Solving small dataset problem

Amir.M Mousavi.H

Department of Computer Engineering
Shahid Rajee University

Tehran, Iran

AmirMahmood.Mousavi@yahoo.com

Abstract **– It is recognized that real world datasets are typically smaller and sometimes more diverse compared to theoretical studies, the influence of availability of data on training machine learning models has been studied in different manners, which leads the possibility to establish accurate predictive rules using small datasets. In this article, we'll briefly touch on the problems that arise when working with a small dataset. Then, we'll discuss the most effective techniques to overcome these problems.**

Keywords— Small Dataset, Neural Network, SMOTE Generating Data , Artificial Data

I. INTRODUCTION

In a real-world setting, you often only have a small dataset to work with. Models trained on a small number of observations tend to overfit and produce inaccurate results. Learn how to avoid overfitting and get accurate predictions even if available data is scarce.

Big data and data science are concepts often heard together. It is believed that nowadays there are large amounts of data and that data science can draw valuable insights from all these terabytes of information.

However, in a practical scenario, you will often have limited data to solve a problem. Gathering a big dataset can be prohibitively expensive or simply impossible (e.g., only having records from a certain time period when doing time series analysis). As a result, there is often no choice but to work with a small dataset, trying to get as accurate predictions as possible.

Problems of small-data are numerous, but mainly revolve around high variance:

- **Over-fitting** becomes much harder to avoid You don't only over-fit to your training data, but sometimes you over-fit to your validation set as well.
- **Outliers** become much more dangerous. Noise in general becomes a real issue, be it in your target variable or in some of the features.

The goal of a machine learning model is to generalize patterns in training data so that you can correctly predict new data that has never been presented to the model. Overfitting occurs when a model adjusts excessively to the training data, seeing patterns that do not exist, and consequently performing poorly in predicting new data:

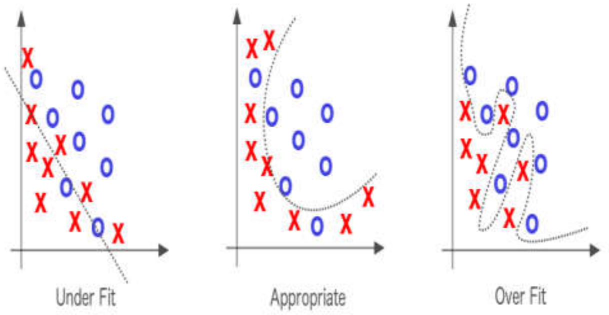


Figure.1.

The fewer samples for training, the more models can fit our data. In an extreme example (a), for just one training point, any model will be able to "explain" it, however simple or complex the model may be. As we get to have more samples (b, c), fewer models are able to explain them:

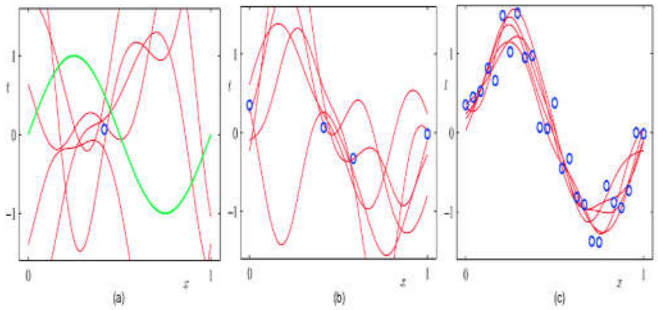


Figure.2.

There are a lot of solution to deal with small dataset for training machine learning algorithms, we have divided the list into 4 sub-topics:

I. Improve Performance With Data.
II. Improve Performance With Algorithms.
III. Improve Performance With Algorithm Tuning.
IV. Improve Performance With Ensembles.

II. METHODS & MATERIALS

In this section we are going to introduce some of the solution of small datasets and how to increase it.

1. Data Augmentation

The performance of deep learning neural networks often improves with the amount of data available.

Data augmentation is a technique to artificially create new training data from existing training data. This is done by applying domain-specific techniques to examples from the training data that create new and different training examples.

Image data augmentation is perhaps the most well-known type of data augmentation and involves creating transformed versions of images in the training dataset that belong to the same class as the original image.

Transforms include a range of operations from the field of image manipulation, such as shifts, flips, zooms, and much more. In figure.3 the example has been shown.

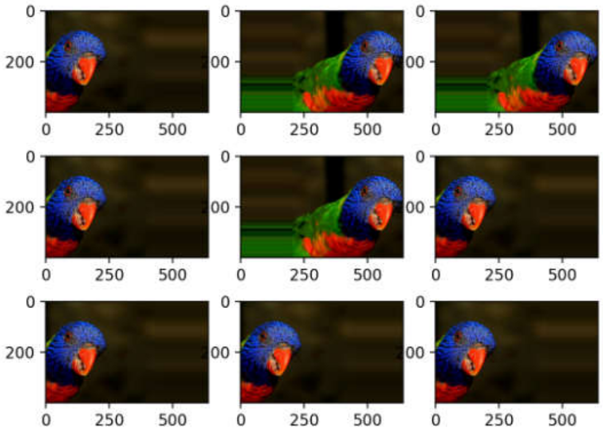


Figure.3. example of data augmentation (random shift)

2. Rescaling Data

A traditional rule of thumb when working with neural networks is: Rescaling your data to the bounds of your activation functions.
If you are using sigmoid activation functions, rescale your data to values between 0-and-1. If you’re using the Hyperbolic Tangent (tanh), rescale to values between -1 and 1.
This applies to inputs (x) and outputs (y). For example, if you have a sigmoid on the output layer to predict binary values, normalize your y values to be binary. If you are using softmax, you can still get benefit from normalizing your y values. This is still a good rule of thumb, but you would go further and create a few different versions of new training dataset as follows:

- Normalized to 0 to 1.
- Rescaled to -1 to 1.
- Standardized.

3. Synthetic data

Let's look at the distribution of target values in figure.4. In addition to being extremely small, our training dataset has the unbalanced target binary variable, which can undermine some models' predictability. We will perform an oversampling, which consists of creating new samples to increase the 0 minority class. For this we will use the SMOTE technique.

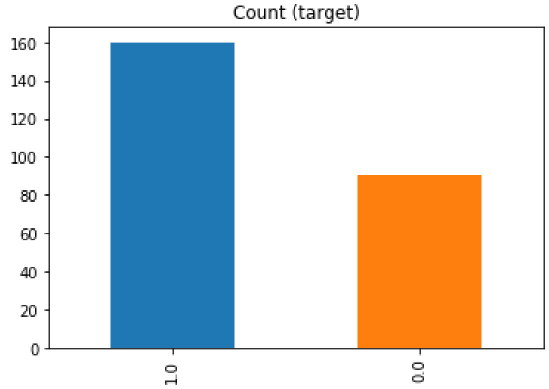


Figure.4. distribution of target values

SMOTE (Synthetic Minority Oversampling Technique) consists of synthesizing elements for the minority class, based on those that already exist. It works randomly picking a point from the minority class and computing the k-nearest neighbors for this point. The synthetic points are added between the chosen point and its neighbors.

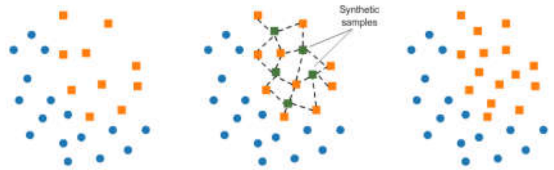


Figure.5.example of SMOTE

III. EXPERIMENTAL RESULTS

we implemented data augmentation on ((cat vs dog)) dataset and rescaled the dataset in order to test the results:

1. without rescaling and data augmentation:

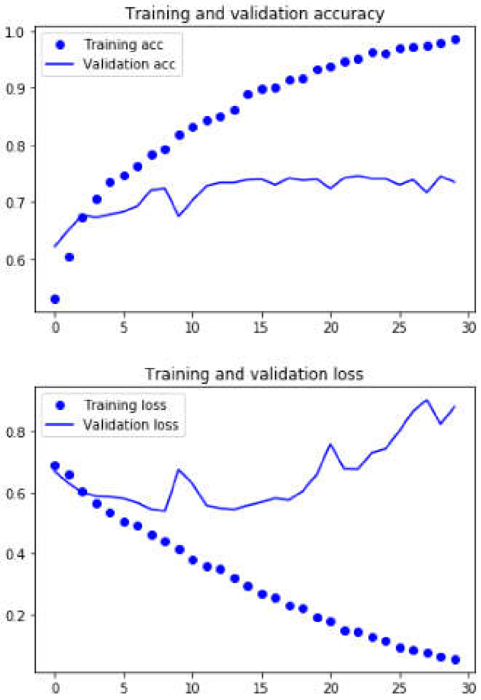


Figure.6. accuracy and loss of training and validation

2. **with rescaling and without data augmentation:**

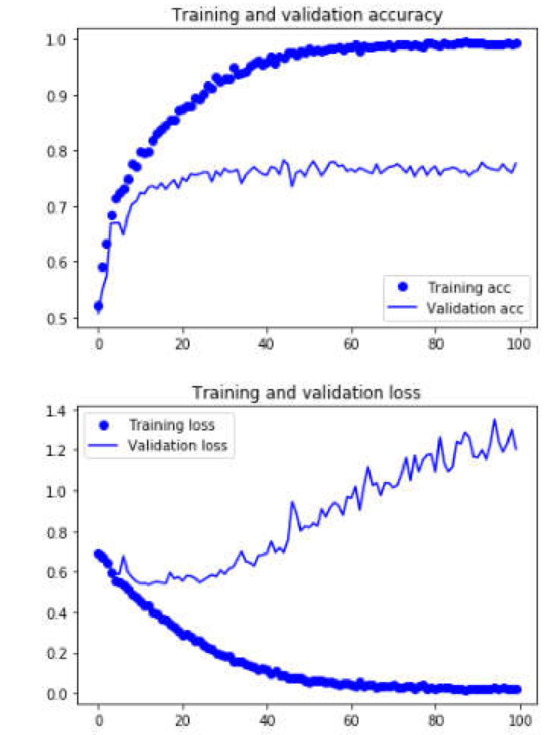


Figure.7. accuracy and loss of training and validation

3. **with rescaling and data augmentation:**

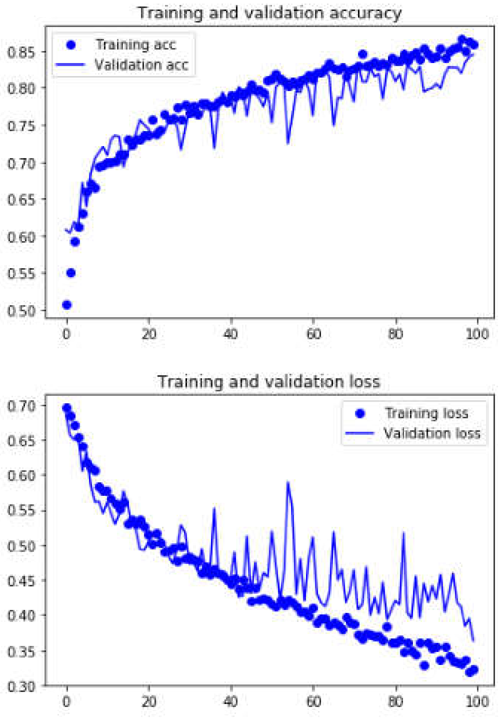


Figure.8. accuracy and loss of training and validation

As shown, rescaling couldn't effect very much, but data augmentation could prevent overfitting and achieve a good performance. Data augmentation's information is:
rescale=1./255,
rotation_range=40,
width_shift_range=0.2
height_shift_range=0.2,
shear_range=0.2,
zoom_range=0.2,
horizontal_flip=True

Some example of data augmentation is shown in figure.9.

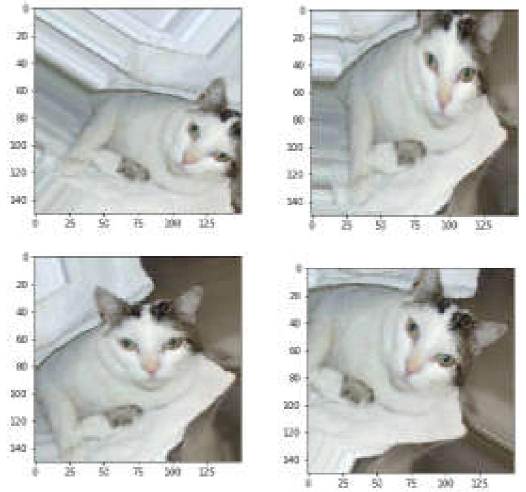


Figure.9. example of cat

For the next step we implemented synthetic data and SMOTE in an Artificial data to observe the result :
First we attended to regression problem we noisy and sparse data :

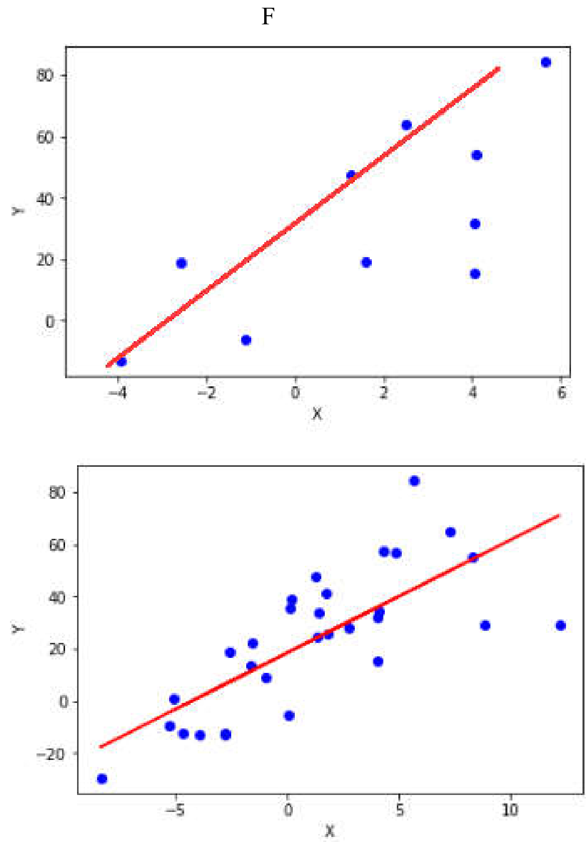
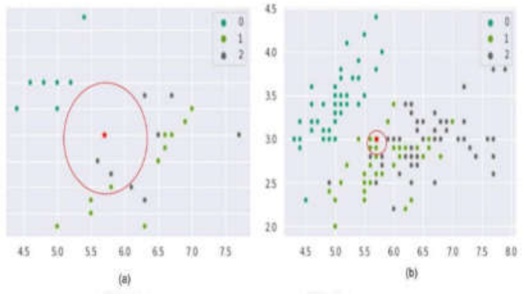


figure.10. result of regression

For more example we can point to KNN classifiers that increasing data can be useful.



As we can see, it has improved the results and got closed to the optimal point.

For the next step we went for STOME on artificial data.
In figure.11 the original data set is shown:

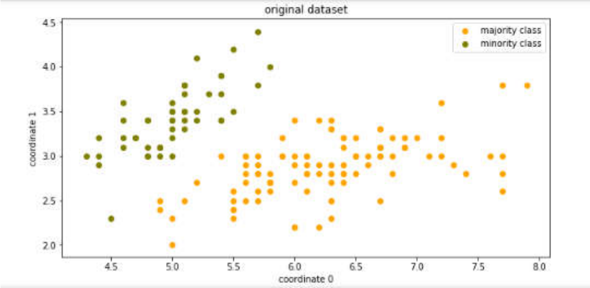
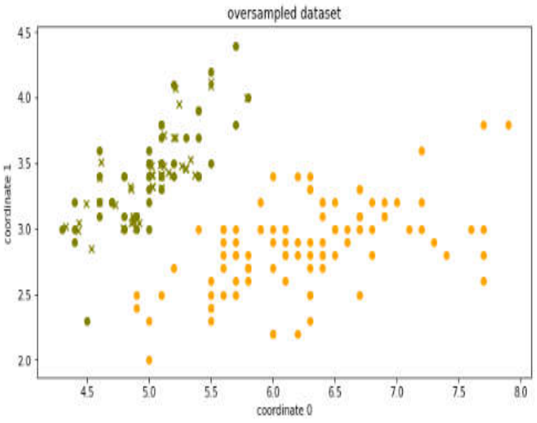
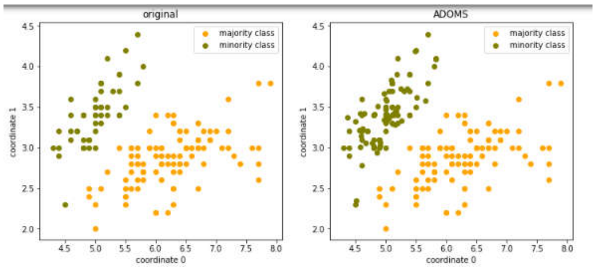


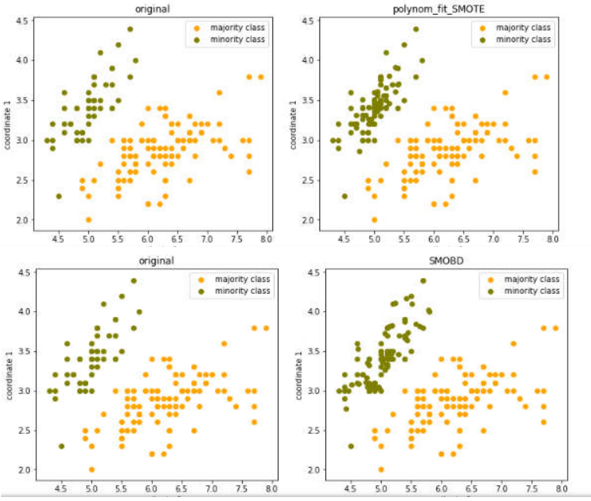
Figure.11. original dataset

Then we applied SMOTE for the class green in order to balance data for both classes. The multiplication signs are the new data.



We went further and applied more libraries of SMOTE and here is the some of them.





IV. CONCLUSION

In the discipline of machine learning, data science is a quickly moving field. In this paper an attempt is made to the small datasets problems. Here we discussed what are the problems and analyzed a few solution. We saw for the images, data augmentation is a very good solution, we also tested synthetic data and saw how the can be effective for regression or classifiers like KNN. For future work, we can focus on GAN networks.

REFERENCES

[1] ow to handle Imbalanced Classification Problems in machine learning?

[2] What to do with “small” data?

[3] How to Handle Imbalanced Classes in Machine Learning

[4] Small Data & Deep Learning (AI) — A Data Reduction Framework

[5] Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates