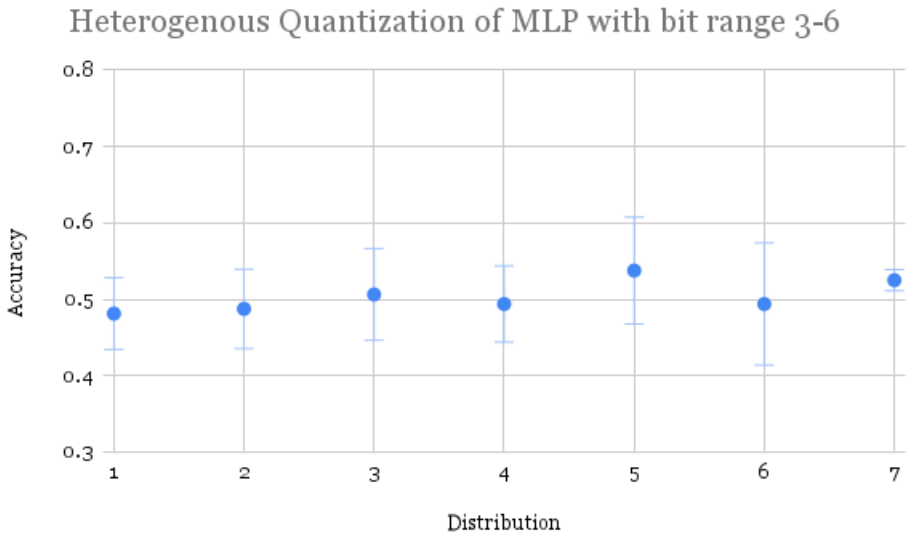

Figure 4.11: Heterogeneous quantization of MLP with bitwidth range 3-6 on CIFAR-10

As seen in Figure 4.11, the average accuracy of the model quantized at a bitwidth range of 3-6 ranged from the lowest value of approximately 48% at distribution 1 to the highest value of approximately 54% at distribution 5. The MLP model was then quantized with a bit range of 6-9 and evaluated.

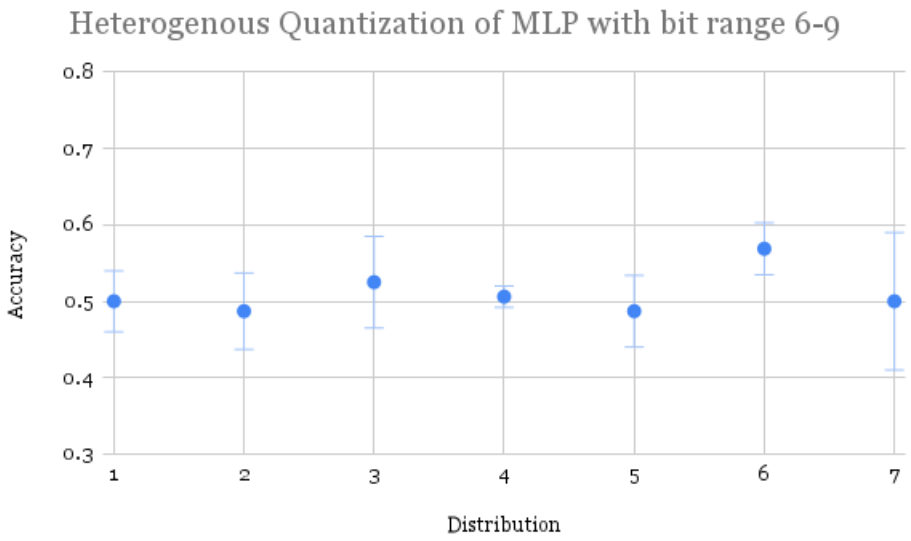

Figure 4.12: Heterogeneous Quantization of MLP with bit range 6-9 on CIFAR-10.

In Figure 4.12, the highest average accuracy of the model is obtained at distribution 6 ($\approx$ 57%). The lowest average accuracy was at distributions 2 and 5 with a value of $\approx$ 49%. Once again this range of values indicates mediocre performance from the model across distributions.

The findings from the ANOVA/t-test conducted on the results on the tests from the heterogeneously quantized MLP on CIFAR-10 showed that none of the results either within bitwidth ranges or across them were statistically significant.

4.2 Objective 2

4.2.1 VGG-16 and Adversarial Robustness

In order to evaluate adversarial robustness,the homogeneously and heterogeneously quantized VGG-16 models were subjected to different strengths of FGSM attack: .03, .06, .1. The models were evaluated on both the MNIST and CIFAR-10 datasets. Figure 4.13 shows the adversarial robustness of a homogenously quantized VGG-16 model on MNIST.

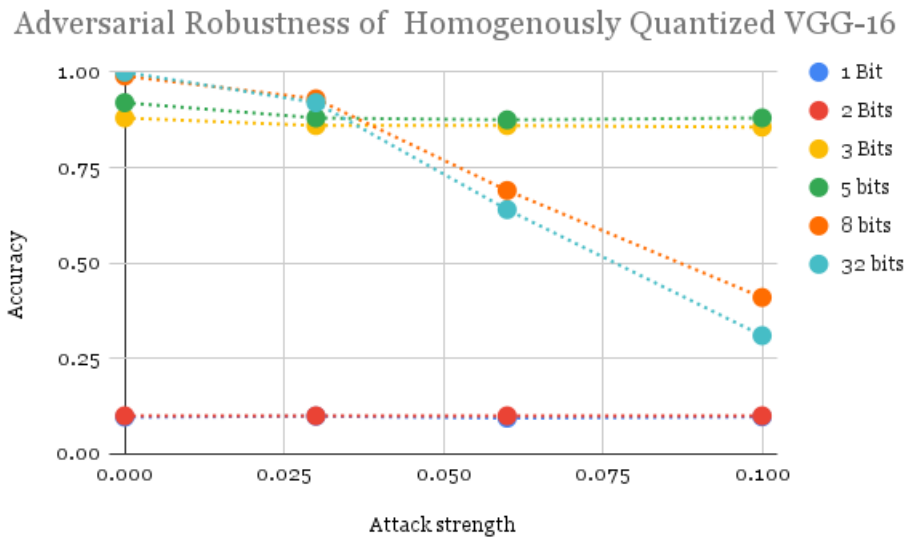

Figure 4.13: Adversarial robustness of homogeneously quantized VGG-16 on MNIST.

Quantization of the model to 1 and 2 bits resulted in poor average accuracies across attack strengths. At quantization values of 3 and 5 bits, the average model accuracy remains fairly consistent across attack strengths with substantially increased values ranging from $\approx$ 86% to $\approx$ 92%. When the model was quantized to 8 and 32 bits, a trend of decreasing average accuracy with increasing attack strength is shown. The VGG-16 model was quantized heterogeneously with a bitwidth range of 3-6 at distributions 1, 3, 5, and 7. The adversarial robustness of the quantized model evaluated on MNIST is shown in Figure 4.14.

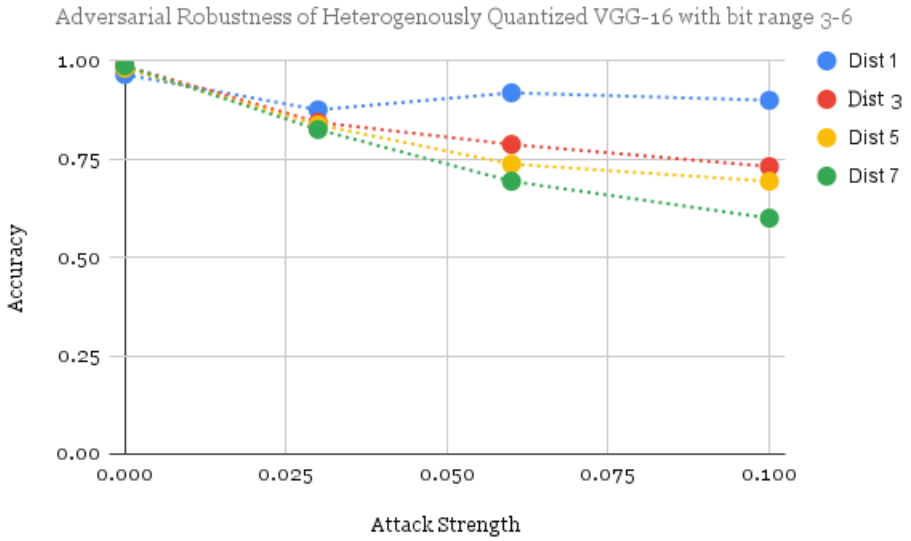

Figure 4.14: Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 3-6 on MNIST.

Different distributions can clearly be seen to have different performances. Distribution 1 had the best performance with a consistent high average accuracy across attack strengths. The remaining distributions show a trend of decreasing average accuracy with increasing attack strength. Overall, distribution 3 outperformed distribution 5 which outperformed distribution 7. The bitwidth range for the quantized VGG-16 model was increased to 6-9 and the model was again evaluated for adversarial robustness with the same distributions on MNIST. The results are shown in Figure 4.15.

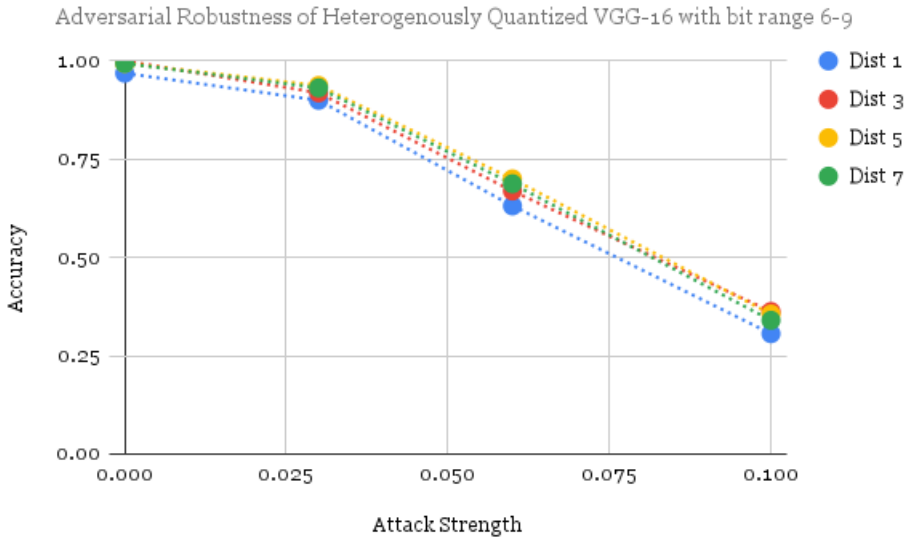

Figure 4.15: Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 6-9 on MNIST.

The model at all distributions starts out fairly robust against the initial attack of .03 strength and then show a decrease in average accuracy as the attack strength increases. The tests were repeated using the CIFAR-10 dataset and an increased selection of attack strength: .01, .03, .045, .06, .075, and .1. The adversarial robustness of a homogeneously quantized VGG-16 is shown in Figure 4.16.

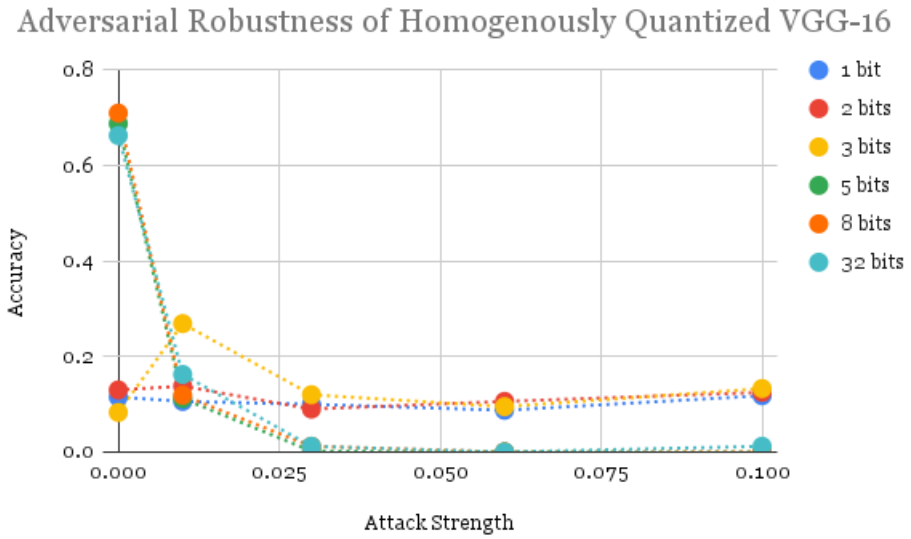

Figure 4.16: Adversarial robustness of homogeneously quantized VGG-16 on CIFAR-10.

The adversarial robustness of the model when quantized at 1, 2, and 3 bits was poor across all attack strengths with low average accuracies. When quantized at 5, 8, and 32 bits, the model was not able to defend against attack well, with the average accuracy dropping immediately at the attack strength of .01 and continuing to decrease as the attacks strength increases. The model was then heterogeneously quantized using a bitwidth range of 3-6. Results are shown in Figure 4.17.

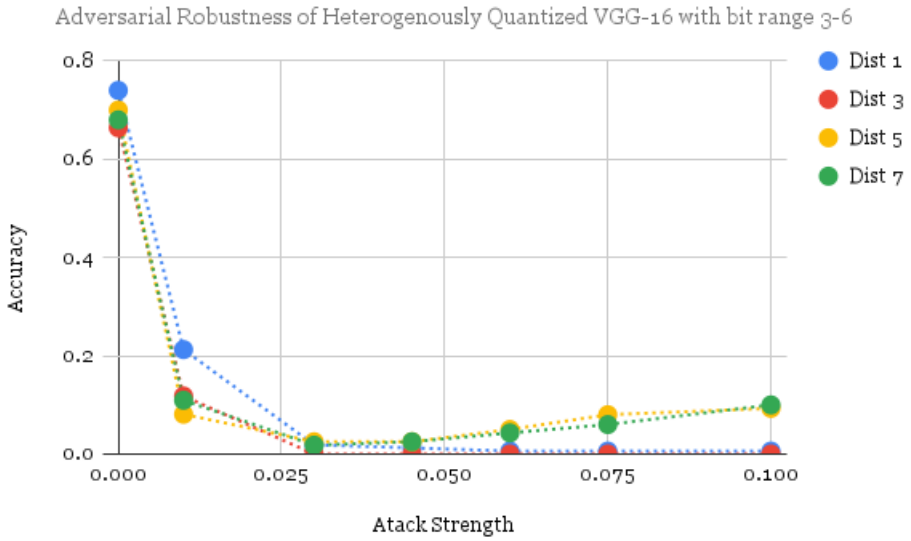

Figure 4.17: Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 3-6 on CIFAR-10.

The performance of all distributions was similar. The average accuracy dropped immediately at attack strength .01 and continued to decrease to values of less than 10% as the attack strength increased. The bitwidth range of quantization was increased to 6-9 and results are displayed in Figure 4.18.

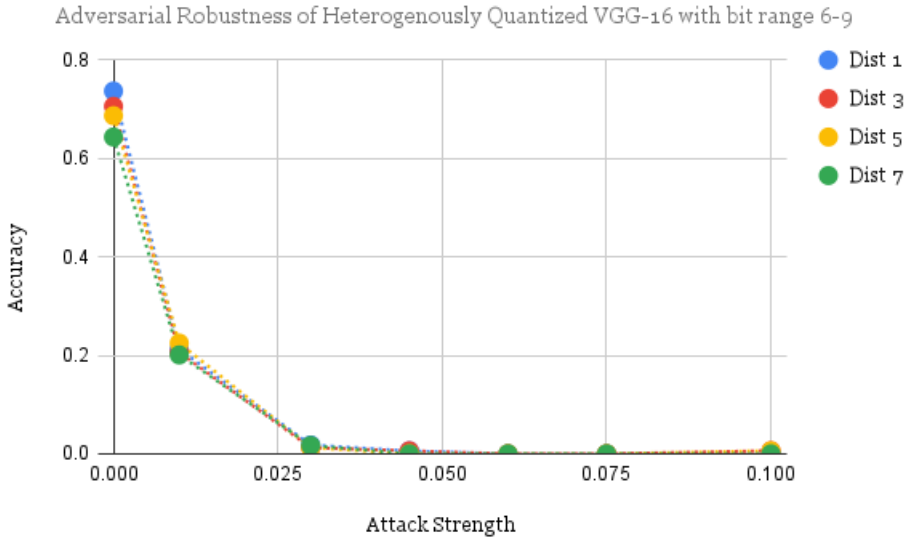

Figure 4.18: Adversarial robustness of heterogeneously quantized VGG-16 with bitwidth range 6-9 on CIFAR-10.

The performance of the model quantized at these distributions was very similar to the performance at the lower bitwidth heterogeneous quantization. The model had poor adversarial robustness with the average accuracy dropping at the first attack strength and then continuing to decrease to 0 for the rest.

4.2.2 MLP and Adversarial Robustness

The adversarial robustness of a homogeneously and heterogeneously quantized MLP was evaluated in the same manner as the VGG-16 model using different levels of FGSM attack strength with the MNIST and CIFAR-10 datasets. The adversarial robustness of a homogeneously quantized MLP on MNIST is show in Figure 4.19.

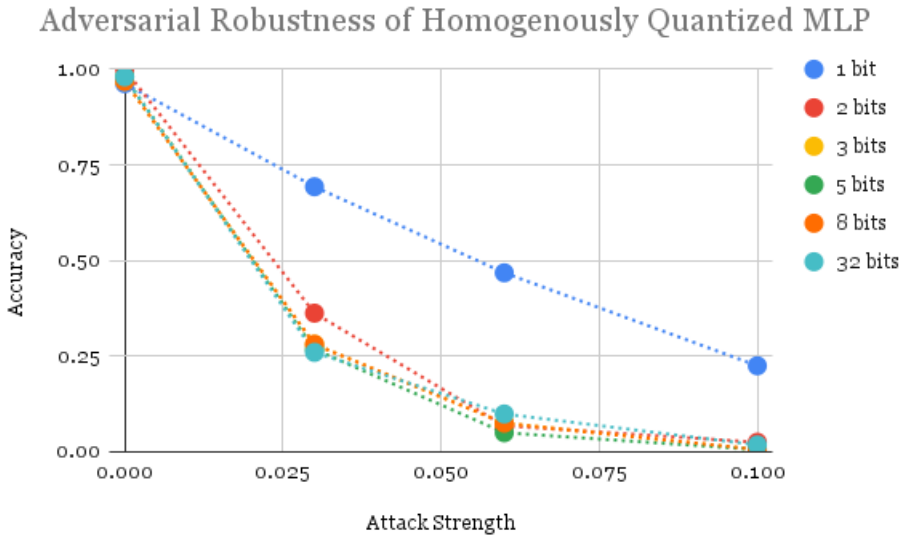

Figure 4.19: Adversarial robustness of homogeneously quantized MLP on MNIST.

The MLP, when quantized to any quantization value shows a trend of decreasing performance with increasing attack strength. The 1 bit quantized MLP displays the best performance with a higher average accuracy at each attack strength than the other quantization values. The test was repeated with a heterogeneously quantized MLP with bitwidth range 3-6 using distributions 1,3,5, and 7. The adversarial robustness of the quantized model evaluated on MNIST is shown in Figure 4.20

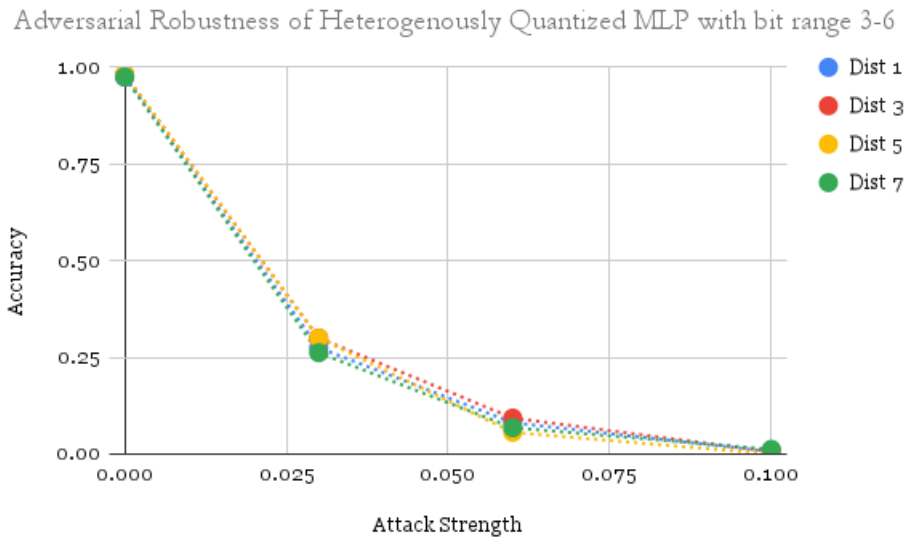

Figure 4.20: Adversarial robustness of heterogeneously quantized MLP with bitwidth range 3-6 on MNIST.

In Figure 4.20, it can be seen that for all distributions, the average accuracy decreases when attack strength increases. Also the average accuracy at each attack strength is very similar across attack strengths. The bitwidth for the heterogeneous quantization was increased to 6-9 and the same adversarial robustness of the same distributions were evaluated on MNIST. The results are shown in Figure 4.21.

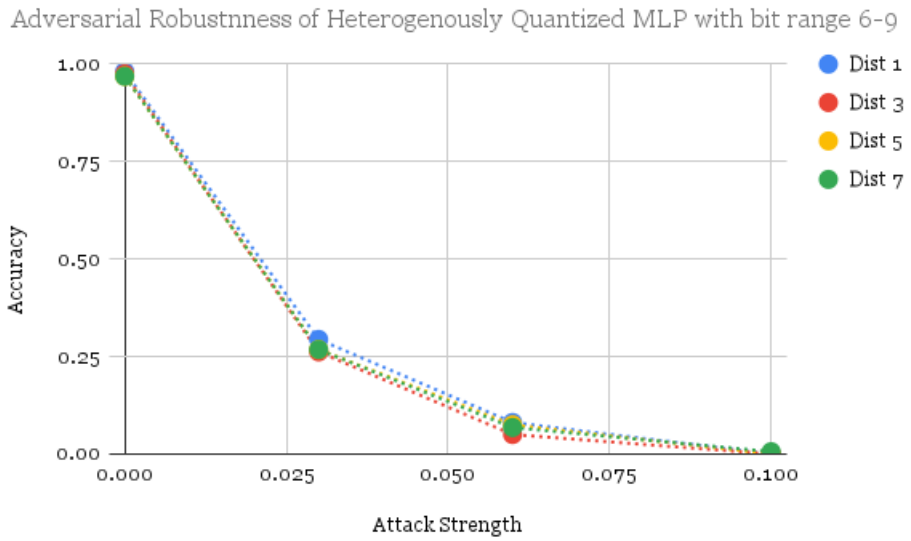

Figure 4.21: Adversarial robustness of heterogeneously quantized MLP with bitwidth range 6-9 on MNIST.

Once again, as seen in figure 4.21, when the attack strength increases, the average accuracy decreases for all distributions. The average accuracy is very close in value at each attack strength across distributions. The adversarial robustness tests were repeated for the CIFAR-10 dataset. The results from the homogeneously quantized MLP is displayed in Figure 4.22.

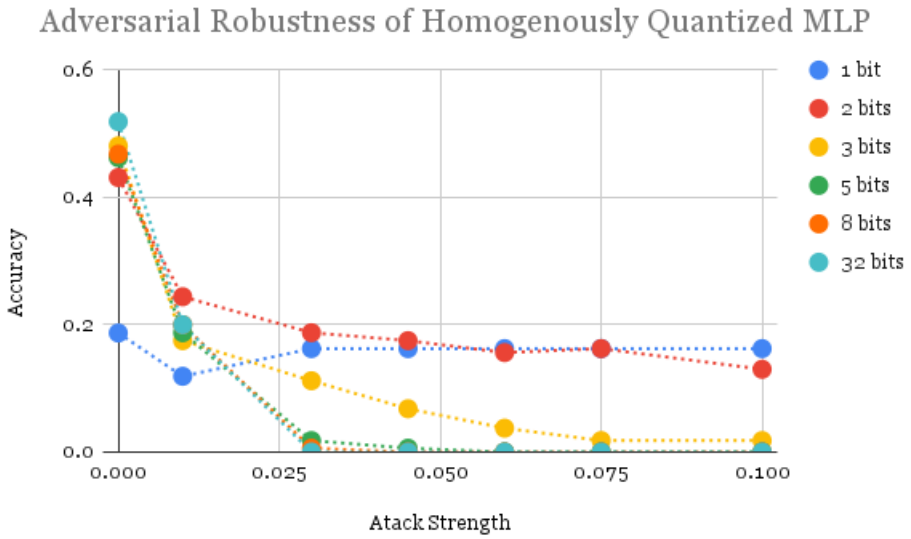

Figure 4.22: Adversarial robustness of homogeneously quantized MLP on CIFAR-10.

The 1 and 2 bit quantized model had a higher average accuracy at each attack strength the rest of the rest of the quantization bitwidth. However, the overall performance of the model was poor with the highest average accuracy of approximately 25% achieved by the 2 bit quantized model at an attack strength of .01. The MLP was heterogeneously quantized with a bitwidth of 3-6. Results of this are shown in Figure 4.23.

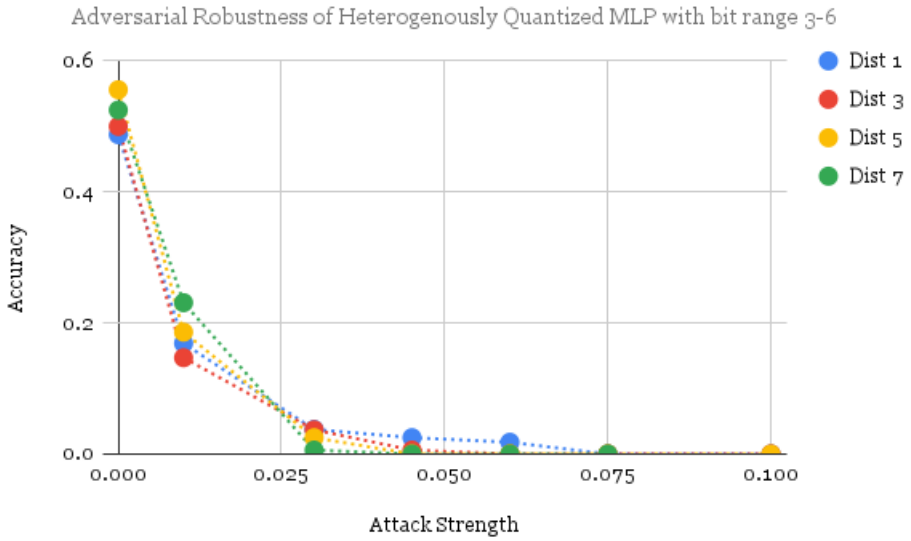

Figure 4.23: Adversarial robustness of heterogeneously quantized MLP with bitwidth range 3-6 on CIFAR-10.

The performance all quantization distributions was similar. The average accuracy of the model dropped low at the attack strength of .01 and continued to decrease to values close to 0 at each increasing attack strength. The bitwidths were increased to 6-9 and the adversarial robustness was evaluated again. Results shown in Figure 4.24.

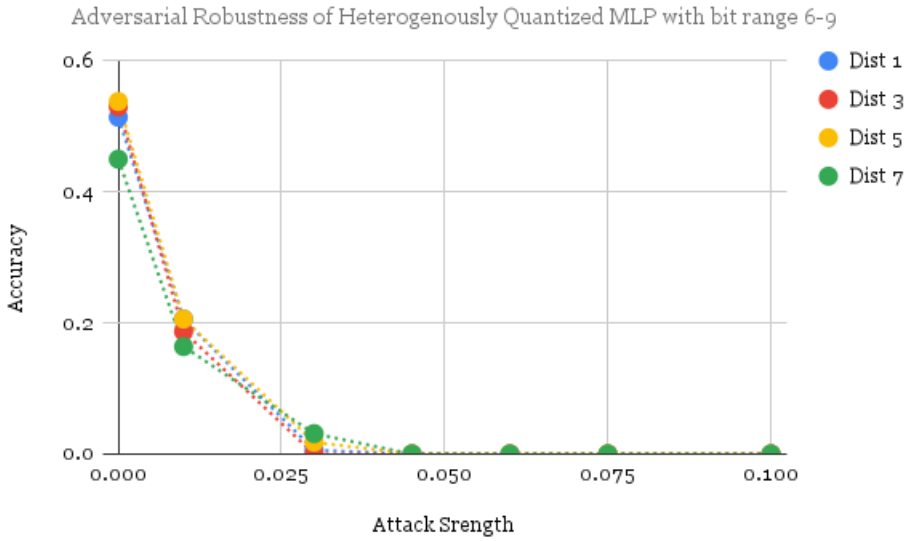

Figure 4.24: Adversarial robustness of heterogeneously quantized MLP with bit range 6-9 on CIFAR-10.

The performance of the model is similar to the performance when heterogeneously quantized at lower bitwidths. All distributions had similar performances and average accuracies started low at the attack strength of .01 dropped for all to 0 at the higher attack strengths. In order to verify whether lower bit quantization could provide a defense against adversarial attack for the MLP model, the bitwidths were lowered to values in the range of 1-4. The adversarial robustness of this quantized model is show in Figue 4.25.

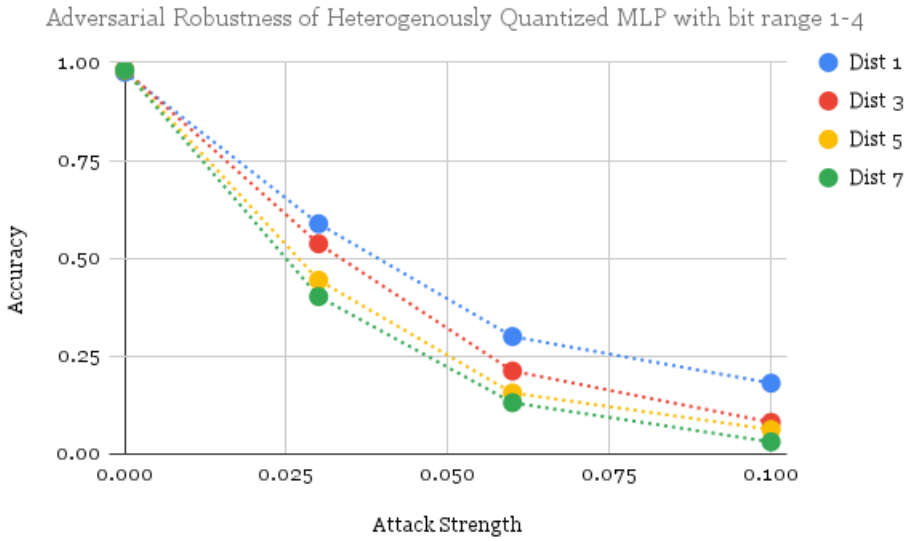

Figure 4.25: Adversarial robustness of heterogeneously quantized MLP with bitwidth range 1-4 on MLP.

It can be seen that distribution 1, which has the highest quantity of low bitwidth weights, performs the best and as the number of low bitwidth weights decreases, so does the average accuracy of the model at each attack strength.

Chapter 5

Discussion

5.1 Homogeneous Quantization

Table 5.1: Accuracy of Homogeneously Quantized Models

Model	Quantized Bitwidths	Dataset
	1 2 3 5	
VGG-16	10.2% 9.0% 90.6% 98.4%	MNIST
MLP	94.9% 96.9% 97.5% 98.1%	
VGG-16	13.1% 9.4% 15% 66.2%	CIFAR-10
MLP	18.8% 38.8% 45.6% 49.4%	

Homogeneous quantization of the VGG-16 architecture to low bitwidths of 1 and 2 bits resulted in poor performance on both datasets. The complexity of this convolutional architecture makes the model especially susceptible to the performance degrading effects of quantization. This can be seen when these results are compared to the results from the simpler 3 layer multilayer perceptron network which performed well when quantized to low bitwidths. Once the VGG-16 model was quantized to values of 3 bits or above, the average accuracy increased significantly on the MNIST dataset indicating that even 3 bits could be enough to achieve satisfactory performance.

5.2 Heterogeneous Quantization

The heterogeneously quantized VGG-16 with a bit range of 3-6 evaluated on MNIST showed an increase in average accuracy as the variance of the quantization distribution decreased (figure 4.3). Distributions with smaller variance contained greater numbers of high value weights. This result combined with the poor performance of the VGG-16 at the lowest quantization values of 1 and 2 bits, a performance bottleneck suggests whereby the performance of the model is constrained by the number of low precision weights it contains. This is supported by the results from the heterogeneously quantized VGG-16 with a bit range of 6-9 (Figure 4.4). The VGG-16 model performs well regardless of distribution due to the fact that there are no low precision weights present in the quantization. The results of the t-test conducted showed that distributions 1, 2, and 6 showed statistically significant results across bit ranges. This means that changing the values of the weight precision range did have an effect on the average accuracy further supporting the conclusion that increasing the bit range resulted in an increased average accuracy.

When evaluated on the CIFAR-10 dataset, the heterogeneously quantized VGG-16 once again showed a trend of increase in average accuracy with a decrease in variance within the distribution (Figure 4.5). However, when the bit range was increase 6-9, there was no significant differences in the average accuracy across distributions demonstrating that the distribution had no effect due to the increased quantity of high precision weights in all distributions. Distributions 1, 2, and 5 had a statistically significant difference across the two bit ranges where increasing the bit range resulted in greater average accuracies.

The performance of the heterogeneously quantized MLP model when evaluated on MNIST was consistently good across both bit ranges as well as all distributions. A statistical significance was present between the bit ranges for distributions 3 and

7. For these distributions, increasing the bit range caused an increase in average accuracy showing that the MLP performance still was aided by having more high precision weights.

Even when heterogeneously quantized with a combination of high and low precision weights, it is apparent that more complex architectures, such as the VGG-16, are more susceptible to the effects of quantization.

No one distribution had a consistent performance across datasets or models. This coupled with the differences in model performance across bit ranges when heterogeneously quantized and bitwidths when homogeneously quantized, suggests that the distribution of precisions among individual weights has an effect on accuracy and thus each weight should be quantized to the appropriate number of bits for optimal performance.

The overall lower performance of both models on the CIFAR-10 dataset must be noted. This can be attributed to the lack of data augmentation performed on this multi-faceted dataset which was done in an effort to keep comparisons between the two models consistent.

5.3 Adversarial Robustness

When homogeneously quantized to 1 and 2 bits, the VGG-16 model did not achieve satisfactory levels of performance when evaluated on MNIST at any attack strength. Based on the very low average accuracies obtained by the VGG-16 model at an attack strength of 0, it is evident that the VGG-16 model was unable to perform in any capacity at quantization values this low. At quantization values of 3 and 5 bits, the VGG-16 model was able to maintain a high average accuracy against attacks at all tested strengths. During training, the model when quantized to these lower values was able to learn a more rigid decision boundary. This means the model did not have the flexibility to overfit the data which allowed it to defend against the adversarial attack.

At higher values of precision, the proximity of the data to the decision boundary was very close, making the model highly susceptible to the effects of adversarial attack. This is evidenced by the accuracy of the VGG-16 model dropping with increasing attack strength when quantized to 8 and 32 bits.

For the heterogeneously quantized VGG-16 with bit range 3-6 evaluated on MNIST, distribution 1 had the highest average accuracy at each attack strength. As the variance of the distribution decreased so did the average accuracy at each attack strength. The greater number of low precision weights in distributions with greater variance, once again, allowed the model to be able to learn an non-adaptable decision boundary to defend against adversarial attack. Once the bit range was increased to 6-9, the average accuracy for each distribution decreased as the attack strength increased as no low value bits were present.

The low value quantization defense is somewhat present in the MLP. When homogeneously quantized and evaluated on MNIST, the 1 bit quantized model had the best performance at all attack strengths. However, because the quantization did not have as great as an effect on the MLP, the defense is not apparent in any other homogeneous quantization nor any heterogeneous quantization distribution with either bit range.

Due to the complexity issues of the CIFAR-10 dataset discussed previously, neither model was able to satisfactorily defend against attack.

5.4 Comparison to Established Results

Zhou et al. [23] investigated the effects of low bit-width quantization on the accuracy of convolutional neural networks. Their results show that low weight bitwidths can have a significant effect on the accuracy of the CNN models when evaluated on complex datasets. Their model was only able to achieve an accuracy .395 on ImageNet when quantized to a bitwidths of 1. This is also reflected in the results of this

study. The VGG-16 model achieved low average accuracies on the CIFAR-10 dataset when quantized to low bitwidths as well as when quantized with distributions that contained greater numbers of low bitwidths.

Cohelo Jr et al. [39] saw accuracies of 72.3% and 72.8% for their two energy-optimized and bit-optimized heterogeneously quantized models that were trained with quantization aware training. When performing on MNIST, the VGG-16 and MLP model used in this study were both able to reach greater level of accuracy of 90% or higher at each quantization distribution. This difference may be attributed to the importance of weight-by-weight quantization used in this study versus the layer-wise quantization used by Cohelo Jr et al. In addition, there were differences in method due to the implementation of bit and energy quantization that were not explored in this study.

Bernhard et al.[27] and Gorsline et al. [14] obtained results about the effect of quantization on the adversarial robustness of various models. Bernhard et al. homogeneously quantize their model to low bitwidths ranging from 1 to 5 bits and evaluate their models on the CIFAR-10 dataset as well as checking for adversarial robustness against an FGSM attack. While the accuracies achieved by their low bitwidth quantized models were higher than those found in this study, they also found that none of the quantized models were able to successfully defend against FGSM attack, which is consistent with the results from this study obtained on the CIFAR-10 dataset.Gorsline et al. investigated the adversarial robustness of quantized MLP models on the MNIST dataset against FGSM attack. Their results which showed that increasing attack strength caused a decrease in test accuracy across quantization bit levels is consistent with the results obtained in this study. The results from this study additionally show that low precision weights can help a network maintain adversarial robustness. A greater number of low precision weights in a model causes the model to be less likely to overfit the data and therefore less susceptible to adversarial attacks.

Chapter 6

Conclusion and Future Work

This study investigated the performance and adversarial robustness effects of quantization on two different neural network architectures and two different datasets. The study was unique due to its analysis of the effect that the distribution of precision among model weights when the total number of bits is limited had on both clean and adversarial accuracy. Results showed that for the VGG-16 quantization as low as 3 bits could produce adequately high average accuracies when evaluated on MNIST. The MLP was able to achieve high average accuracies when evaluated on MNIST starting at quantization of 1 bit. Results from heterogeneous quantization showed that higher quantities of high bitwidth weights does lead to better performance, although generally performance was high for both models across distributions. Lower average accuracies were obtained for both models on the CIFAR-10 dataset but similar trends were displayed.

In terms of adversarial robustness, the VGG-16 model when quantized using distributions that contained a greater number of low bitwidths, was able to perform better against adversarial attack than when quantized with distributions that contained greater numbers of high value bitwidths.The MLP model showed an inclination of the low bitwidth defense as well due to the fact that it achieved the highest adversarial accuracy when quantized to very low precision and when using distributions that had high numbers of very low precision bits. Neither model was able to satisfactorily

defend against attack on the CIFAR-10 dataset.

These results lead to the conclusion that the total number of bits does have an effect on the performance of the model. In order to obtain good clean performance, the performance bottleneck must be overcome by allowing for an adequate total number of bits which allows for an adequate number of higher precision weights. However, the total number of bits is also limited by the distribution used for quantization. The distribution must contain an adequate number of lower precision weights that will enable the model to be less susceptible to adversarial attack.

Steps for future work include expanding both the neural network architectures and datasets that are utilized. The effects of quantization was already more apparent in the more complex architecture of the VGG-16 as compared to the simpler MNIST. It is likely that an even more complex neural network architecture would be even more susceptible to the effects of quantization. Additionally, quantization of different network parameters such as activations and gradients would have an effect on overall performance. Data augmentation will also be required to understand how the quantized models perform and the results from this study transfer to bigger datasets such as CIFAR-100 and Tiny ImageNet.

Additionally, considerations for implementation onto hardware must be made. Potential challenges arise from the usage of bitwidths that are not powers of 2, such as 3, 6, 5, and 7, since these cannot be easily implemented into hardware. It is possible that these values could be replaced with powers of 2 values, as long as enough bits are allocated such that the performance bottleneck is overcome and the low precision defense is enabled. However, the results of doing so will have to be investigated.

Overall the results and findings of this study, while novel, were consistent when compared to the results from established studies. The results showed that both clean accuracies and adversarial robustness are affected by quantization distributions.

Bibliography

[1] J. Sevilla, "Parameter counts in machine learning," Jul 2021. [Online]. Available: https://towardsdatascience.com/parameter-counts-in-machine-learning-a312dc4753d0

[2] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, "A survey of quantization methods for efficient neural network inference," *arXiv preprint arXiv:2103.13630*, 2021.

[3] K. Leung, "How to easily draw neural network architecture diagrams," *Towards Data Science*, Aug 2021. [Online]. Available: https://towardsdatascience.com/how-to-easily-draw-neural-network-architecture-diagrams-a6b6138ed875

[4] P. Wayner, "What is ai hardware? how gpus and tpus give artificial intelligence algorithms a boost," Sep 2022. [Online]. Available: https://venturebeat.com/ai/what-is-ai-hardware-how-gpus-and-tpus-give-artificial-intelligence-algorithms-a-boost/

[5] Y. Guo, "A survey on methods and theories of quantized neural networks," *arXiv preprint arXiv:1808.04752*, 2018.

[6] M. Courbariaux, Y. Bengio, and J.-P. David, "Binaryconnect: Training deep neural networks with binary weights during propagations," in *Advances in neural information processing systems*, 2015, pp. 3123–3131.

[7] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning lecture 6a overview of mini-batch gradient descent," *Cited on*, vol. 14, no. 8, p. 2, 2012.

[8] S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, "Deep learning with limited numerical precision," in *International conference on machine learning*. PMLR, 2015, pp. 1737–1746.

[9] S. Wu, G. Li, F. Chen, and L. Shi, "Training and inference with integers in deep neural networks," *arXiv preprint arXiv:1802.04680*, 2018.

[10] Y. Gong, L. Liu, M. Yang, and L. Bourdev, "Compressing deep convolutional networks using vector quantization," *arXiv preprint arXiv:1412.6115*, 2014.

[11] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European conference on computer vision*. Springer, 2016, pp. 525–542.

[12] H. Li, S. De, Z. Xu, C. Studer, H. Samet, and T. Goldstein, "Training quantized nets: A deeper understanding," *CoRR*, vol. abs/1706.02379, 2017. [Online]. Available: http://arxiv.org/abs/1706.02379

[13] C. Zhu, S. Han, H. Mao, and W. J. Dally, "Trained ternary quantization," *arXiv preprint arXiv:1612.01064*, 2016.

[14] M. Gorsline, J. Smith, and C. E. Merkel, "On the adversarial robustness of quantized neural networks," *CoRR*, vol. abs/2105.00227, 2021. [Online]. Available: https://arxiv.org/abs/2105.00227

[15] Z. Liu, K. Cheng, D. Huang, E. P. Xing, and Z. Shen, "Nonuniform-to-uniform quantization: Towards accurate quantization via generalized straight-through estimation," *CoRR*, vol. abs/2111.14826, 2021. [Online]. Available: https://arxiv.org/abs/2111.14826

[16] C. Baskin, E. Schwartz, E. Zheltonozhskii, N. Liss, R. Giryes, A. M. Bronstein, and A. Mendelson, "UNIQ: uniform noise injection for the quantization of neural networks," *CoRR*, vol. abs/1804.10969, 2018. [Online]. Available: http://arxiv.org/abs/1804.10969

[17] A. Krizhevsky, V. Nair, and G. Hinton, "The cifar-10 dataset," *online: http://www. cs. toronto. edu/kriz/cifar. html*, vol. 55, no. 5, 2014.

[18] 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.

[19] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.04861, 2017. [Online]. Available: http://arxiv.org/abs/1704.04861

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[21] Y. Li, X. Dong, and W. Wang, "Additive powers-of-two quantization: An efficient non-uniform discretization for neural networks," *arXiv preprint arXiv:1909.13144*, 2019.

[22] D. Miyashita, E. H. Lee, and B. Murmann, "Convolutional neural networks using logarithmic data representation," *arXiv preprint arXiv:1603.01025*, 2016.

[23] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen, and Y. Zou, "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients," *arXiv preprint arXiv:1606.06160*, 2016.

[24] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," *CoRR*, vol. abs/1404.5997, 2014. [Online]. Available: http://arxiv.org/abs/1404.5997

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[26] "Adversarial example using fgsm tensorflow core." [Online]. Available: https://www.tensorflow.org/tutorials/generative/adversarial_fgsm

[27] R. Bernhard, P.-A. Moellic, and J.-M. Dutertre, "Impact of low-bitwidth quantization on the adversarial robustness for embedded neural networks," in *2019 International Conference on Cyberworlds (CW)*. IEEE, 2019, pp. 308–315.

[28] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," 2011.

[29] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[30] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *CoRR*, vol. abs/1607.02533, 2016. [Online]. Available: http://arxiv.org/abs/1607.02533

[31] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.

[32] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE transactions on automatic control*, vol. 37, no. 3, pp. 332–341, 1992.

[33] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," ser. AISec '17. New York, NY, USA:

Association for Computing Machinery, 2017, p. 15–26. [Online]. Available: https://doi.org/10.1145/3128572.3140448

[34] S. Gui, H. N. Wang, H. Yang, C. Yu, Z. Wang, and J. Liu, "Model compression with adversarial robustness: A unified optimization framework," *Advances in Neural Information Processing Systems*, vol. 32, pp. 1285–1296, 2019.

[35] S. Zagoruyko and N. Komodakis, "Wide residual networks," *CoRR*, vol. abs/1605.07146, 2016. [Online]. Available: http://arxiv.org/abs/1605.07146

[36] T. Zheng, C. Chen, and K. Ren, "Distributionally adversarial attack," *CoRR*, vol. abs/1808.05537, 2018. [Online]. Available: http://arxiv.org/abs/1808.05537

[37] J. Lin, C. Gan, and S. Han, "Defensive quantization: When efficiency meets robustness," *arXiv preprint arXiv:1904.08444*, 2019.

[38] Y.-L. Tsai, C.-Y. Hsu, C.-M. Yu, and P.-Y. Chen, "Non-singular adversarial robustness of neural networks," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 3840–3844.

[39] C. N. Coelho, A. Kuusela, S. Li, H. Zhuang, J. Ngadiuba, T. K. Aarrestad, V. Loncar, M. Pierini, A. A. Pol, and S. Summers, "Automatic heterogeneous quantization of deep neural networks for low-latency inference on the edge for particle detectors," *Nature Machine Intelligence*, vol. 3, no. 8, pp. 675–686, 2021.

[40] Y. Xu, Y. Wang, A. Zhou, W. Lin, and H. Xiong, "Deep neural network compression with single and multiple level quantization," ser. AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[41] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4820–4828.

[42] C. Zhang and P. Zhou, "A quantized training framework for robust and accurate reram-based neural network accelerators," in *2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2021, pp. 43–48.