DATA AUGMENTATION WITH MOBIUS TRANSFORMATIONS

Derek T. Brown, Dariya Mamyrbek, Hong Nguyen, Margo Thompson

1 INTRODUCTION

Machine learning is revolutionizing almost all fields of computer science nowadays. Particularly, deep neural networks have shown a good result in solving many problems in the field of computer vision. To achieve the best results, deep learning models have to be trained on large data sets. In the real world, insufficient data is one of the common issues. This problem could possibly lead to a model overfitting when a model performs poorly on an unseen data, and on contrary, fits the training data well. There are several ways to handle the lack of data including building a simple model with fewer parameters, transfer learning, synthetic data, and data augmentation. Data augmentation is a valuable tool to solve the problem of limited data. It performed well on different computer vision tasks including image classification, object detection, and instance segmentation [1]. One of the ways to expand the existing dataset is to use conformal transformations which are transformations that preserve local angles. The primary research paper this project will be referring to is "Data Augmentation with Mobius transformations" [Zho+21], where they proposed a novel data augmentation technique using bijective conformal maps known as Mobius transformations. Mobius transformations preserve image-level labels by minimizing local distortions in an image. In our project we are interested to test the results obtained in the above-mentioned paper, thus, we expect that with the use of data augmentation technique, the validation error will decrease with the training error. Hence, this would improve the generalizability of the model and increase the accuracy of the classification prediction. The outline of the project is as follows: we will discuss methods including datasets which will be used, mobius transformation, and cross validation. Methods will be followed by the experiments and analysis parts. To finish, there will be a conclusion part.

2 METHODS

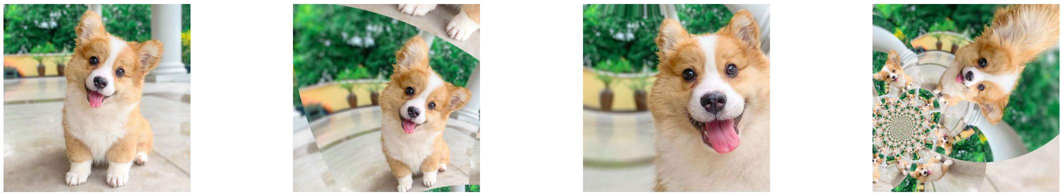
2.1 DATA SOURCES AND SOFTWARE

The input images we will use in our data augmentation will be taken from the CIFAR-10 dataset. The CIFAR-10 is a collection of images collected by groups at MIT and NYU that are commonly used to train machine learning and computer vision algorithms [KH+09]. It is a poplular dataset used for machine learning research with Google Scholar listing it as having over twelve thousand citations. It consists of 10 classes of images (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck) with 6000 images per class. We will gather a subset of the CIFAR-10 images and train the neural network to classify them. We will then apply Mobius transformations on the images to have a larger collection of photos. The remaining images from CIFAR-10 will be used to test the accuracy of the neural network. The analysis will be performed using the software Python and its package PyTorch. We are using PyTorch for neural network architecture.

2.2 MOBIUS TRANSFORMATIONS

Mobius transformations are a type of conformal mapping. A conformal mapping is a mapping that preserves local angles [Zho+21]. This means that any angle between a set of two intersecting lines of an image will remain remain the same after a conformal mapping has been applied to it. Mobius transformations also preserve the anharmonic ratio [Zho+21].The anharmonic ratio, also known as cross ratio has been used for identifying objects from different perspectives with a high accuracy [Fry00]. The preservation of the this ratio will allow mobius transormations of an image to represent the subject of the image from multiple perspectives.

Mobius transformations can be defined as:



(a) Original Dog (b) Aug. Dog 1 (c) Aug. Dog 2 (d) Aug. Dog 3

Figure 1: Examples of Augmentation [Zho+21]

$$f(z) = \frac{az + b}{cz + d} : ad - bc \neq 0$$

Where a, b, c, d are complex numbers.This set of transformations contains basic translations, rotations, and inversions in addition to many other types of translations. Mobius transforms reflect around the unit circle which allows for different distortions in scale [Zho+21]. This allows for different representations of the image at different scales which will aid in training the neural network. One issue that comes up is balancing the correct amount of distortion on an image. Too much distortion will result in the image not representing the object it is classified as. Too little distortion will result in the image serving as a duplicate of the original which is not helpful. This is addressed by adding the following constraint as done by Sharon Zhou et al. [Zho+21]:

$$\frac{1}{M} < |f'| < M : M > 1$$

Examples of these augmentations are shown above in Figure 1.

2.3 Cross Validation

If we augment our training dataset with the Mobius transformation and consider the few following facts:

- If we train the same network (that is, utilize the same network structure and configuration parameters) using the same training dataset, then the final network parameters will be the same, provided the training is done without validation or with fixed validation sets
- If the network has the same parameters, then its accuracy when testing on the same testing dataset will be the same.
- If we apply the same Mobius transformation (that is, Mobius transformations with the same parameters) to the same dataset, the new dataset that is generated will be the same.

It then follows that if we:

- Fix the initial training and testing datasets
- Fix the validation sets or ignore validation during training
- Fix the network structure and configurations

Then the network's accuracy on the testing dataset can be expressed as a function of the Mobius transformation parameters. That is, nothing else can change the network's accuracy except for the parameters a, b, c, d of the Mobius transformation used for data augmentation, save for random variation. We can then build a regression model using the network accuracy as our output and the Mobius transformation parameters as our explanatory variables. The weights of the parameters in this model will tell us roughly which parameters we should focus on increasing or decreasing when we want to improve the network accuracy by augmenting the dataset with Mobius transformations. The higher the weight, the more increasing the associated parameter is likely to improve the network's accuracy. By analysing the characteristics of this model, we can also test if this hypothesis is accurate. That is, how confident are we that modifying a certain parameter will improve the network's accuracy.

2.4 Our analysis

In our analysis we used CIFAR-10 dataset with 10 classes with 10,000 images per class. We have used VGG-16 convolutional neural network architecture. As a training data 1000 images were picked from CIFAR-10 with 100 images per class. For the testing dataset, 200 images were used with 20 images per class. All images from the training dataset were augmented using Mobius transformation with 14 different sets of parameters (a,b,c,d). The augmented data from each of these parameters set were added to the base training dataset, and each of these augmented dataset was used to train a different neural network. All these neural networks had the same architecture and training parameters. After training, we tested these neural networks on the same testing dataset and compared the accuracy, and as discucces in section 2.3, these accuracy can be expressed as a function of the Mobius parameters, so we can analyze the correlation between the Mobius parameters and the according neural network's accuracy.

3 Results and Discussion

In this section, we discuss our results. We performed linear regression with 8 explanatory variables, that is, the real and imaginary parts of the Mobius transformation parameters. In order to do this, we first transformed the output variable - the neural network accuracy - from a sigmoid function used to express probabilities to a linear function using the formula

$$ln(\frac{y}{1-y})$$

, and run the least-squares linear regression model. To transform a prediction a acquired from this linear model back to a predicted neural network accuracy value, we use the regular sigmoid function:

$$1 - \frac{1}{exp(a) + 1}$$

In table 1, there are input variables, corresponding accuracy and predicted values of the accuracy according to the linear model.

Index	Rea	Ima	Reb	Imb	Rec	Imc	Red	Imd	NN Accuracy (Y)	ln(y/(1-y))	Predicted ln(y/(1-y))	Predicted Accuracy
0	2653,425	1278,376	-69670,2	-40441,1	25,48252	0,614965	-2023,24	1823,168	**0,295**	-0,871222446	-0,97041537	***0,275***
14	15472,36	-1833,08	-504284	-241991	42,55477	90,77381	-8353,8	4681,196	**0,305**	-0,823600069	-0,93618723	***0,282***
15	9364,018	19069,61	439679,1	-445856	20,51377	131,2122	11704,43	-14187,8	**0,280**	-0,944461609	-0,91977508	***0,285***
16	-6737,98	2054,642	71188,13	-95406,1	71,26073	-59,6906	-10297,7	-4586,06	**0,290**	-0,895384047	-0,92489857	***0,284***
17	5409,857	-17533	-581465	-24664,6	28,9687	-110,757	-16453,3	879,2039	**0,280**	-0,944461609	-0,86620137	***0,296***
23	-8083,16	-1886,84	167780,9	292350,4	-37,7551	13,71607	6576,829	1396,496	**0,285**	-0,919793362	-0,99626427	***0,270***
24	-2759,23	-11165,3	-178710	629291	35,31696	61,4023	8133,831	13157,86	**0,300**	-0,84729786	-0,80722615	***0,308***
25	2756,119	-3283,11	-192023	-4274,53	33,77483	-35,6558	-3558,69	2551,567	**0,285**	-0,919793362	-1,02833081	***0,263***
26	13104,19	-1109,93	-171365	-465360	29,14511	-157,913	-9737,25	-7177,35	**0,265**	-1,020140673	-0,94694503	***0,279***
27	2740,334	6130,075	-58244,1	-22138,8	31,47845	73,45601	74,63194	4252,172	**0,250**	-1,098612289	-0,95624654	***0,278***
28	-1375,97	7306,895	212942,7	-271822	39,33925	2,827207	-3102,25	-6942,78	**0,230**	-1,208311206	-1,13595885	***0,243***
29	-9415,08	-3163,02	-146157	224783,6	138,8144	144,1789	-7609,43	4053,692	**0,260**	-1,045968555	-0,99428488	***0,270***
30	2253,525	-1351,14	2531,832	-13248,1	6,417891	24,56157	1378,483	-2952,37	**0,305**	-0,823600069	-0,92287287	***0,284***
31	9402,671	-10744,2	-338707	453639,7	-232,458	105,8951	18644,79	14668,6	**0,245**	-1,125459539	-1,08249968	***0,253***
Control	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	**0,050**	-2,944438979		

Figure 2: Table 1 - Mobius transformation parameters, actual neural network accuracy and predicted accuracy through linear regression model

In Table 2, we collected some important statistics.

4 Conclusion

From the linear regression, the R-squared value is 0.488 which is a comparatively good result considering the small number of data points we have, so it is practical to predict the neural network's accuracy with any set of Mobius transformation parameters using a linear model before the actual training process. The accuracy for almost any case with data augmentation would be much higher

OLS Regression Results

Dep. Variable:	Accuracy_ln	**R-squared:**	0.488
Model:	OLS	**Adj. R-squared:**	-0.331
Method:	Least Squares	**F-statistic:**	0.5959
Date:	Sat, 13 Nov 2021	**Prob (F-statistic):**	0.755
Time:	03:17:22	**Log-Likelihood:**	15.116
No. Observations:	14	**AIC:**	-12.23
Df Residuals:	5	**BIC:**	-6.480
Df Model:	8		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	-1.0270	0.051	-20.031	0.000	-1.159	-0.895
Ima	3.74e-05	4.3e-05	0.871	0.424	-7.3e-05	0.000
Imb	3.071e-06	2.03e-06	1.511	0.191	-2.15e-06	8.3e-06
Imc	-0.0001	0.002	-0.073	0.945	-0.004	0.004
Imd	-4.639e-05	3.95e-05	-1.173	0.294	-0.000	5.53e-05
Rea	6.015e-05	5.4e-05	1.114	0.316	-7.86e-05	0.000
Reb	3.906e-07	2.28e-06	0.171	0.871	-5.48e-06	6.26e-06
Rec	-0.0007	0.002	-0.359	0.734	-0.006	0.004
Red	-5.119e-05	6.1e-05	-0.839	0.440	-0.000	0.000

Omnibus:	2.952	**Durbin-Watson:**	1.896
Prob(Omnibus):	0.229	**Jarque-Bera (JB):**	1.102
Skew:	0.090	**Prob(JB):**	0.576
Kurtosis:	1.637	**Cond. No.**	4.58e+05

Figure 3: Table 2 - Statistics of the linear regression model

than the control case - the case without any data augmentation. Hence, we can conclude that Mobius transformation is a helpful tool for data augmentation. With Mobius transformation for data augmentation, the neural network's accuracy in any case is also higher than random guessing (there are 10 classes, each with an equal number of images in the testing dataset, so the expected value of the neural network accuracy through random guessing is 0.1). Thus, we can conclude that the data augmentation using Mobius transformations helped the neural network learn to better classify the images, whereas the unaugmented dataset did not: In the control case, the accuracy was actually lower than random guessing, suggesting that there was a lot of cross-confusion between classes. The data augmentation alleviated this issue.

REFERENCES

[Fry00] John Fryer. "An object space technique independent of lens distortion for forensic videogrammetry". In: (Jan. 2000).

[KH+09] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images". In: (2009).

[Zho+21] Sharon Zhou et al. "Data augmentation with Mobius transformations". In: *Machine Learning: Science and Technology* 2.2 (2021), p. 025016. DOI: 10.1088/2632-2153/abd615.