REPRESENTATION QUALITY OF NEURAL NETWORKS LINKS TO ADVERSARIAL ATTACKS AND DEFENCES

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural networks have been shown vulnerable to a variety of adversarial algorithms. A crucial step for understanding the rationale behind this lack of robustness is to assess the potential of the neural networks' representation to encode the existing features. Here, we propose a method to understand the representation quality of the neural networks using a novel test based on Zero-Shot Learning, entitled Raw Zero-Shot. The principal idea is that if an algorithm learns rich features, such features should be able to interpret *'new or unknown'* classes as a combination of previously learned features. This is because unknown classes usually share several regular features with recognised (learned) classes, given that the features learned are general enough. We further introduce two metrics to assess this learned representation which interprets unknown classes. One is based on inter-cluster validation technique, while the other is based on the difference in the representation between the case when the class is unknown and the case when it is known to the classifier. Experiments suggest that several adversarial defences not only decrease the attack accuracy of some attacks but also improve the representation quality of the classifiers. Further, a low p-value of the paired-samples t-test suggests that several adversarial defences, in general, change the representation quality significantly. Moreover, experiments also reveal a relationship between the proposed metrics and adversarial attacks (a high Pearson correlation coefficient and low p-value).

1 INTRODUCTION

Adversarial samples are noise-perturbed samples that can fail neural networks for tasks like image classification. Since they were discovered some years ago by Szegedy (2014), both the quality and variety of adversarial samples have grown. These adversarial samples can be generated by a specific class of algorithms known as adversarial attacks (Nguyen et al., 2015; Brown et al., 2017; Moosavi-Dezfooli et al., 2017; Su et al., 2019). Most of these adversarial attacks can also be transformed into real-world attacks (Sharif et al., 2016; Kurakin et al., 2016; Athalye & Sutskever, 2018), which confer a big issue as well as a security risk for current neural networks' applications. Despite the existence of many variants of defences to these adversarial attacks (Goodfellow et al., 2014; Huang et al., 2015; Papernot et al., 2016; Dziugaite et al., 2016; Hazan et al., 2016; Das et al., 2017; Guo et al., 2018; Song et al., 2018; Xu et al., 2017; Madry et al., 2018; Ma et al., 2018; Buckman et al., 2018), 'no known learning algorithm or procedure can defend consistently' (Carlini & Wagner, 2017; Tramèr et al., 2017; Athalye et al., 2018; Uesato et al., 2018; Vargas & Kotyan, 2019; Tramer et al., 2020). This shows that a more profound understanding of the adversarial algorithms is needed to enable the formulation of consistent and robust defences.

Several works have focused on understanding the reasoning behind such a lack of robust performance. It is hypothesised in Goodfellow et al. (2014) that neural networks' linearity is one of the main reasons for failure. Other investigation by Thesing et al. (2019) shows that with deep learning, neural networks learn false structures that are simpler to learn rather than the ones expected. Moreover, research by Vargas & Su (2019) unveil that adversarial attacks are altering where the algorithm is paying attention. In Sabour et al. (2015), it is discussed that an adversarial sample may have different internal representation than the benign sample. The authors show that internal representations of adversarial samples are remarkably similar to different images of different true-class and links adversarial robustness to representations learned by deep neural networks.

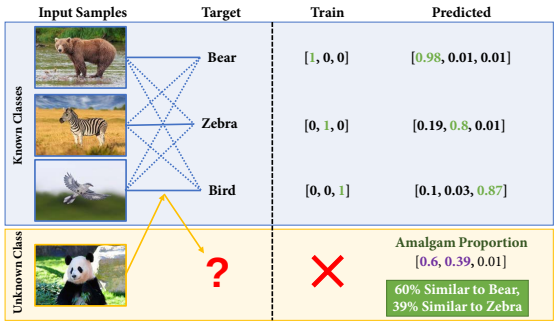


Figure 1: Raw Zero-Shot Illustration

Contributions: In this article, we try to open up a new perspective on understanding adversarial algorithms based on evaluating the representation quality of unseen classes based on learned classes. We do this, by verifying that the representation quality of neural networks is indeed linked with the adversarial attacks and defences. Specifically, we propose a methodology based on Zero-Shot Learning entitled Raw Zero-Shot (Section 3) for evaluating the representation quality of the neural networks.

We conducted experiments over the soft-labels of an unfamiliar class to assess the representation quality of the classifiers. This is based on the hypothesis that, if the classifier is capable of learning useful features, an unfamiliar class would also be associated with some of these learned features (Amalgam Proportion) (Figure 1). We call this type of inspection over unfamiliar class, Raw Zero-Shot (Section 3). Furthermore, we also introduce two associated metrics to evaluate the representation quality of neural networks. One is based upon Clustering Hypothesis (Section 3.1), while the other is based on Amalgam Hypothesis (Section 3.2).

We evaluated our Raw Zero-Shot test over a wide assortment of datasets (and classifiers) such as Fashion MNIST, CIFAR-10, and a customised Imagenet to assess the representation quality of the vanilla classifiers (Section 4). We also evaluated different adversarial defences to prove that when an adversarial defence is applied to a classifier, it gives better representation quality than the vanilla classifier. We also conducted a paired samples t-test to determine the statistical relevance of the effect of adversarial defences on the representation quality (Section 5). We then reveal a link between the representation quality and attack susceptibility by verifying that the proposed metrics have a high Pearson correlation coefficient with the adversarial attacks (Section 6).

2 Related Works

Understanding Adversarial Attacks: Since the discovery of adversarial samples in Szegedy (2014), many researchers have tried to understand the adversarial attacks. It is hypothesised in Goodfellow et al. (2014) that neural networks' linearity is one of the principal reasons for failure against an adversary. A geometric perspective is analysed in Moosavi-Dezfooli et al. (2018), where it is shown that adversarial samples lie in shared subspace, along which the decision boundary of a classifier is positively curved. Further, in Fawzi et al. (2018), a relationship between sensitivity to additive perturbations of the inputs, and the curvature of the decision boundary of deep networks is shown. Another aspect of robustness is discussed in Madry et al. (2018), where authors suggest that the capacity of the neural networks' architecture is relevant to the robustness. It is also stated in Ilyas et al. (2019) that the adversarial vulnerability is a significant consequence of the dominant supervised learning paradigm and a classifier's sensitivity to well-generalising features in the known input distribution. Also, research by Tao et al. (2018) argue that adversarial attacks are entangled with interpretability of neural networks as results on adversarial samples can hardly be explained. Further, the existence of different internal representations learned by neural networks for an adversarial sample compared to a benign sample is shown in Sabour et al. (2015). In this article, we explore a new perspective to understand adversarial attacks and defences based on the representation quality of the neural networks evaluated using Amalgam Proportion.

Zero-Shot learning: Zero-Shot learning is a method to estimate unfamiliar classes which do not appear in the training data. The motivation of Zero-Shot learning is to transfer knowledge from recognised (learned) classes to unfamiliar classes. Existing methods address the problem by estimating unfamiliar classes from an attribute vector defined manually for all classes. For each class, whether such an attribute (like colour, shape) relates to the class or not is represented by one or zero. Lampert et al. (2009) introduced *Direct Attribute Prediction (DAP)* model, which learns each parameter of the input sample for estimating the attributes of the sample from the feature vector generated. Based on this research, other zero-shot learning methods have been proposed which uses an embedded representation generated using a natural language processing algorithm instead of a manually created attribute vector (Norouzi et al., 2013; Fu et al., 2015; Akata et al., 2015; Zhang & Saligrama, 2016; Bucher et al., 2016). Zhang & Saligrama (2015) proposed a different strategy by constructing the histogram of known classes distribution for an unknown class to estimate unknown classes. They assume that the unknown classes are the same if these histograms generated in the prediction domain and the source domain are similar. Our Raw Zero-Shot test is distinguished from other zero-shot learning algorithms as in Raw Zero-Shot the neural network has no access to features (attribute vector), or additional supplementary knowledge.

3 RAW ZERO-SHOT

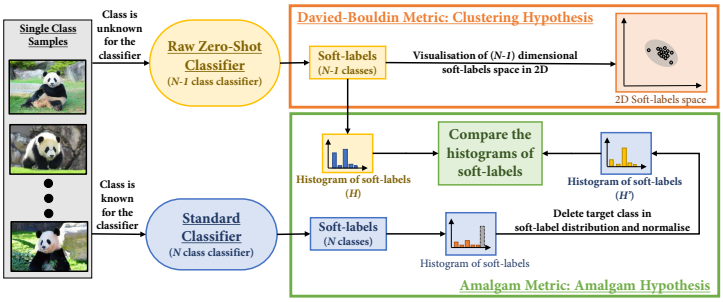


Figure 2: Illustration of proposed metrics.

Raw Zero-Shot is a learning test in which only $N-1$ of the N classes in the dataset are presented to the classifier during training, or in other words, all the samples of one specific class are removed from the standard training dataset. Such a classifier trained on only $N-1$ of the N classes is called *'Raw Zero-Shot Classifier'*. Please note that a *'Standard Classifier'* is trained on all N classes has N soft-label dimensions in the soft-label space. In contrast, a Raw Zero-Shot Classifier has only $N-1$ soft-label dimensions in the soft-label space due to the forced exclusion of a class. The excluded unknown class then can be predicted as a combination of the remaining $N-1$ soft-label dimensions of the known (learned) classes. We call this combination as *'Amalgam Proportion'* (Figure 1). During testing, only the unknown class (excluded class from N) is provided to the classifier. Amalgam Proportion for the given unknown class is recorded for the classifier. This process is iterated for all potential (N) classes, excluding a different class each time.

Soft-labels of a classifier composes a space in which a given image would be categorised as a weighted vector involving the previously learned classes. If neural networks can learn the features existing in the classes, it would be reasonable to consider that the Amalgam Proportion also describes a given image as a combination of the previously learned classes (Figure 1). Similar to a vector space in linear algebra, the soft-labels can be combined to describe unknown objects in this space. In our example (Figure 1), the unknown class (Giant Panda) is represented as a combination of previously recognised (learned) classes (Bear, Zebra, Bird) where 60% of the features of Bear (like body-shape) and 39% of the features of Zebra (like stripes pattern) is 'associated' with the Giant Panda. This is analogous to how children associate unseen objects (Giant Panda) as a combination of recognised objects (Bear and Zebra) when they are asked to describe the unseen object with their learned knowledge (Walker & Gopnik, 2014; Walker et al., 2016). Thus, all the images of the class Giant Panda should have similar Amalgam Proportion as the hypothetical classifier can associate Giant Panda with some features of Zebra and Bear classes.

Metrics are then computed over the Amalgam Proportion of the unknown (excluded) class to assess this representation quality of a classifier, (Figure 2). These metrics are each based on a different hypothesis of what defines a feature or a class. In the same way, as there are various aspects of robustness, *there are also different variations of representation quality*. Therefore, *our metrics are complementary, each highlighting a different perspective of the whole*. The following subsections define them.

3.1 DAVIES–BOULDIN METRIC (DBM) – CLUSTERING HYPOTHESIS

We can use cluster validation techniques to assess the representation (Amalgam Proportion), considering that the cluster of Amalgam Proportion of an unfamiliar class would constitute a class in itself. Here, we choose for simplicity Davies-Bouldin Index (Davies & Bouldin, 1979), one of the most used metrics in internal cluster validation. Hence, Davies–Bouldin Metric (DBM) for an unknown class can be defined as follows:

$$\text{DBM} = \left(\frac{1}{n} \sum_{j=1}^{n} |z_j - G|^2 \right)^{1/2}$$

in which, n is the number of samples (samples from unknown class), G is the centroid of the cluster formed by the soft-labels of all the n samples, and z is soft-label of a single sample of unknown class. A denser cluster would have a lower DBM Score representing a consistent view taken by the classifier in terms of features learned from the known classes.

3.2 AMALGAM METRIC (AM) – AMALGAM HYPOTHESIS

Differently from the previous metric, here we establish our metric on the hypothesis that the classes that are learned by a classifier share some similarity with the unfamiliar class and the classifier can associate this similarity in its representation while evaluating these unfamiliar classes. This hypothesis formulates from the fact that humans can combine available perceptual information with stored knowledge of experiential regularities which helps us to describe things that are 'similar' as close and things that are 'dissimilar' as far apart (Casasanto, 2008). However, what would constitute the baseline Amalgam Proportion for a given unfamiliar class still needs to be determined to assess the extent of the classifier to exploit this existence of similarity between classes.

To calculate the baseline Amalgam Proportion of a given unknown class, we use here the assumption that 'Standard Classifiers should output a good approximation of the Amalgam Proportion since the class is known to the Standard Classifier in the training phase. We thus associate the evaluated Amalgam Proportion of the Raw Zero-Shot Classifier and the baseline Amalgam Proportion of the Standard Classifier for a given class with our Amalgam Metric (AM) (Figure 2) as,

$$\text{AM} = \frac{\|H' - H\|_1}{N - 1} \quad \text{where} \quad H = \sum_{j=1}^{n} z_j, \quad H' = \sum_{j=1}^{n} z'_j$$

in which, z' is the normalized soft-labels of non-target classes from the Standard classifier, and z is the soft-labels of known classes from the Raw Zer-Shot Classifier. Note that, the given class is 'known' (target) by the standard classifier and is 'unknown' to the Raw Zero-Shot Classifier. Hence, the Amalgam Metric captures the existence of some unique features learned which are specific to a class which in-turn changes the Amalgam Proportion between Raw Zero-Shot Classifier and Standard Classifier. A higher AM score corresponds to a classifier preferring to learn special features of a class over general features present across the distribution. In other words, a lower AM score corresponds to a classifier preferring to learn general features over special features. A non-zero AM score thus verifies the existence of the unique special features to a class which are learned by training the classifier on that specific class.

4 EXPERIMENTAL DESIGN AND RESULTS

Considered Datasets: We conducted experiments on three diverse datasets to evaluate the representation of the neural networks. We used Fashion MNIST (F-MNIST) (Xiao et al., 2017), CIFAR-10 (Krizhevsky et al., 2009) and a customised Sub-Imagenet (Sub) dataset for our evaluations. The details of the customised Sub-Imagenet dataset is mentioned in Appendix B. Note that, the number of samples (7000 for Fashion MNIST, 6000 for CIFAR-10, and roughly 13500 samples for Sub-Imagenet dataset) in the assumed unknown class differ with the dataset. We use the samples from both training and testing dataset for the 'unknown' class for evaluation because we exclude these samples in the training process.

Considered Classifiers: We evaluated different architectures for different datasets. For the Fashion MNIST datasets, we chose to evaluate Multi-Layer Perceptron (MLP), and a shallow Convolution Neural Network (ConvNet). For the CIFAR-10 dataset, LeNet (a simpler architecture which is a historical mark) (LeCun et al., 1998), VGG (a previous state-of-the-art architecture which is a historical mark) (Simonyan & Zisserman, 2014), All Convolutional Network (AllConv) (an architecture without max pooling and fully-connected layers) (Springenberg et al., 2014), Network in Network (NIN) (an architecture which uses micro neural networks instead of linear filters) (Lin et al., 2013), Residual Networks (ResNet) (an architecture based on skip connections) (He et al., 2016), Wide Residual Networks (WideResNet) (an architecture which also expands in width) (Zagoruyko & Komodakis, 2016), DenseNet (an architecture which is a logical extension of ResNet) (Huang et al., 2017), and Capsule Networks (CapsNet) (a recently proposed completely different architecture based on dynamic routing and capsules) (Sabour et al., 2017). For our Sub-Imagenet dataset, we chose InceptionV3 (Szegedy et al., 2016), and ResNet-50 (He et al., 2016). Details about the Standard and Raw Zero-Shot Classifiers are mentioned in Appendix C.

Considered Adversarial Defences: We also evaluated the representation quality of some of the adversarial defences for CIFAR-10 dataset, such as Feature Squeezing (FS) (Xu et al., 2017), Spatial Smoothing (SS) (Xu et al., 2017) , Label Smoothing (LS) (Hazan et al., 2016), Thermometer Encoding (TE) (Buckman et al., 2018), and Adversarial Training (AT) (Madry et al., 2018). We also evaluate classifiers trained with augmented dataset having Gaussian Noise of $\sigma = 1.0$ (G Aug). Details about the adversarial defences are mentioned in Appendix D. For a discussion about the performance of adversarial defences in general, please refer to Athalye et al. (2018).

Considered Attacks: We also evaluated all our standard vanilla classifiers against well-known adversarial attacks such as Fast Gradient Method (FGM) (Goodfellow et al., 2014), Basic Iterative Method (BIM) (Kurakin et al., 2016), Projected Gradient Descent Method (PGD) (Madry et al., 2018), DeepFool (DF) (Moosavi-Dezfooli et al., 2016), and NewtonFool (NF) (Jang et al., 2017). Details about the adversarial attacks are mentioned in Appendix E.

Architecture	DBM	AM
For Fashion MNIST Dataset		
MLP	0.51±0.09	670.71±81.79
ConvNet	0.47±0.10	683.55±76.39
For Sub-Imagenet Dataset		
InceptionV3	0.56±0.07	1335.65±31.83
ResNet-50	0.55±0.15	1311.97±37.59

Architecture	DBM	AM
For CIFAR-10 Dataset		
LeNet	0.54±0.04	473.97±91.53
VGG	0.61±0.12	645.86±15.19
AllConv	0.64±0.08	634.04±22.01
NIN	0.63±0.09	646.04±16.40
ResNet	0.64±0.13	654.90±6.40
DenseNet	0.61±0.14	658.21±4.05
WideResNet	0.58±0.15	660.00±3.60
CapsNet	0.43±0.03	385.85±83.77

Table 1: Mean and Standard Deviation of DBM and AM Scores for vanilla Raw Zero-Shot Classifiers.

Experimental Results For Vanilla Classifiers: Table 1 shows the results of our metrics (DBM and AM) for vanilla classifiers. Note that, we use mean across all the metric values for N classes of the dataset to be characteristic metric value for an architecture. To enable the visualisation of DBM, we plot a projection of all the points in the decision space of unknown class ($N - 1$ dimensions) into two-dimensional space (Appendix F). Similarly, we can also visualise AM, in the form of histograms of soft-labels for the classifiers (Appendix G). Table 1 reveals that for CIFAR-10 dataset, CapsNet possesses the best representation quality amongst all classifiers examined as it has the least (best) score in both of our metrics. At the same time, LeNet has the second-best representation quality. Moreover, other architectures possess similar representation quality. Also for Sub-Imagenet dataset,

both architectures (InceptionV3 and ResNet-50) are equally clustered and predict the Amalgam Proportion similarly. However, ResNet-50 has marginally better representation quality than the InceptionV3 as it has better scores for both of our metrics. Similarly, for Fashion MNIST dataset, both architectures (MLP and ConvNet) have a similar quality of representation. While ConvNet seems marginally superior to the MLP in terms of clustering the unknown classes more tightly (suggested by DBM), MLP seems marginally superior to predict the Amalgam Proportion better than the ConvNet (suggested by AM).

5 Link Between Representation Quality And Adversarial Defences

Davies–Bouldin Metric (DBM)							
Architecture	**No Defence**	**Gaussian Augmentation**		**Label Smoothing**		**Adversarial Training**	
LeNet	0.54±0.04	0.56±0.04	(0.00)	0.43±0.02	(0.00)	0.32±0.04	(0.00)
VGG	0.61±0.12	0.63±0.12	(0.07)	0.55±0.10	(0.00)	0.47±0.07	(0.00)
AllConv	0.64±0.08	0.66±0.11	(0.27)	0.48±0.05	(0.00)	0.50±0.06	(0.00)
NIN	0.63±0.09	0.64±0.11	(0.17)	0.52±0.08	(0.00)	0.43±0.06	(0.00)
ResNet	0.64±0.13	0.63±0.14	(0.09)	0.54±0.11	(0.00)	0.43±0.07	(0.00)
DenseNet	0.61±0.14	0.60±0.15	(0.05)	0.55±0.13	(0.00)	0.50±0.10	(0.02)
WideResNet	0.58±0.15	0.59±0.15	(0.58)	0.46±0.09	(0.00)	0.61±0.10	(0.13)
CapsNet	0.22±0.01	0.23±0.01	(0.00)	0.18±0.01	(0.00)	0.15±0.02	(0.00)
Architecture	**No Defence**	**Feature Squeezing**		**Spatial Smoothing**		**Thermometer Encoding**	
LeNet	0.54±0.04	0.54±0.04	(0.38)	0.50±0.03	(0.01)	0.52±0.04	(0.09)
VGG	0.61±0.12	0.62±0.11	(0.14)	0.63±0.09	(0.52)	0.65±0.05	(0.27)
AllConv	0.64±0.08	0.64±0.08	(0.20)	0.63±0.08	(0.66)	0.67±0.05	(0.12)
NIN	0.63±0.09	0.63±0.09	(0.13)	0.65±0.06	(0.39)	0.65±0.06	(0.14)
ResNet	0.64±0.13	0.65±0.13	(0.20)	0.66±0.11	(0.61)	0.71±0.06	(0.02)
DenseNet	0.61±0.14	0.62±0.12	(0.16)	0.64±0.11	(0.57)	0.69±0.09	(0.00)
WideResNet	0.58±0.15	0.59±0.14	(0.13)	0.62±0.11	(0.51)	0.66±0.08	(0.02)
CapsNet	0.22±0.01	0.22±0.01	(0.00)	0.21±0.01	(0.09)	0.20±0.02	(0.03)
Amalgam Metric (AM)							
Architecture	**No Defence**	**Gaussian Augmentation**		**Label Smoothing**		**Adversarial Training**	
LeNet	115.97±36.92	84.00±26.39	(0.03)	177.08±97.77	(0.10)	29.93±16.06	(0.00)
VGG	270.76±186.04	287.75±122.58	(0.75)	579.05±121.89	(0.00)	218.47±100.50	(0.44)
AllConv	150.35±39.16	153.73±65.96	(0.90)	395.28±143.78	(0.00)	188.66±67.98	(0.16)
NIN	186.14±97.41	222.68±104.12	(0.03)	503.32±145.15	(0.00)	86.45±17.60	(0.01)
ResNet	233.84±109.08	266.61±124.12	(0.17)	592.57±119.06	(0.00)	86.71±46.24	(0.00)
DenseNet	314.93±130.50	303.04±120.54	(0.70)	629.48±131.86	(0.00)	187.34±71.01	(0.04)
WideResNet	417.37±180.78	443.95±157.46	(0.13)	586.84±132.92	(0.00)	365.29±199.90	(0.13)
CapsNet	96.96±38.59	111.46±56.69	(0.07)	100.01±42.72	(0.54)	54.48±20.38	(0.00)
Architecture	**No Defence**	**Feature Squeezing**		**Spatial Smoothing**		**Thermometer Encoding**	
LeNet	115.97±36.92	116.85±37.42	(0.37)	72.13±20.02	(0.00)	272.03±80.86	(0.00)
VGG	270.76±186.04	271.42±184.10	(0.78)	183.06±128.16	(0.02)	510.39±85.82	(0.00)
AllConv	150.35±39.16	149.47±38.17	(0.50)	179.44±68.03	(0.14)	537.48±74.51	(0.00)
NIN	186.14±97.41	185.82±100.53	(0.92)	148.72±100.69	(0.00)	516.72±92.20	(0.00)
ResNet	233.84±109.08	226.19±105.21	(0.06)	199.64±99.87	(0.14)	531.54±80.03	(0.00)
DenseNet	314.93±130.50	319.33±136.19	(0.68)	246.08±99.05	(0.09)	585.38±56.48	(0.00)
WideResNet	417.37±180.78	402.62±185.48	(0.04)	207.62±131.18	(0.00)	646.85±10.66	(0.00)
CapsNet	96.96±38.59	96.95±38.57	(0.82)	84.02±31.37	(0.03)	280.39±58.42	(0.00)

Table 2: Mean and Standard Deviation of DBM and AM values for different Raw Zero-Shot Classifiers with and without the adversarial defences on CIFAR-10. Values in the parentheses are p-values of the paired samples t-test between the metric values of defences and those without defences.

Table 2 shows the results of our metrics (DBM and AM) for vanilla classifiers and classifiers employed with a variety of adversarial defences for improving the robustness of vanilla classifiers for CIFAR-10. We also analyse the statistical relevance of the change in metric values due to introduction of adversarial defences. A paired samples t-test (David & Gunnink, 1997) was conducted for our metrics' distributions (DBM and AM) of Vanilla Classifiers (without adversarial defence), and Adversarially defended classifiers (Table 2) to test the significance in the change in metric values due to Adversarial Defences. The Null hypothesis of paired samples t-test assumes that the true mean difference between the distributions is equal to zero. Based on the results (Table 2) adversarial defences, 'in general', tend to improve the representation quality of the neural networks evaluated using Amalgam Proportion. It does so by either by creating a more dense cluster of the soft-labels (suggested by DBM) or learning more general/special features (suggested by AM), or both.

Raw DBM Score values for weaker defences such as G Aug, FS, SS and TE lie within the standard deviation of vanilla classifiers suggesting that they affect minimally in clustering the Amalgam Proportion of unknown classes. At the same time, DBM Score values for defences such as LS and AT are noticeably lower than vanilla classifiers suggesting they try to form a denser cluster of Amalgam Proportion compared to the vanilla classifiers. Thus a better association of available features is observed for the more robust defences. From the perspective of AM Score values, the results suggest that LS favours learning special features belonging to a class while AT favours to learn more general features. Interestingly, a general low p-value for the paired samples t-test is observed for the adversarial defences, which suggests that underlying representation of adversarial defences differ from the vanilla classifiers with high statistical relevance.

6 Link Between Representation Quality And Adversarial Attacks

Architecture	DBM with Mean L_2 Score					AM with Mean L_2 Score				
	FGM	BIM	PGD	DF	NF	FGM	BIM	PGD	DF	NF
Fashion MNIST										
MLP	-0.20 (0.58)	-0.17 (0.64)	-0.17 (0.64)	-0.04 (0.91)	-0.02 (0.97)	0.82 (0.00)	0.26 (0.47)	0.26 (0.47)	0.83 (0.00)	0.84 (0.00)
ConvNet	-0.24 (0.50)	-0.30 (0.40)	-0.30 (0.40)	-0.26 (0.46)	-0.22 (0.55)	0.83 (0.00)	-0.07 (0.84)	-0.09 (0.80)	0.81 (0.00)	0.82 (0.00)
CIFAR-10										
LeNet	-0.18 (0.61)	-0.70 (0.02)	-0.66 (0.04)	-0.51 (0.13)	-0.36 (0.31)	0.93 (0.00)	0.32 (0.36)	0.25 (0.49)	0.81 (0.00)	0.89 (0.00)
VGG	-0.62 (0.06)	-0.21 (0.55)	-0.20 (0.58)	-0.52 (0.13)	-0.63 (0.05)	0.71 (0.02)	-0.04 (0.91)	-0.07 (0.85)	0.87 (0.00)	0.74 (0.01)
AllConv	-0.31 (0.39)	-0.56 (0.09)	-0.54 (0.11)	-0.10 (0.78)	-0.30 (0.41)	0.67 (0.03)	0.42 (0.23)	0.41 (0.24)	0.94 (0.00)	0.73 (0.02)
NIN	-0.56 (0.09)	-0.57 (0.08)	-0.57 (0.09)	-0.42 (0.22)	-0.43 (0.21)	0.78 (0.01)	0.84 (0.00)	0.84 (0.00)	0.96 (0.00)	0.89 (0.00)
ResNet	-0.52 (0.12)	-0.76 (0.01)	-0.76 (0.01)	-0.47 (0.17)	-0.51 (0.13)	0.35 (0.32)	0.57 (0.09)	0.57 (0.09)	0.79 (0.01)	0.83 (0.00)
DenseNet	-0.62 (0.06)	-0.50 (0.14)	-0.49 (0.15)	-0.16 (0.65)	-0.22 (0.55)	0.53 (0.11)	0.78 (0.01)	0.78 (0.01)	0.78 (0.01)	0.84 (0.00)
WideResNet	-0.68 (0.03)	-0.75 (0.01)	-0.75 (0.01)	-0.68 (0.03)	-0.75 (0.01)	0.66 (0.04)	0.68 (0.03)	0.68 (0.03)	0.78 (0.01)	0.68 (0.03)
CapsNet	-0.71 (0.02)	-0.45 (0.19)	-0.49 (0.15)	-0.39 (0.26)	-0.48 (0.17)	0.98 (0.00)	0.69 (0.03)	0.73 (0.02)	-0.17 (0.63)	0.47 (0.17)
Sub-Imagenet										
InceptionV3	-0.76 (0.01)	-0.52 (0.13)	-0.52 (0.13)	-0.35 (0.32)	-0.50 (0.14)	0.75 (0.01)	0.14 (0.70)	0.14 (0.70)	0.28 (0.44)	0.25 (0.49)
ResNet-50	-0.34 (0.34)	-0.12 (0.74)	-0.12 (0.74)	-0.54 (0.10)	-0.25 (0.48)	0.82 (0.00)	0.31 (0.39)	0.31 (0.39)	0.51 (0.13)	0.50 (0.15)

Table 3: Pearson correlation coefficient of DBM and AM with Mean L_2 Score of Adversarial Attacks for each vanilla classifier and attack pair. Values in the parentheses are p-values of the Pearson correlation test.

Since, the results in Table 2, suggests a link between the representation quality and the adversarial defences as discussed above. It is intuitive to assume that there also exists a link between the representation quality and the adversarial attacks. To evaluate the statistical relevance of this link between representation quality evaluated using Amalgam Proportion and adversarial attacks, we conducted a Pearson correlation coefficient test (Freedman et al., 2007) of our metrics (DBM and AM) of the vanilla classifiers with adversarial attacks. The Pearson correlation analysis of our metrics suggests a relationship between our metrics and the adversarial attacks in general.

We use the analysis of adversarial attacks in the form of Mean L_2 Score (L_2 difference between the original sample and the adversarial one) to compute the correlation (Moosavi-Dezfooli et al., 2016). The Pearson correlation coefficients of our metrics (DBM and AM) with Mean L_2 Score is shown in Table 3 for every architecture and attacks. Moreover, these Pearson relationships between our metrics and Mean L_2 Score can also be visualised (Appendix H). We also analyse the impact of adversarial attacks on the correct class soft-label (Appendix I).

We do observe some anomalies in the Pearson correlation coefficient of AM with BIM and PGD attacks for the ConvNet, and VGG network and DeepFool for CapsNet. These anomalies are studied in detail (Appendix H) to understand their existence. Our extended analysis suggests that these anomalies exist due to abnormal behaviour of some classes. On careful study, we note that for all the classes of VGG network BIM and PGD have similar AM Scores, while at the same time the Mean L2 Score differs for across classes. We observe that for the CapsNet, the Airplane class had abnormally low Mean L_2 score suggesting less perturbation which was abnormal compared to the other classes in the same setting. These anomalies further suggest that baseline Amalgam Proportion for some of the classes differ. However, the study of these representation qualities of 'individual' classes and their effect overall representation quality is beyond the scope of the current article and hence, left as future work.

7 General Discussion On Representation Quality

On carefully observing the metric values (Tables 1, 2, and 3), we found that our assessment of representation quality using Amalgam Proportion also explains some of the propositions by other researchers, we highlight some of our key findings below,

Does a model with high capacity will have a better representation quality? Our results reveal that a deeper network which generally has a higher capacity (Madry et al., 2018) does not necessarily correspond to have a better representation quality of the input features. As CapsNet and LeNet, which are much shallower than the other deeper networks, are shown to have superior representation quality than other deeper networks (Table 2).

Why CapsNet has better representation quality than other deeper networks? We observe that Capsule Networks (CapsNet) has the best representation amongst other neural networks (Table 2). Our results suggest that CapsNet not only produces a denser cluster for Amalgam Proportion but also learns more general features it might be because of the dynamical nature (routing) of the CapsNet. Thus our results call for a more in-depth investigation of Capsule Networks and their representation quality.

How does augmenting the dataset with Gaussian Noise affect the representation quality? We observe that Gaussian Augmentation degrades the representation quality of all the classifiers (Table 2). This supports our intuition (Section 3), as adding Gaussian noise to the images subdue the features of the image by blurring making the classifier harder to interpret these features. Consequently, a weaker association of the representation with these features is observed through the perspective of Amalgam Proportion.

How does Label Smoothing improve the representation quality? Our results corroborate the analysis in Müller et al. (2019) that Label Smoothing (LS) encourages the representations to group in tight, equally distant clusters. The raw metric values from our experiments for LS suggests that classifiers employed with LS do form a tighter cluster in soft-label space (as suggested by DBM) (Table 2). At the same time, LS also favours the classifiers to learn special features belonging to a class (as suggested by AM).

Do adversarial defences which work on the principle of obfuscated gradients affect representation quality? Since, some adversarial defences rely on obfuscating gradients (Athalye et al., 2018) such as Feature Squeezing, Spatial Smoothing, and Thermometer Encoding, they fail to improve the representation quality of the classifiers (suggested by DBM). At the same time, more robust adversarial defences like Adversarial Training which do not rely on obfuscating gradients have better representation quality. Hence, adversarial defences can be evaluated using our metrics to analyse, if an adversarial defence improves the robustness of the classifier by improving the representation quality of the classifier or rely on some other criterion.

8 Conclusions

In this article, we propose a novel Zero-Shot learning-based method, entitled Raw Zero-Shot, to assess the representation of the several neural networks. In order to assess the representation, two associated metrics are formally defined based on different hypotheses of representation quality. Results from the experiments reveal that classifiers employed with adversarial defences not only decrease the attack accuracy as presumed but also improve the representation quality of the classifiers as evaluated by our proposed metrics (DBM and AM). Further, adversarial defences, have a low p-value in the paired samples t-test when compared to vanilla classifiers in general, suggesting that representation quality is significantly affected by various adversarial defences. Moreover, a high Pearson correlation coefficient and low p-value of the Pearson correlation test between the proposed metrics and the adversarial attacks suggest a link between the representation quality and the adversarial attacks. Our experimental results suggest that CapsNet (dynamic routing network) has the best representation quality amongst classifiers which calls for a more in-depth investigation of Capsule Networks. Hence, the proposed Raw Zero-Shot was able to assess and understand the representation quality from the perspective of unknown classes of different neural networks' architectures, along with the adversarial defences and link this representation quality of the neural networks with adversarial attacks and defences. It also opens up new possibilities of using representation quality for both evaluation (i.e. as a quality assessment) and the development (e.g. as a loss function) of neural networks.

REFERENCES

Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2927–2936, 2015.

Anish Athalye and Ilya Sutskever. Synthesizing robust adversarial examples. In *ICML*, 2018.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *European Conference on Computer Vision*, pp. 730–746. Springer, 2016.

Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. *ICLR*, 2018.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57. IEEE, 2017.

Daniel Casasanto. similarity and proximity: When does close in space mean close in mind? *Memory & Cognition*, 36(6):1047–1056, 2008.

Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Li Chen, Michael E Kounavis, and Duen Horng Chau. Keeping the bad guys out: Protecting and vaccinating deep learning with jpeg compression. *arXiv preprint arXiv:1705.02900*, 2017.

Herbert A David and Jason L Gunnink. The paired t test under artificial pairing. *The American Statistician*, 51(1):9–12, 1997.

David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 1979.

Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016.

Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, Pascal Frossard, and Stefano Soatto. Empirical study of the topology and geometry of deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3762–3770, 2018.

David Freedman, Robert Pisani, and Roger Purves. Statistics (international student edition). *Pisani, R. Purves, 4th edn. WW Norton & Company, New York*, 2007.

Yanwei Fu, Yongxin Yang, Tim Hospedales, Tao Xiang, and Shaogang Gong. Transductive multi-label zero-shot learning. *arXiv preprint arXiv:1503.07790*, 2015.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In *ICLR*, 2018.

Tamir Hazan, George Papandreou, and Daniel Tarlow. *Perturbations, Optimization, and Statistics*. MIT Press, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.

Ruitong Huang, Bing Xu, Dale Schuurmans, and Csaba Szepesvári. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034*, 2015.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pp. 125–136, 2019.

Uyeong Jang, Xi Wu, and Somesh Jha. Objective metrics and gradient descent algorithms for adversarial examples in machine learning. In *Proceedings of the 33rd Annual Computer Security Applications Conference*, pp. 262–277. Acm, 2017.

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

Joseph B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 951–958. IEEE, 2009.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

Xingjun Ma, Bo Li, Yisen Wang, Sarah M Erfani, Sudanthi Wijewickrema, Grant Schoenebeck, Dawn Song, Michael E Houle, and James Bailey. Characterizing adversarial subspaces using local intrinsic dimensionality. *arXiv preprint arXiv:1801.02613*, 2018.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582, 2016.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 86–94. IEEE, 2017.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, Pascal Frossard, and Stefano Soatto. Robustness of classifiers to universal perturbations: A geometric perspective. In *International Conference on Learning Representations*, 2018.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems*, pp. 4696–4705, 2019.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427–436, 2015.

Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.1.0. *CoRR*, 1807.01069, 2018. URL `https://arxiv.org/pdf/1807.01069`.

Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*, 2013.

Nicolas Papernot, Patrick McDaniel, Xi Wu, Somesh Jha, and Ananthram Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 582–597. IEEE, 2016.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

Sara Sabour, Yanshuai Cao, Fartash Faghri, and David J Fleet. Adversarial manipulation of deep representations. *arXiv preprint arXiv:1511.05122*, 2015.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. In *Advances in neural information processing systems*, pp. 3856–3866, 2017.

Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1528–1540. Acm, 2016.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *ICLR*, 2018.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, 2016.

Christian et al. Szegedy. Intriguing properties of neural networks. In *In ICLR*. Citeseer, 2014.

Guanhong Tao, Shiqing Ma, Yingqi Liu, and Xiangyu Zhang. Attacks meet interpretability: Attribute-steered detection of adversarial samples. In *Advances in Neural Information Processing Systems*, pp. 7717–7728, 2018.

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

Laura Thesing, Vegard Antun, and Anders C Hansen. What do ai algorithms actually learn?-on false structures in deep learning. *arXiv preprint arXiv:1906.01478*, 2019.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*, 2017.

Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

Jonathan Uesato, Brendan O'Donoghue, Pushmeet Kohli, and Aaron Oord. Adversarial risk and the dangers of evaluating against weak attacks. In *International Conference on Machine Learning*, pp. 5032–5041, 2018.

Danilo Vasconcellos Vargas and Shashank Kotyan. Robustness assessment for adversarial machine learning: Problems, solutions and a survey of current neural networks and defenses. *arXiv preprint arXiv:1906.06026*, 2019.

Danilo Vasconcellos Vargas and Jiawei Su. Understanding the one-pixel attack: Propagation maps and locality analysis. *arXiv preprint arXiv:1902.02947*, 2019.

Caren M Walker and Alison Gopnik. Toddlers infer higher-order relational principles in causal learning. *Psychological science*, 25(1):161–169, 2014.

Caren M Walker, Sophie Bridgers, and Alison Gopnik. The early emergence and puzzling decline of relational reasoning: Effects of knowledge and search on inferring abstract concepts. *Cognition*, 156:30–40, 2016.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017. URL `http://arxiv.org/abs/1708.07747`.

Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155*, 2017.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE international conference on computer vision*, pp. 4166–4174, 2015.

Ziming Zhang and Venkatesh Saligrama. Zero-shot recognition via structured prediction. In *European conference on computer vision*, pp. 533–548. Springer, 2016.

A On Links Of Representation Quality With Adversarial Attacks And Defences

We hypothesise based on our experiments and results that the cause of the links of representation quality is due to the presence of a bias introduced in the training of neural networks. We call this bias as Dataset Bias and define it as a bias towards the classes and data distribution present in a dataset. It is already proven theoretically that it is possible to separate any number of classes, provided enough samples are evaluated. However, this separation only exists inside the underlying data distribution and classes of the evaluated samples. With the introduction of noise in the underlying data distribution, this separation of classes is not valid anymore as the distribution is substantially modified. The area related to Zero-Shot Learning and Transfer Learning, investigate this bias by introducing unknown class samples at the time of inferring while in the field of adversarial machine learning, the same bias is studied by introducing noisy samples.

B Details About Customised Sub-Imagenet Dataset

Super-Classes	Training Images	Testing Images	Corresponding Imagenet (ILSVRC 2012) Classes
Automobile	12981	500	407, 468, 555, 627, 654, 779, 817, 802, 866, 867
Ball	12971	500	429, 430, 522, 574, 722, 746, 768, 805, 852, 890
Bird	12990	500	7, 8, 9, 16, 22, 23, 24, 84, 94, 100
Dog	12904	500	205, 206, 207, 208, 209, 210, 211, 212, 213, 214
Feline	13000	500	283, 284, 285, 286, 287, 288, 289, 290, 291, 292
Fruit	12986	500	948, 949, 950, 951, 952, 953, 954, 955, 956, 957
Insect	12985	500	300, 301, 302, 303, 304, 305, 306, 307, 308, 309
Snake	12758	500	55, 56, 57, 58, 59, 60, 61, 62, 63, 64
Primate	12979	500	365, 366, 367, 368, 369, 370, 371, 372, 373, 374
Vegetable	12815	500	935, 936, 937, 938, 939, 943, 944, 945, 946, 947
Total	***129359***	***5000***	

Table 4: Description of Super-Classes used in the Sub-ImageNet.

Sub-Imagenet is a subset of the Imagenet (ILSVRC 2012) (Russakovsky et al., 2015) dataset. It is intuitive for us to expect that as the number of classes (N) grows, the decision boundary will become more complicated, causing the classifier to smoothen the representation (Amalgam Proportion) more. Therefore, to prevent this bias, we grouped a subset of 100 existing semantically alike ImageNet classes into 10 distinct super-classes, as described in Table 4. Our Sub-Imagenet dataset has some desired characteristics for our experiments which are also similar to the CIFAR-10 dataset. These features are:

1. It is relatively balanced dataset as other datasets used in the experiments. The dataset has a mean of 12937 training images with a standard deviation of 80 images. All super-classes have relatively the same number of images with a minimum of 12758 images for super-class Snake and a maximum of 13000 for super-class Feline. Thus, the samples in the unknown class in our experiments remain relative same.
2. Type of super-classes is similar to CIFAR-10, having six animal classes and four non-animal classes.
3. Abstract Relationships between super-classes also exists similar to the CIFAR-10. The CIFAR-10 have a Cat-Dog and Automobile-Truck relationships in which they are semantically similar. Similarly, our Sub-imagenet also exhibits Dog-Feline and Fruit-Vegetable relationships. These abstract relationships are essential to validate our hypothesis of Amalgam Proportion.

C Details About Standard and Raw Zero-Shot Classifiers

Architecture	Standard Classifier	Raw Zero-Shot Classifiers (excluding one class)									
		Fashion MNIST									
		T-Shirt	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	AnkleBoot
MLP	88.26% (0.3283)	90.61% (0.2636)	87.73% (0.3440)	91.30% (0.2498)	89.18% (0.2986)	90.96% (0.2606)	88.02% (0.3263)	93.53% (0.1884)	88.57% (0.3241)	87.58% (0.3446)	88.26% (0.3328)
ConvNet	90.47% (0.3280)	91.90% (0.2940)	89.43% (0.3556)	90.64% (0.3193)	90.67% (0.3193)	91.70% (0.2933)	88.86% (0.3675)	94.18% (0.2336)	90.20% (0.3309)	90.02% (0.3496)	90.52% (0.3360)
		CIFAR-10									
		Airplane	Automobile	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
LeNet	73.86% (0.8223)	74.52% (0.7858)	74.91% (0.7945)	77.33% (0.7075)	79.34% (0.6642)	77.13% (0.7223)	78.32% (0.6990)	75.95% (0.7569)	75.90% (0.7615)	74.46% (0.7946)	75.67% (0.7735)
VGG	92.65% (0.5467)	92.85% (0.5370)	92.06% (0.5600)	93.27% (0.5026)	94.43% (0.4660)	92.73% (0.5241)	93.85% (0.4890)	92.82% (0.5400)	92.64% (0.5301)	92.41% (0.5415)	92.60% (0.5442)
AllConv	87.93% (0.6823)	88.63% (0.6501)	86.75% (0.7601)	89.07% (0.6313)	90.04% (0.5917)	87.98% (0.6737)	89.81% (0.6147)	87.38% (0.6778)	87.67% (0.7085)	87.35% (0.7034)	87.00% (0.7205)
NIN	90.45% (0.5020)	90.92% (0.4752)	90.55% (0.4974)	91.04% (0.4666)	92.84% (0.3962)	91.02% (0.4773)	92.01% (0.4315)	90.77% (0.4646)	90.13% (0.5155)	90.51% (0.5041)	90.26% (0.5068)
ResNet	92.58% (0.4685)	92.82% (0.4494)	92.67% (0.4824)	93.42% (0.4328)	94.25% (0.3673)	92.48% (0.4724)	93.75% (0.4119)	92.58% (0.4641)	92.73% (0.4636)	92.53% (0.4740)	92.95% (0.4509)
DenseNet	93.97% (0.3643)	94.27% (0.3540)	94.08% (0.3644)	94.20% (0.3341)	95.86% (0.2702)	93.68% (0.3924)	95.15% (0.3054)	94.11% (0.3804)	93.81% (0.3841)	94.07% (0.3627)	94.32% (0.3656)
WideResNet	95.02% (0.2705)	94.90% (0.2808)	94.96% (0.2872)	94.96% (0.2761)	96.57% (0.2005)	94.67% (0.3001)	95.98% (0.2318)	94.73% (0.2943)	94.71% (0.2842)	95.01% (0.2844)	94.96% (0.2842)
CapsNet*	74.74% (0.2017)	75.20% (0.1953)	74.43% (0.2022)	76.74% (0.1878)	77.35% (0.1875)	76.86% (0.1877)	77.33% (0.1838)	75.92% (0.1949)	74.71% (0.2004)	74.92% (0.2001)	74.43% (0.2015)
		Sub-Imagenet									
		Automobile	Ball	Bird	Dog	Feline	Fruit	Insect	Snake	Primate	Vegetable
InceptionV3	94.06% (0.1907)	94.22% (0.1968)	95.00% (0.1686)	93.91% (0.2787)	93.93% (0.1977)	93.60% (0.1997)	95.11% (0.1702)	94.60% (0.2031)	94.40% (0.1870)	94.33% (0.1937)	94.66% (0.3634)
ResNet-50	92.58% (0.2590)	91.04% (1.7212)	91.08% (1.0540)	92.91% (0.3529)	92.15% (0.3986)	31.40% (1.8681)	95.64% (0.1647)	94.17% (0.4806)	92.66% (0.2975)	94.02% (0.2278)	94.68% (0.2891)

Table 5: Classifier Accuracy (and loss value) on test dataset of the learned classes for different architectures.

Table 5 shows the classifier accuracy and corresponding loss value on the test dataset of the learned classes. All the classifiers except CapsNet are trained using standard cross-entropy loss. In contrast, CapsNet uses 'margin loss' (Sabour et al., 2017) to train the parameters of the network. As Raw Zero-Shot Classifier, forcefully excludes the images of a class for training, we get the accuracy of the Raw Zero-Shot Classifier on $N - 1$ learned classes of the dataset.

D Details About Adversarial Defences

Defence	Parameters
Gaussian Augmentation (G Aug)	$\sigma = 1.0$
Feature Squeezing (FS) **Spatial Smoothing (SS)**	bit depth $= 5$ window size $= 3$
Label Smoothing (LS) **Thermometer Encoding (TE)**	max value $= 0.9$ num space $= 16$
Adversarial Training (AT)	Attack: Projected Gradient Descent (PGD) Attack Parameters: norm $= L_\infty$, $\epsilon = 8$, $\epsilon_{\text{step}} = 2$, iterations $= 10$

Table 6: Description of Adversarial Defence Parameters

All the adversarial defences used in the article have been evaluated using Adversarial Robustness 360 Toolbox (ART v1.2.0) Nicolae et al. (2018). Table 6 describes the defence parameters used for the evaluated adversarial defences. Table 7 shows the classifier accuracy and corresponding loss value on the test dataset of the learned classes for various adversarial defences. All the classifiers except CapsNet use standard cross-entropy loss, while CapsNet uses margin loss (Sabour et al., 2017). As Raw Zero-Shot Classifier, forcefully excludes the images of a class for training, we get the accuracy of the Raw Zero-Shot Classifier on $N - 1$ learned classes of the dataset.

E Details About Adversarial Attacks

All the adversarial attacks used in the article have been evaluated using Adversarial Robustness 360 Toolbox (ART v1.2.0) Nicolae et al. (2018). We evaluated the test samples of Fashion MNIST, CIFAR-10 and Sub Imagenet datasets for the adversarial attacks on standard classifiers. We fixed the parameters of the attacks evaluated, and Table 8 describes the attack parameters used for the evaluated adversarial attacks. Table 9 shows the adversarial accuracy and Mean L_2 Score for each classifier and adversarial attack pair. Here, Adversarial Accuracy corresponds to the percentage of adversarial images misclassified by a standard classifier. While Mean L_2 Score corresponds to the Mean L_2 norm of the perturbation in the adversarial image.

Architecture	Standard Classifier	Raw Zero-Shot Classifiers (excluding one class) Airplane	Automobile	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
Gaussian Augmentation (G Aug)											
LeNet	76.05% (0.7795)	76.35% (0.7736)	73.73% (0.8315)	77.15% (0.7151)	79.43% (0.6736)	76.57% (0.7396)	77.11% (0.7175)	76.20% (0.7579)	75.18% (0.7836)	75.91% (0.7763)	75.27% (0.7999)
VGG	92.64% (0.5276)	93.01% (0.5123)	92.58% (0.5392)	93.63% (0.4749)	94.63% (0.4425)	92.65% (0.5289)	94.21% (0.4566)	92.78% (0.5272)	92.77% (0.5074)	92.53% (0.5505)	92.96% (0.5291)
AllConv	87.74% (0.7554)	88.68% (0.6927)	86.95% (0.7693)	89.00% (0.6644)	90.33% (0.5983)	88.58% (0.6783)	90.27% (0.6190)	86.72% (0.8040)	88.12% (0.7196)	87.54% (0.7415)	86.88% (0.8059)
NIN	91.14% (0.4845)	91.31% (0.4728)	91.31% (0.4910)	91.96% (0.4600)	93.64% (0.3706)	91.31% (0.4791)	92.30% (0.4249)	91.04% (0.4803)	91.20% (0.4926)	91.02% (0.4811)	91.05% (0.4790)
ResNet	93.02% (0.4340)	93.33% (0.4232)	92.77% (0.4151)	93.57% (0.3862)	94.81% (0.3136)	93.13% (0.4103)	94.20% (0.3320)	92.76% (0.4058)	93.06% (0.4214)	92.63% (0.4466)	92.94% (0.4281)
DenseNet	94.56% (0.2963)	94.65% (0.2949)	94.81% (0.3136)	94.94% (0.2710)	96.08% (0.2036)	94.23% (0.3092)	95.80% (0.2255)	94.36% (0.2969)	94.47% (0.3068)	94.34% (0.3127)	95.01% (0.2850)
WideResNet	95.05% (0.2580)	95.24% (0.2529)	95.20% (0.2701)	95.42% (0.2406)	96.55% (0.1848)	95.22% (0.2532)	96.08% (0.2078)	.% (0.)	94.86% (0.2673)	95.14% (0.2717)	95.25% (0.2485)
CapsNet	76.29% (0.1938)	77.08% (0.1833)	76.00% (0.1932)	78.20% (0.1771)	76.87% (0.1913)	78.21% (0.1790)	79.52% (0.1712)	77.21% (0.1864)	76.73% (0.1879)	75.87% (0.1926)	76.28% (0.1877)
Feature Squeezing (FS)											
LeNet	73.92% (0.8218)	74.45% (0.7856)	74.94% (0.7945)	77.26% (0.7077)	79.38% (0.6630)	76.94% (0.7228)	78.22% (0.6987)	76.05% (0.7564)	75.95% (0.7608)	74.36% (0.7954)	75.61% (0.7736)
VGG	92.65% (0.5470)	92.77% (0.5377)	92.06% (0.5614)	93.25% (0.5026)	94.42% (0.4662)	92.67% (0.5250)	93.87% (0.4890)	92.88% (0.5402)	92.66% (0.5303)	92.52% (0.5424)	92.57% (0.5449)
AllConv	87.93% (0.6812)	88.63% (0.6488)	86.78% (0.7609)	89.00% (0.6310)	90.06% (0.5915)	87.87% (0.6744)	89.77% (0.6151)	87.46% (0.6768)	87.50% (0.7090)	87.36% (0.7027)	86.97% (0.7210)
NIN	90.54% (0.5032)	90.96% (0.4769)	90.52% (0.4995)	91.05% (0.4674)	92.80% (0.3966)	91.03% (0.4778)	92.03% (0.4321)	90.67% (0.4650)	90.12% (0.5162)	90.47% (0.5057)	90.18% (0.5090)
ResNet	92.54% (0.4707)	92.72% (0.4499)	92.50% (0.4802)	93.47% (0.4332)	94.08% (0.3690)	92.34% (0.4756)	93.82% (0.4114)	92.62% (0.4646)	92.85% (0.4656)	92.35% (0.4749)	92.98% (0.4519)
DenseNet	93.92% (0.3662)	94.23% (0.3553)	93.98% (0.3661)	94.35% (0.3338)	95.81% (0.2712)	93.72% (0.3938)	95.10% (0.3061)	94.08% (0.3799)	93.82% (0.3843)	94.02% (0.3665)	94.21% (0.3648)
WideResNet	94.96% (0.2722)	94.92% (0.2815)	94.81% (0.2876)	94.92% (0.2775)	96.58% (0.2001)	94.51% (0.3015)	95.93% (0.2332)	94.50% (0.2960)	94.77% (0.2859)	94.94% (0.2855)	95.00% (0.2870)
CapsNet	74.73% (0.2016)	75.26% (0.1952)	74.31% (0.2022)	76.80% (0.1879)	77.33% (0.1875)	76.80% (0.1877)	77.38% (0.1838)	75.81% (0.1947)	74.66% (0.2004)	74.88% (0.2001)	74.36% (0.2016)
Spatial Smoothing (SS)											
LeNet	70.01% (0.9215)	69.64% (0.9056)	69.58% (0.9161)	73.24% (0.8163)	74.84% (0.7866)	71.96% (0.8391)	72.98% (0.8421)	71.16% (0.8789)	70.32% (0.9043)	69.44% (0.9182)	70.45% (0.8919)
VGG	83.32% (0.9154)	83.84% (0.9086)	83.13% (0.9237)	85.28% (0.8036)	86.71% (0.7975)	83.46% (0.8935)	86.31% (0.8000)	83.20% (0.9398)	83.75% (0.8702)	83.96% (0.8852)	83.71% (0.8982)
AllConv	81.31% (0.8986)	81.53% (0.9149)	79.12% (1.0347)	82.81% (0.8419)	82.75% (0.9173)	81.23% (0.8904)	83.66% (0.8442)	80.65% (0.9401)	81.84% (0.8994)	81.70% (0.8861)	82.55% (0.8602)
NIN	84.57% (0.7081)	85.11% (0.6947)	84.71% (0.7023)	85.84% (0.6542)	87.78% (0.5963)	84.86% (0.6872)	87.71% (0.5978)	84.94% (0.7071)	84.82% (0.7129)	85.16% (0.6877)	85.00% (0.6977)
ResNet	77.19% (1.4403)	74.57% (1.6944)	78.64% (1.3998)	79.53% (1.3088)	82.51% (1.0585)	76.51% (1.6006)	79.72% (1.3452)	78.96% (1.2993)	79.78% (1.2429)	77.46% (1.4125)	78.62% (1.3955)
DenseNet	78.59% (1.1361)	76.18% (1.4147)	77.20% (1.3156)	79.07% (1.2178)	80.38% (1.0384)	77.88% (1.1653)	80.24% (1.0934)	78.75% (1.2140)	79.35% (1.1429)	77.67% (1.2486)	79.26% (1.1859)
WideResNet	76.93% (1.1204)	77.41% (1.0795)	78.77% (0.9918)	77.33% (1.0777)	79.92% (0.9149)	77.16% (1.1398)	79.94% (0.9821)	77.77% (1.0508)	77.04% (1.1066)	77.62% (1.0670)	78.72% (0.9773)
CapsNet	72.58% (0.2213)	73.32% (0.2139)	71.55% (0.2226)	74.23% (0.2051)	74.68% (0.2072)	73.97% (0.2082)	74.52% (0.2034)	72.96% (0.2137)	72.23% (0.2184)	72.80% (0.2170)	71.96% (0.2204)
Label Smoothing (LS)											
LeNet	75.15% (0.8138)	73.75% (0.8275)	74.04% (0.8233)	77.30% (0.7345)	78.94% (0.7011)	76.04% (0.7668)	77.65% (0.7297)	75.88% (0.7764)	74.70% (0.7967)	74.71% (0.8141)	74.93% (0.8047)
VGG	92.63% (0.5205)	92.94% (0.4901)	92.06% (0.5153)	93.25% (0.4723)	94.51% (0.4390)	92.46% (0.4896)	93.91% (0.4609)	92.46% (0.5008)	92.51% (0.4922)	92.56% (0.5025)	92.52% (0.5034)
AllConv	89.03% (0.5349)	89.23% (0.5044)	86.54% (0.5973)	89.37% (0.5091)	90.24% (0.4730)	89.06% (0.5272)	89.97% (0.4995)	88.51% (0.5381)	88.00% (0.5374)	87.85% (0.5568)	88.46% (0.5402)
NIN	90.28% (0.4102)	90.78% (0.3986)	90.13% (0.4148)	91.26% (0.3819)	92.95% (0.3297)	90.97% (0.3984)	92.02% (0.3582)	90.74% (0.3921)	90.56% (0.4020)	90.17% (0.4157)	90.34% (0.4085)
ResNet	92.48% (0.4126)	93.01% (0.3910)	92.54% (0.4036)	93.42% (0.3724)	94.38% (0.3399)	92.81% (0.3949)	94.02% (0.3500)	92.58% (0.4010)	92.06% (0.4166)	93.04% (0.3977)	92.41% (0.4048)
DenseNet	94.08% (0.3712)	94.70% (0.3424)	94.10% (0.3616)	94.85% (0.3405)	95.47% (0.3067)	94.03% (0.3606)	95.31% (0.3180)	94.50% (0.3485)	93.63% (0.3738)	94.30% (0.3620)	94.20% (0.3626)
WideResNet	95.12% (0.2954)	95.33% (0.2935)	95.03% (0.3014)	95.44% (0.2849)	96.61% (0.2486)	94.97% (0.3018)	95.84% (0.2646)	95.06% (0.2998)	95.21% (0.2957)	95.14% (0.2982)	95.18% (0.2914)
CapsNet	74.24% (0.2115)	75.95% (0.1979)	74.95% (0.2034)	77.83% (0.1849)	78.40% (0.1863)	72.74% (0.2220)	77.88% (0.1865)	76.48% (0.1978)	76.35% (0.2004)	75.35% (0.2004)	74.84% (0.2072)
Thermometer Encoding (TE)											
LeNet	65.73% (1.0599)	64.36% (1.0869)	64.21% (1.0705)	68.68% (0.9427)	68.63% (0.9557)	67.84% (0.9763)	67.95% (0.9905)	64.60% (1.0635)	66.60% (1.0139)	65.70% (1.0272)	65.92% (1.0513)
VGG	84.01% (0.8730)	82.77% (0.9430)	83.72% (0.8934)	84.96% (0.8305)	86.97% (0.7566)	85.01% (0.8250)	85.74% (0.8114)	84.62% (0.8435)	83.78% (0.8753)	83.94% (0.8783)	81.83% (0.9777)
AllConv	77.71% (1.1347)	78.80% (1.0777)	76.86% (1.1854)	79.41% (1.0531)	81.21% (0.9174)	79.24% (0.9883)	80.01% (1.0531)	78.84% (1.0680)	78.40% (1.0330)	77.47% (1.0763)	77.34% (1.1180)
NIN	81.75% (0.8457)	82.80% (0.8031)	81.78% (0.8305)	83.48% (0.7702)	84.91% (0.7120)	82.80% (0.7990)	84.03% (0.7526)	82.00% (0.8240)	82.00% (0.8579)	81.38% (0.8481)	81.27% (0.8554)
ResNet	83.04% (1.0909)	83.55% (1.0777)	82.65% (1.1141)	84.56% (0.9993)	86.14% (0.8782)	83.81% (1.0596)	85.31% (0.9196)	83.44% (1.0720)	82.66% (1.1020)	82.91% (1.1110)	83.06% (1.1231)
DenseNet	85.08% (0.9698)	85.93% (0.9420)	84.73% (0.9839)	86.97% (0.8584)	88.05% (0.7693)	86.27% (0.9042)	87.41% (0.8407)	85.08% (0.9649)	85.33% (0.9369)	85.18% (0.9697)	85.93% (0.9338)
WideResNet	86.55% (0.7288)	86.33% (0.7479)	86.14% (0.7623)	87.78% (0.6851)	89.26% (0.5762)	86.83% (0.7120)	88.42% (0.6256)	86.48% (0.7444)	86.40% (0.7646)	86.16% (0.7493)	86.48% (0.7286)
CapsNet	32.45% (0.6018)	25.58% (0.6569)	60.31% (0.2874)	62.42% (0.2744)	49.08% (0.4128)	41.34% (0.5214)	60.75% (0.2868)	47.27% (0.4277)	41.77% (0.4795)	59.63% (0.2926)	44.30% (0.4665)
Adversarial Training (AT)											
LeNet	60.87% (1.2041)	60.47% (1.1789)	61.47% (1.1583)	64.53% (1.0746)	65.66% (1.0460)	63.73% (1.0766)	65.13% (1.0730)	61.81% (1.1341)	61.91% (1.1414)	61.13% (1.1649)	61.62% (1.1502)
VGG	82.20% (0.7734)	83.35% (0.7194)	84.45% (0.7192)	84.96% (0.6834)	87.33% (0.6344)	85.11% (0.6706)	86.97% (0.6412)	82.95% (0.7273)	83.28% (0.7395)	83.80% (0.7537)	84.61% (0.7087)
AllConv	80.99% (0.7222)	81.23% (0.7077)	78.63% (0.7888)	82.55% (0.6645)	84.20% (0.6321)	81.42% (0.6756)	83.30% (0.6543)	80.45% (0.7303)	80.05% (0.7271)	79.95% (0.7501)	80.84% (0.7262)
NIN	83.98% (0.6079)	84.73% (0.5774)	83.21% (0.6130)	85.57% (0.5305)	87.57% (0.4820)	85.33% (0.5294)	86.92% (0.5127)	84.83% (0.5729)	84.38% (0.5866)	84.26% (0.5928)	85.41% (0.5868)
ResNet	82.53% (0.6259)	81.63% (0.6726)	83.72% (0.6039)	84.27% (0.5981)	86.56% (0.5302)	82.97% (0.6103)	84.45% (0.6006)	83.21% (0.6110)	82.00% (0.6728)	82.12% (0.6802)	79.30% (0.7027)
DenseNet	85.40% (0.5815)	84.95% (0.6055)	84.08% (0.6355)	86.34% (0.5379)	88.14% (0.4942)	85.91% (0.5378)	88.27% (0.5096)	85.16% (0.5934)	82.91% (0.8311)	84.16% (0.6158)	83.80% (0.6019)
WideResNet	84.67% (0.8057)	84.90% (0.7483)	84.11% (0.8626)	85.78% (0.7780)	88.43% (0.6124)	85.30% (0.8425)	87.57% (0.6585)	85.08% (0.7759)	84.50% (0.8096)	83.55% (0.8256)	84.35% (0.8034)
CapsNet	65.97% (0.2746)	65.75% (0.2678)	64.92% (0.2694)	69.87% (0.2457)	71.95% (0.2377)	70.55% (0.2425)	52.71% (0.3352)	62.71% (0.2842)	67.48% (0.2605)	65.63% (0.2697)	67.58% (0.2579)

Table 7: Classifier Accuracy on test dataset of the learned classes for different architectures.

Attack	For Fashion MNIST	For CIFAR-10 and Sub Imagenet
FGM	norm $= L_\infty$, $\epsilon = 0.3$, $\epsilon_{\text{step}} = 0.01$	norm $= L_\infty$, $\epsilon = 8$, $\epsilon_{\text{step}} = 2$
BIM	norm $= L_\infty$, $\epsilon = 0.3$, $\epsilon_{\text{step}} = 0.01$, iterations $= 80$	norm $= L_\infty$, $\epsilon = 8$, $\epsilon_{\text{step}} = 2$, iterations $= 10$
PGD	norm $= L_\infty$, $\epsilon = 0.3$, $\epsilon_{\text{step}} = 0.01$, iterations $= 40$	norm $= L_\infty$, $\epsilon = 8$, $\epsilon_{\text{step}} = 2$, iterations $= 20$
DF	iterations $= 100$, $\epsilon = 0.02$	iterations $= 100$, $\epsilon = 0.000001$
NF	iterations $= 100$, eta $= 0.375$	iterations $= 100$, eta $= 0.01$

Table 8: Description of Adversarial Attack Parameters

Classifier	Adversarial Accuracy (in %) FGM	BIM	PGD	DF	NF	Mean L_2 Score FGM	BIM	PGD	DF	NF
Fashion MNIST										
MLP	91.08	91.29	91.29	27.16	25.39	210.73	638.83	638.83	309.41	289.28
ConvNet	86.89	89.20	89.18	23.63	22.67	306.25	669.56	665.76	314.81	263.65
CIFAR-10										
LeNet	84.58	89.12	89.25	31.70	84.12	152.37	345.27	357.34	132.32	49.61
VGG	82.79	94.97	94.99	65.08	92.43	181.29	321.86	329.96	651.65	77.01
AllConv	67.09	69.11	69.11	51.46	61.86	155.95	273.90	274.15	487.46	61.05
NIN	72.49	74.26	74.26	59.94	66.76	140.46	216.97	216.96	492.90	54.78
ResNet	52.75	55.41	55.41	58.71	54.39	124.70	164.64	164.64	458.57	51.56
DenseNet	50.78	52.11	52.11	60.83	50.81	120.03	160.34	160.38	478.03	53.89
WideResNet	69.59	89.42	89.44	60.10	82.73	159.88	208.44	208.49	613.14	63.13
CapsNet	70.02	82.23	84.46	87.40	90.04	208.89	361.63	370.90	258.08	1680.83
Sub-Imagenet										
InceptionV3	85.76	87.24	87.24	86.94	58.44	796.53	1204.01	1204.01	609.54	319.73
ResNet-50	85.74	86.72	86.72	84.78	60.84	826.06	1264.30	1264.34	633.30	336.80

Table 9: Adversarial Accuracy and Mean L_2 Score for each classifier and adversarial attack pair.

F VISUALISATIONS OF DAVIES-BOULDIN METRIC (DBM)

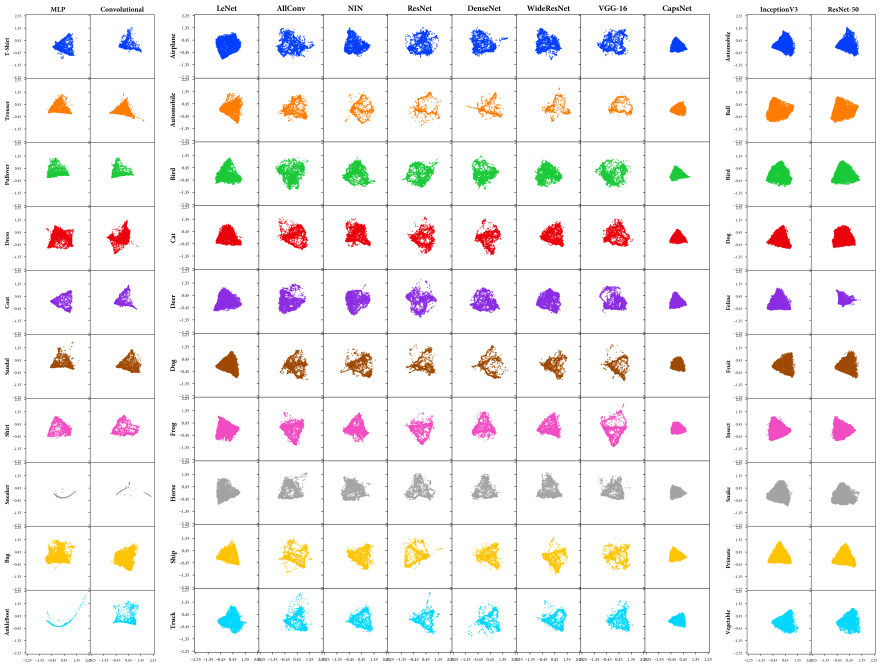


Figure 3: Visualisation of the DBM results for vanilla classifiers using a topology preserving two-dimensional projection with Isometic Mapping (IsoMap). Each row represents a classifier trained with a label excluded whose projection is visualised.

Figures 3-6 shows visualization of DBM metric using Isometric Mapping (IsoMap) (Tenenbaum et al., 2000), t-Distributed Stochastic Neighbour Embedding (t-SNE) (Maaten & Hinton, 2008), Multi-dimensional Scaling (MDS) (Kruskal, 1964), and Spectral Embedding (SE) (Belkin & Niyogi, 2003) respectively. The characteristic of IsoMap is that it seeks a lower-dimensional embedding which maintains geodesic distances between all sample points that is it preserves the high-dimensional distance between the points. t-SNE tries to model similar data points in higher-dimensional space through small pairwise distances in lower-dimensional space. In other words, it tries to minimise the Kullback–Leibler divergence between the two distributions of points in the map. SE is a non-linear embedding, which finds a lower-dimensional representation of the sample points using a spectral decomposition of the graph Laplacian Eigenmaps. It is to be noted that Isomap (Figure 3), t-SNE (Figure 4), MDS (Figure 5), and SE (Figure 6) are different visualisations for the same feature space. The idea for having these visualisations is to investigate whether the cluster for the unknown class can be segregated into one or more different classes. In other words, we try to investigate visually whether there exists a single combination of Amalgam Proportion for the unknown class.

The projections (Figures 4-6) of CapsNet is uniform and dense while the other networks have more scattered non-uniform projections. The non-uniform projection, which can be split into multiple clusters, of the other networks might suggest that the learned representation is not continuous/homogeneous enough. Interestingly, LeNet have more dense and uniform projections compared to other static neural networks, further suggesting the better representation of the LeNet. These results are in accordance with the previous experiments (Section 5) on representation quality.

Another way to verify this interpretation is to look at the gaps in the projections, which is observing the behaviour of different data points. For CapsNet, even if we form clusters to have different classes, the gaps between the classes will be too small relative to other architectures. It also shows that in the high-dimensional space, all the soft-labels are moderately close to each other, also verified using Amalgam Metric (Table 2 and Figure 7). While for the other architectures, there exist some points which can form their separate cluster and be termed as a different class. Hence, for these architectures, it can have one or more different Amalgam proportion for the same unknown class

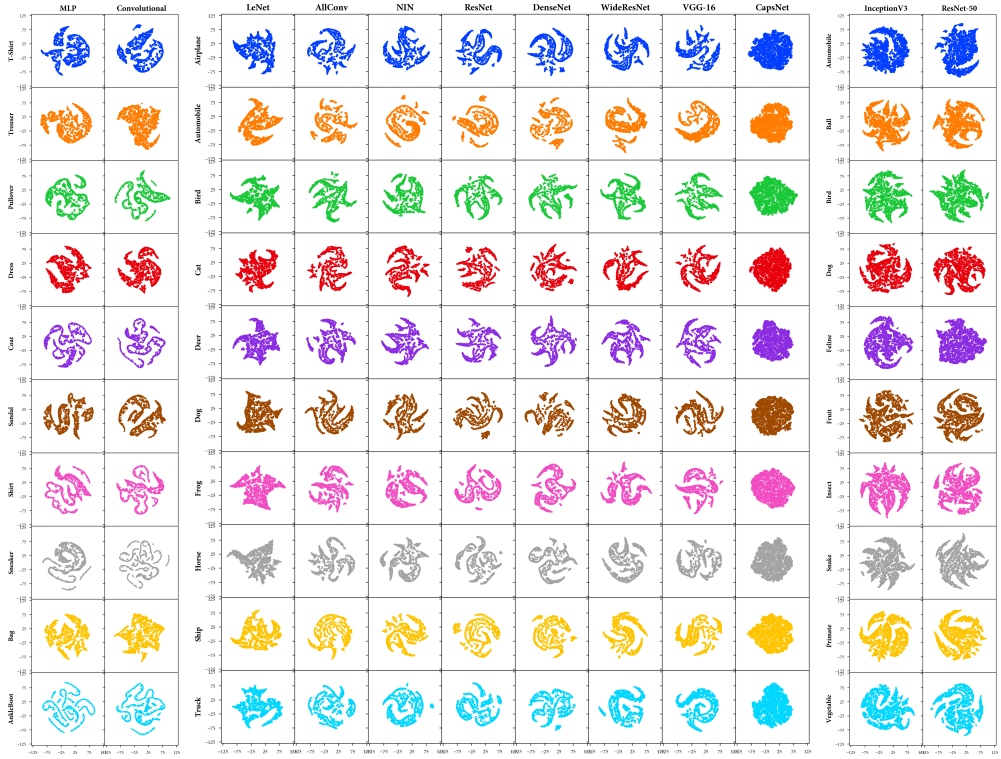


Figure 4: Visualisation of the DBM results for vanilla classifiers using t-Distributed Stochastic Neighbour Embedding (t-SNE). Each row represents a classifier trained with a label excluded whose projection is visualised.

which is contradicting to our hypothesis that there should exist only a single Amalgam proportion for a single unknown class. Note that, this dense projection does not necessarily mean that the unknown class has converged to a single known class. It gives a visualisation that the Amalgam Proportion of the unknown class is similar.

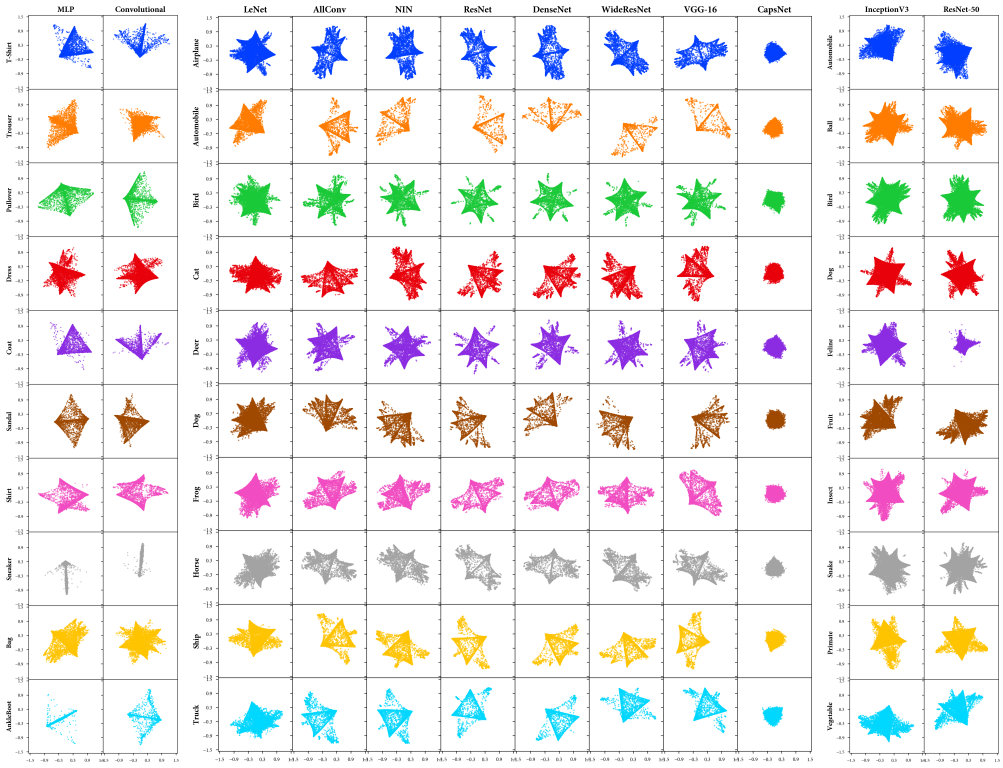


Figure 5: Visualisation of the DBM results for vanilla classifiers using Multi Dimensional Scaling (MDS). Each row represents a classifier trained with a label excluded whose projection is visualised.

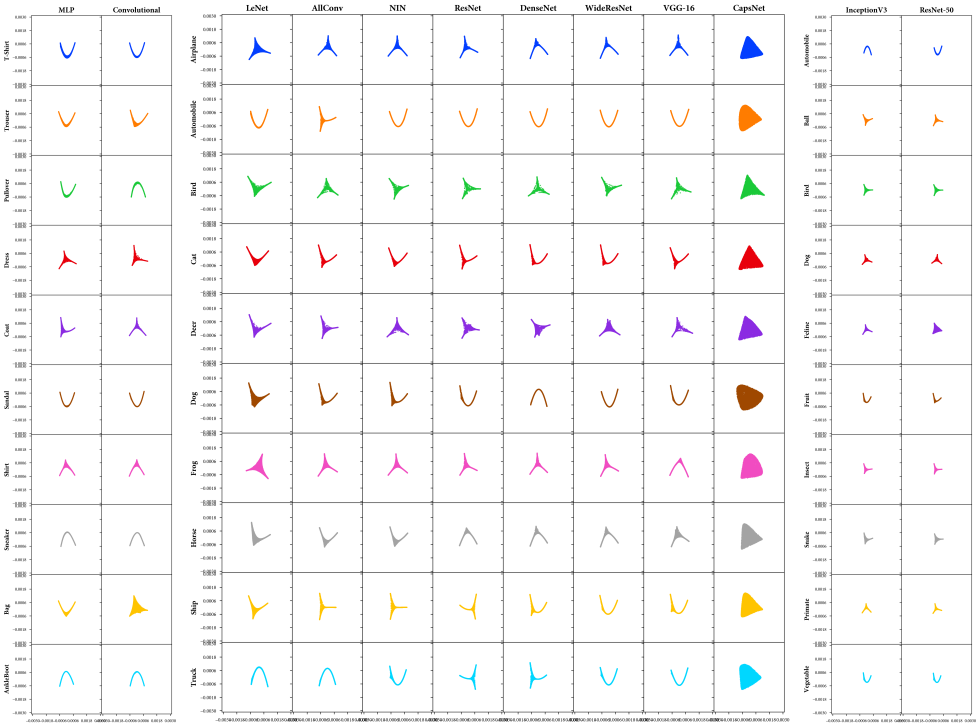
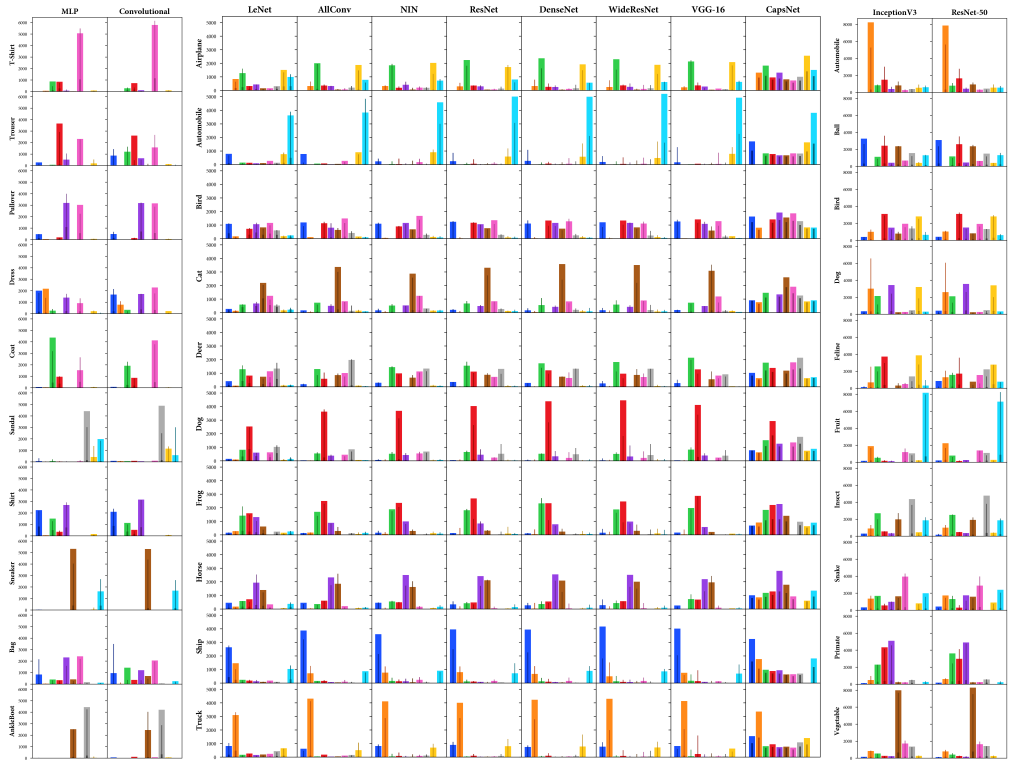


Figure 6: Visualisation of the DBM results for vanilla classifiers using Spectral Embedding (SE). Each row represents a classifier trained with a label excluded whose projection is visualised.

G VISUALISATION OF AMALGAM METRIC (AM)



Color Encoddings Of Classes For Fashion MNIST, CIFAR-10 and Sub Imagenet (In Order)

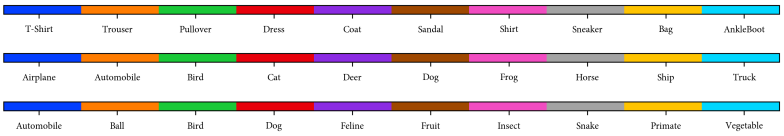


Figure 7: Histograms of soft-labels (H' and H) from which the AM is calculated. Each row shows the histograms of one classifier with one class excluded. Dark-shaded thinner and light-shaded broader bins are respectively the soft-labels from the ground-truth (H') from the classifier trained on all classes and the soft-labels of the classifier trained on $N-1$ classes (H').

To enable the visualisation of the Amalgam Metric, the computed histograms (H' and H) is plotted for every class and classifier (Figure 7). It is interesting to note that the histograms of CapsNet (Figure 7) are different from the other ones. This reveals that this metric can capture such representation differences. It can be noted (Figure 7) that for most classes of CapsNet, the variation is relatively low than the other architectures. This contributes to having a good representation of CapsNet.

A further study can also be carried out to analyse the characteristics of representation of the neural network, which makes a class more robust than the other classes. Further investigations can be also be carried out to analyse the effect of a class for an adversarial attack based on this. This can also provide insight into the classes which are robust to adversarial attacks. However, these analyses are beyound the scope for the current article, and hence, left for future work.

H VISUALISATION OF PEARSON CORRELATION

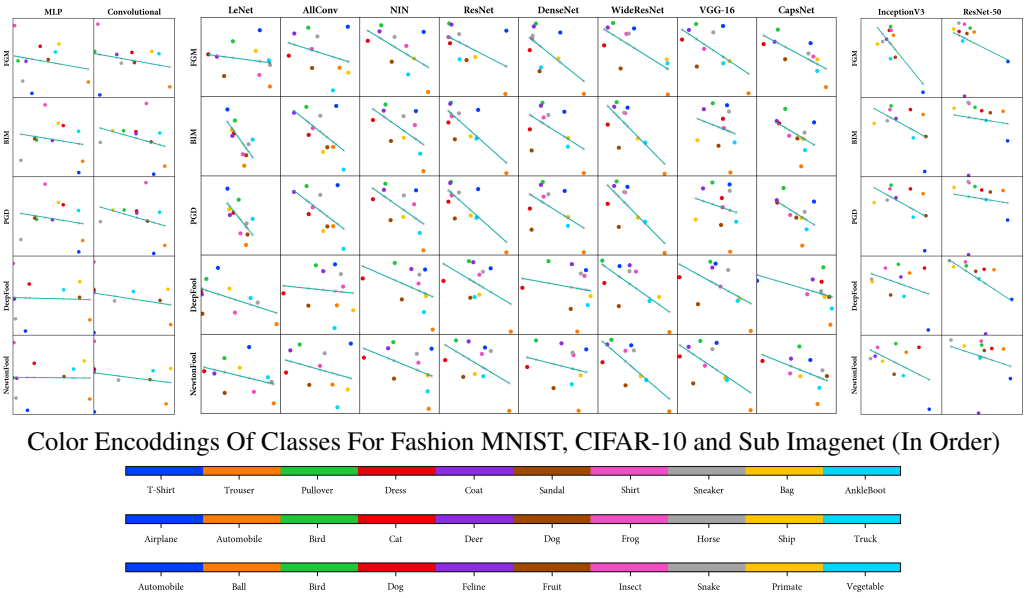


Figure 8: Visualisation of Pearson Correlation of Davies-Bouldin Metric (DBM) with Mean L_2 Score of adversarial attacks (Table 3). Here, the x-axis represents the Mean L_2 Scores while the y-axis represents the DBM values. Each point represent a DBM value and Mean L_2 Score for a labelled class.

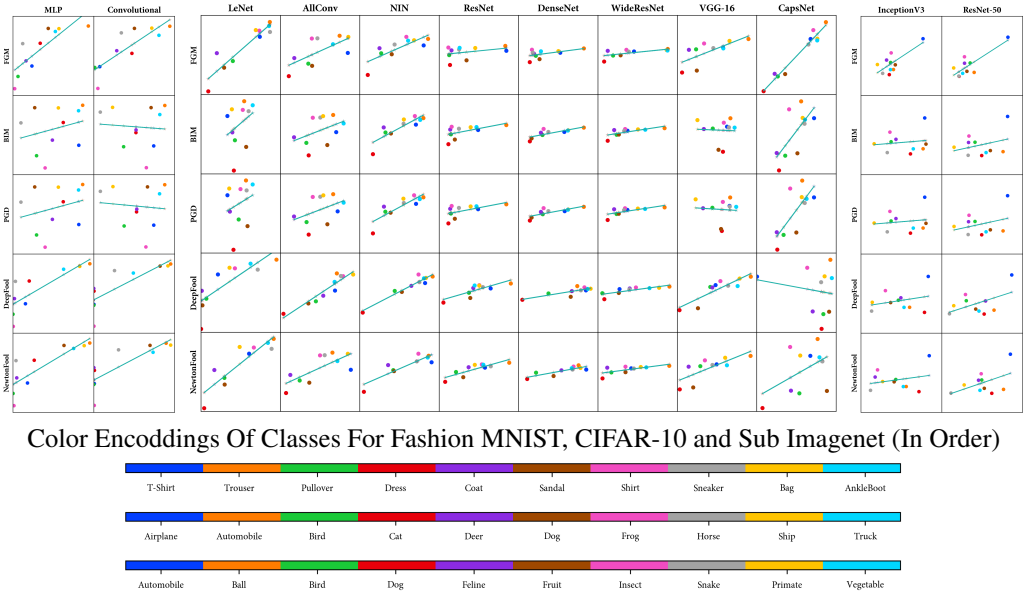


Figure 9: Visualisation of Pearson Correlation of Amalgam Metric (AM) with Mean L_2 Score of adversarial attacks (Table 3). Here, the x-axis represents the Mean L_2 Scores while the y-axis represents the AM values. Each point represent an AM value and Mean L_2 Score for a labelled class.

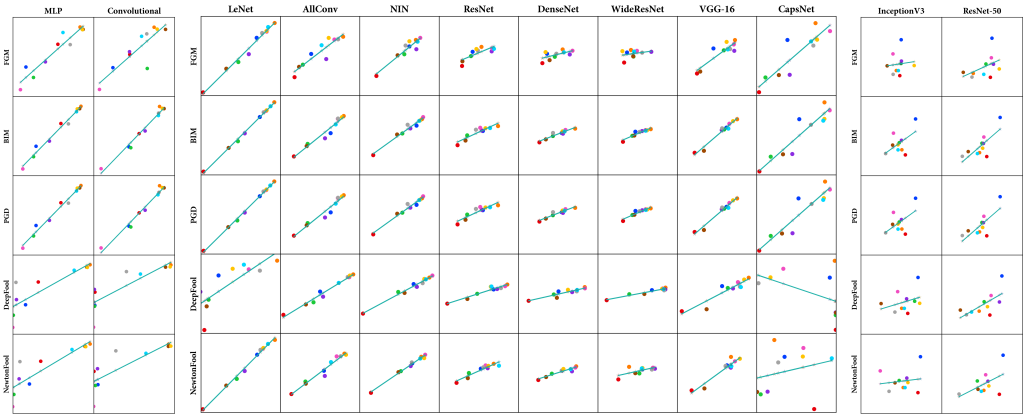
Here, we visualise the Pearson correlation between the Raw Zero-Shot metrics (DBM and AM) with the adversarial metrics (Adversarial Accuracy and Mean L_2 Score) mentioned in Tables 3. Figures 8 and 9, visualizes the relationship of Raw Zero-Shot metrics with adversarial metrics. In the Section 6, we observed some anomalies in the Pearson correlation values (Table 3). Here we try to understand

these anomalies with the help of our visualisation (Figure 9). On visualising the Pearson correlation, we identify that DeepFool attacks the Airplane class of CapsNet with much less L_2 score compared to the other classes. This abnormal behaviour of DeepFool for the Airplane class causes the anomaly for the Pearson Correlation.

I ANOTHER OUTLOOK ON LINK BETWEEN REPRESENTATION QUALITY AND ADVERSARIAL ATTACKS

Classifier	Confidence Score					Pearson Correlation of AM with Confidence Score				
	FGM	BIM	PGD	DF	NF	FGM	BIM	PGD	DF	NF
Fashion MNIST										
MLP	0.63	0.90	0.90	0.38	0.35	0.95 (0.00)	0.99 (0.00)	0.99 (0.00)	0.86 (0.00)	0.87 (0.00)
ConvNet	0.62	0.90	0.90	0.33	0.34	0.86 (0.00)	0.99 (0.00)	0.99 (0.00)	0.85 (0.00)	0.82 (0.00)
CIFAR-10										
LeNet	0.58	0.72	0.72	0.12	0.48	0.99 (0.00)	1.00 (0.00)	1.00 (0.00)	0.79 (0.01)	1.00 (0.00)
AllConv	0.82	0.91	0.91	0.71	0.69	0.92 (0.00)	0.97 (0.00)	0.97 (0.00)	0.99 (0.00)	0.98 (0.00)
NIN	0.87	0.93	0.93	0.78	0.75	0.94 (0.00)	0.98 (0.00)	0.98 (0.00)	0.99 (0.00)	0.99 (0.00)
ResNet	0.90	0.94	0.94	0.82	0.76	0.79 (0.01)	0.91 (0.00)	0.91 (0.00)	0.96 (0.00)	0.94 (0.00)
DenseNet	0.91	0.95	0.95	0.85	0.76	0.60 (0.07)	0.95 (0.00)	0.94 (0.00)	0.91 (0.00)	0.94 (0.00)
WideResNet	0.87	0.97	0.97	0.84	0.77	0.31 (0.39)	0.94 (0.00)	0.94 (0.00)	0.96 (0.00)	0.65 (0.04)
VGG-16	0.86	0.95	0.95	0.82	0.75	0.89 (0.00)	0.98 (0.00)	0.98 (0.00)	0.93 (0.00)	0.95 (0.00)
CapsNet	0.17	0.46	0.48	-0.10	0.15	0.89 (0.00)	0.93 (0.00)	0.93 (0.00)	-0.52 (0.13)	0.24 (0.50)
Sub-Imagenet										
InceptionV3	0.92	0.95	0.95	0.86	0.82	0.10 (0.78)	0.54 (0.11)	0.54 (0.11)	0.33 (0.35)	0.14 (0.70)
ResNet-50	0.90	0.94	0.94	0.85	0.81	0.44 (0.20)	0.75 (0.01)	0.75 (0.01)	0.67 (0.03)	0.53 (0.12)

Table 10: Confidence Difference Score and it's Pearson Correation value (and p-value) for each classifier and adversarial attack pair.



Color Encoddings Of Classes For Fashion MNIST, CIFAR-10 and Sub Imagenet (In Order)

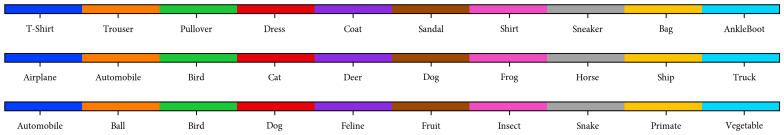


Figure 10: Visualisation of Pearson Correlation of Amalgam Metric (AM) with Confidence Score of adversarial attacks (Table 10). Here, the x-axis represents the Confidence Difference Scores. In contrast, the y-axis represents the AM values. Each point represents an AM value and Confidence Difference Scores for a labelled class.

In this section, we analyse the representation quality from the perspective of Confidence Score, which is defined as the change in the confidence of the true label by an adversarial sample. To further deeply analyse the statistical relevance of this link between representation quality and adversarial attacks, we here conduct a Pearson Correlation test of Amalgam Metric of the vanilla classifiers with

Confidence Score of the adversarial attacks. The Pearson correlation value of the Amalgam Metric with Confidence Score is shown in Table 10 for every architecture and attacks. Table 10 also mentions the Confidence Score of every classifier-attack pair. Moreover, similar to our previous analysis, these Pearson relationships between the Amalgam Metric and Confidence Score can also be visualised (Figure 10).

The purpose of evaluating the Confidence Score as an adversarial metric is because the score effectively assesses the impact of the adversarial attacks on true class soft-label. This perspective gives us the effectiveness of an attack on the soft-label of the representation we evaluate. Therefore, here Confidence Score not only determines the alteration in the representation space, but it also analyses the effectiveness of an attack across different classes.

The correlational analysis of our Amalgam Metric suggests a relationship between our Amalgam Metric and the adversarial attacks in general. We do observe some anomalies in this Pearson correlation also with AM of DeepFool for CapsNet. However, we believe this anomaly is due to the adversarial attack itself. Note that in Table 10 the Confidence Score of the DeepFool attack for CapsNet is negative, which suggests that DeepFool, instead of decreasing the soft-label of the true-class, increases the soft-label of the misclassified class. We do note that more investigations are required to better understand the behaviour of Capsule Networks, in general.