On cloning machine intelligence

Najeeb Khan[a,c,1], **Mridul Gupta**[b,1,2], **and Debarka Sengupta**[a]

[a]Affiliation One; [b]Affiliation Two; [c]Affiliation Three

This manuscript was compiled on July 30, 2017

Abstract to be worked upon after the whole paper has been written.

Keyword 1 | Keyword 2 | Keyword 3 | ...

Machine learning has rapidly taken over a substantial part of the industry due to its ability to solve rather complex problems. Armed with such exponential growth, it is not far away when this field of study finds widespread application in sensitive areas such as defense and medicine (Seurfert et al. and Magoulas et al.)
With the scope of Machine Learning increasing, it becomes necessary to check for various security issues that can arise when applying a machine learning model to a given problem. Goodfellow et al. have described exploitation of adversarial examples to degrade the confidence of Deep Learning models but there has not been any particular study regarding a general security overview of majority of Machine Learning models.
In this paper, we observe the tendency of a machine learning model to clone a black box model using nothing but random data on binary classification datasets. Heuristics are devised to come up with novel approaches to change random data in such a way that a model when trained on this data can mimic the performance of the black box model. The paper also argues about devising a data-less approach towards learning. The first section of this paper discusses the naïve approach of generating random data and using it to judge the performance of a cloned model on the original dataset. We also describe the general methodology implemented in the cloning procedure. The second section discusses the improvements that can be made to the generation of the random dataset so that the cloned model produces a better accuracy on the datasets into consideration leading to a model that captures the intelligence of the black box. The third section deals with the introduction of equitable class random data. We also compare this method with other approaches discussed so far. The fourth section discusses the performance of different statistical and Deep Learning models on each of the approaches and provides a hypothesis as to why some models are particularly better at being cloned and why others are not. The paper ends with a summary and conclusion along with some future work that we wish to perform to bolster the proposed cloning procedure further.

Naive Approach : Generating Random Data from Zero Centered Normal Distribution

We perform the cloning procedure for a model trained on binary classification dataset using any of the statistical methods available. The approach can be easily applied to a model trained on multi-class classification dataset as well as a regression problem by extrapolating the algorithm. The black box model comprises of any machine learning model that has been trained on a dataset. It should produce a good cross-validation accuracy for the original dataset. Cloning of a black box is achieved by predicting the labels on a random dataset or a dataset generated by applying some heuristics. The cloned model is then further trained on this new collection of random data points and their corresponding labels. To determine whether the cloned model is successfully able to mimic the working of the black box we check for the accuracy of the cloned model on the original dataset. Our primary objective is to increase the accuracy of the cloned model on the original dataset and make it as close as possible to the black box classifier.
Under the naïve approach, the random data is generated from a random normal distribution with zero mean and unary standard deviation. Since a dataset comprises of a number of features, each feature value is generated from this normal distribution. The number of features can be easily determined with the help of the black box as it will only accept data of the form [number of samples x number of features]. The algorithm 1 elucidates the generation of random dataset and the cloning procedure.

We performed this experiment in a controlled environment only a certain combination of statistical models as black box and white box (cloned) models. The accuracy score was determined as the ratio of number of correctly classified samples to total number of samples. The observations of the experiment are described in table 1
It can be observed from table 1 that the black box and white box accuracy has a large difference when the number of features for the dataset increases. The high accuracy for two feature dataset can be attributed to the fact that generating random data from zero centred normal distribution in two dimensions is able to provide sufficient information to the white box about the hyperplane that it successfully approximates the original hyperplane. Approximating the original hyperplane allows the white box to perform accurately on the original dataset.

Figure 1a shows the original Iris Dataset in 2 dimensions

Significance Statement

Authors must submit a 120-word maximum statement about the significance of their research paper written at a level understandable to an undergraduate educated scientist outside their field of specialty. The primary goal of the Significance Statement is to explain the relevance of the work in broad context to a broad readership. The Significance Statement appears in the paper itself and is required for all research papers.

Please provide details of author contributions here.

Please declare any conflict of interest here.

[1]A.O.(Author One) and A.T. (Author Two) contributed equally to this work (remove if not applicable).

[2]To whom correspondence should be addressed. E-mail: author.twoemail.com

Data: blackbox , whitebox, number of features, number of samples
Result: a cloned model
data = array of zeros of size [number of features, number of samples];
for *each feature in the data* **do**
    feature = **normal distribution (mean = 0, standard deviation = 1.0)**;
    label = blackbox.predict(data);
    whitebox.fit(data, label);
    // Checking the performance of the whitebox on original data
    print whitebox.score(original data, original labels);
end

Algorithm 1: Algorithm for cloning model using random data

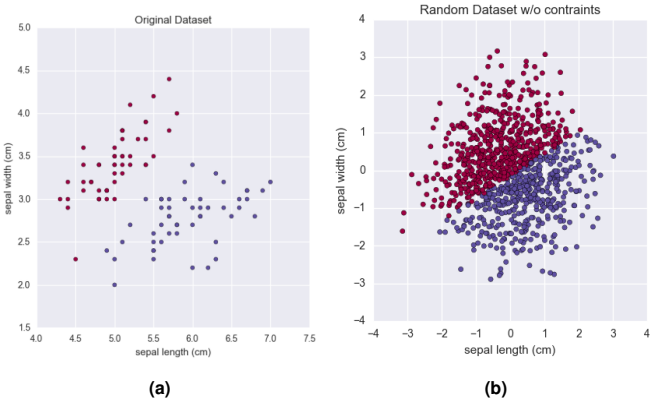

Fig. 1. Iris Dataset in Two Dimensions: (a) Original dataset (b) Randomly generated data

whereas 1b shows the randomly generated dataset using algorithm 1. One can observe that in such low dimensions the cloned model (white box) gets enough information from the random dataset to mimic the hyperplane of the black box. The same can be visualized in the Make Moons nonlinear dataset from Figure 2a and Figure 2b. For higher dimensional data we can observe from the trend of accuracy that this naïve approach tends to perform poorly, only getting around 50 percent of the classification labels correct in its predictions.

Shortcomings of the naïve approach. Some of the shortcomings of the discussed approach are as follows

1. As visualised earlier, the naïve approach does not scale well when the complexity of the data in terms of the number of features is increased. The cloned model accuracy remains lower even when the black box accuracy is quite high for the same statistical model.

2. Since the random data generated is always derived from zero centred normal distribution, the random data won't work for models which were trained on data which does not have zero mean and unary standard deviation. This schism between the data will widen in case of nonlinear classifiers in which the hyperplane may change drastically outside the feature values.

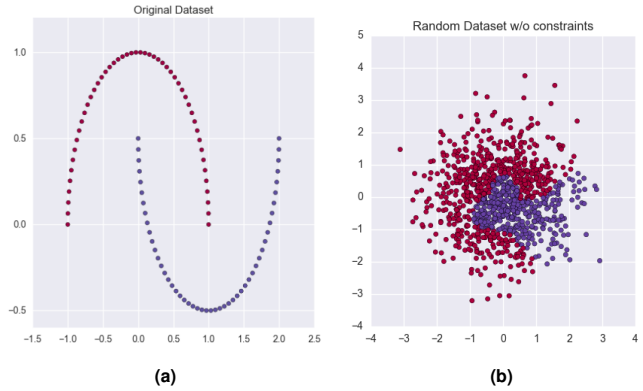

Fig. 2. Make Moons Dataset: (a) Original dataset (b) Randomly generated data

Generating random data under constraints

Machine learning models specifically depend on the data to solve the problem. It is quite common that the data is derived from a natural phenomenon and each of the feature value thus recorded lies under a minimum and maximum value. For example, considering the Iris dataset, the sepal width and sepal length may well be under some specific range. We hypothesise that exploiting this range to generate random data can bolster the performance of our white box even in high dimensions. Algorithm 2 discusses the generation of random data under constraints. The cloning procedure is rather similar and the performance of the white box is judged using the same metric.

Data: blackbox , whitebox, number of features, number of samples, range of features
Result: a cloned model
data = array of zeros of size [number of features, number of samples];
for *each feature in the data* **do**
    feature = **uniform distribution under constraints(range of feature)**;
    label = blackbox.predict(data);
    whitebox.fit(data, label);
    // Checking the performance of the whitebox on original data
    print whitebox.score(original data, original labels);
end

Algorithm 2: Algorithm for cloning model using random data under constraints

The observations are summarized in table 2. It can be easily observed that the black box, as well as white box have almost similar accuracy for all of the datasets. If the black box tends to fail on a given dataset and the similar statistical model is used as the white box then it can be observed that the white box also performs poorly on that dataset. This provides information about the non augmenting nature of cloning. No additional strength is gained by the white box when it is being cloned. The observations favour our hypothesis that random data generated under some constraints helps the cloned model in finding the appropriate position of hyperplane and producing fairly positive results in terms of accuracy.

Figure 3b and figure 4b illustrates the random dataset gen-

Table 1. Cloned model accuracy on original dataset when trained with random dataset

Dataset	No. of Features	Black Box	Black Box Accuracy	White Box	White Box Accuracy
1. Iris	2	Logistic Regression	0.99	Logistic Regression	**0.97**
2. BUPA	7	Logistic Regression	0.59	Logistic Regression	**0.57**
3. Heart	9	Logistic Regression	0.73	Logistic Regression	**0.54**
4. Breast Cancer	30	Logistic Regression	0.95	Logistic Regression	**0.53**
5. Make Moons*	2	Neural Network	0.99	Random Forest	**0.98**

Table 2. Cloned model accuracy on original dataset when trained with random dataset under range constraints

Dataset	No. of Features	Black Box	Black Box Accuracy	White Box	White Box Accuracy
1. Iris	2	Logistic Regression	0.99	Logistic Regression	**0.97**
2. BUPA	7	Logistic Regression	0.59	Logistic Regression	**0.57**
3. Heart	9	Logistic Regression	0.73	Logistic Regression	**0.72**
4. Breast Cancer	30	Logistic Regression	0.95	Logistic Regression	**0.94**
5. Make Moons	2	Neural Network	0.99	Random Forest	**0.98**

Fig. 3. Iris Dataset in Two Dimensions: (a) Original dataset (b) Randomly generated data under constraints

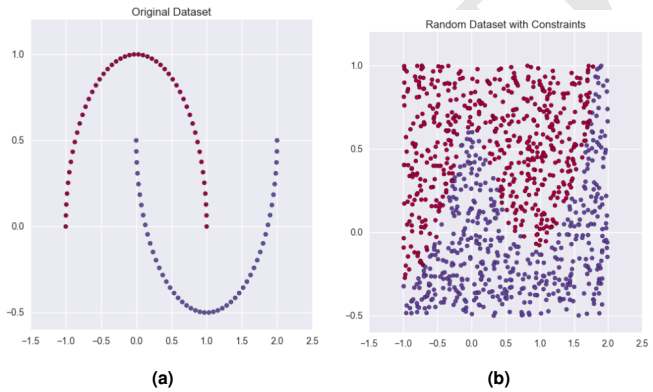

Fig. 4. Make Moons Dataset: (a) Original dataset (b) Randomly generated data under constraints

erated by putting constraints on the data generation procedure. Unlike figure 1b and figure 2b the hyperplane is more clearly visible and can further provide with increased confidence to the model about the position of the hyperplane.

Shortcomings of constrained random dataset. The only shortcoming with this approach is the **assumption that the feature values lie under some kind of constraint**. This assumption may not hold well for dataset where the data is synthetically generated or the feature can have an infinite domain.

Performance analysis of this approach on different classifiers. Performing this experiment on a single type of linear classifier can only guarantee its effectiveness on models that perform classification using any of the parametric approaches i.e by generating the hyperplane. Apart from this, we also need to observe the accuracy of this method on non-parametric models such as Random Forest Classifier. Figure 5 provides some new insights about this approach when it is applied on a dataset with nine features.

It can be observed in figure 5 that Random Forest Classifier when used as a black box and a model to be cloned there exists a rather gap between the classification accuracy. The black box has a high accuracy whereas the white box performs poorly. This observation shows that such models (due to their non-parametric nature) can be difficult to clone and simple hyperplane approximating procedure cannot guarantee accurate cloning. Further observations need to be made to check for white box black box model interaction between Random Forests and other parametric models.

ACKNOWLEDGMENTS. Please include your acknowledgments here, set in a single paragraph. Please do not include any acknowledgments in the Supporting Information, or anywhere else in the manuscript.

*Make Moons is a two dimensional non linear dataset. It was evident from its visualisation that a linear classifier would work poorly on it. So we decided to fit it with a non linear network i.e. a single hidden layer neural network with softmax outputs

Fig. 5. Black box and White box models and their corresponding accuracy