ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

# Estimating Formulas for Model Performance Under Noisy Labels Using Symbolic Regression

Fech Scen Khoo, Dawei Zhu, Michael A. Hedderich, Dietrich Klakow *

Saarland University, Saarland Informatics Campus, Germany

**Abstract**. We present a generic formula characterizing the learning of our model under a variety of label-noise settings. This is achieved by using the symbolic regressor model, a genetic programming algorithm, from which we learn functions based on a large set of performance evaluations. Equipped with the knowledge from the regressor, we find a universal formula governing the model performance with respect to noise. This result from our empirical approach could have qualitative applications in mitigating the performance of real-world noisy data and could complement certain noise-robust models.

## 1 Introduction

The accuracy of classification relies on the correctness of the available data labels. For many applications, high quality labels are lacking. Nonetheless, cheap labels can be obtained by crowd sourcing or by distant supervision. However, noises can arise from human annotations via these crowd-sourcing platforms [1], and data-labelling methods such as weakly supervised learning employed when (ground-truth) labels are scarce [2]. Therefore, training models under noise has become an important enterprise.

How does a model's performance change when there are noises in the dataset? Intuitively, a higher noise level will result in a poorer generalization by the model. Would the performance deteriorate in a linear fashion, or non-linearly? To learn the relation between system performance and noise level, we use a symbolic regressor. A symbolic regressor tackles regression problems using a genetic programming algorithm [3]. It assumes a model structure to be an algebraic expression. Genetic operations are performed to discover the underlying structure in a mathematical form. The idea for using logistic regression and simulation originates in [4]. Symbolic regressor as a powerful tool is used to learn theoretical formulas for complex problems which might otherwise be non-trivial to derive and prove.

This paper attempts to bridge the gap in the understanding of noise level and system performance. We begin by first creating artificial noise guided by conditional probabilities. Next we learn a mathematical relation between our model performance and the amount of noise that we inject into the dataset. By measuring the model performance under the synthetic noises in a controlled way, this approach could provide potential insights into its possible generalization in

*This work has been partially funded by the EU Horizon 2020 project ROXANNE under grant number 833635.

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

a real-world setting. We choose $F1$-score to quantify our model performance for our Named Entity Recognition (NER) task. As $F1$-score takes a value between 0 and 1, one can already hypothesize that a model performance must be described by a bounded function, not a solely monotonous function.

Our main contribution is the discovery of an algebraic expression generically relating the model performance and noise level, independent of the noise types. By leveraging the mathematical formula we can estimate the model performance across all noise levels. Perhaps more exciting is the other way round, when one can infer the amount of noise (and possible noise type) in the dataset knowing the performance of the model. Apart from that, there exist a number of theoretical work on the topic of noisy labels and noise-robust learning algorithms such as [5] which requires an apriori knowledge of noise levels. Our finding could complement such approaches, where the noise rates will be informed by our formula.

## 2 Background and Experimental Setup

For synthetic noise generation, we consider the following three noise types. For $k$ classes of label, we flip the true (or clean) label $y$ to create a noisy label $\hat{y}$, according to the conditional probabilities which are characterized by a noise level $\epsilon$. The probabilities can be represented in terms of noise matrices $M$.
Uniform noise [6],

$$M^{uni}_{i,j} = p_{uni}(\hat{y} = j|y = i) = \begin{cases} 1-\epsilon\ , & i = j \\ \frac{\epsilon}{k-1}\ , & i \neq j\ . \end{cases}$$

Single label-flip noise [7],

$$M^{sf}_{i,j} = p_{sf}(\hat{y} = j|y = i) = \begin{cases} 1-\epsilon\ , & i = j \\ \epsilon\ , & \text{one } i \neq j \\ 0\ , & \text{otherwise}\ . \end{cases}$$

Multi-flip noise [8], [9],

$$\sum_{j=1}^{k} M^{mf}_{i,j} = \sum_{j=1}^{k} p_{mf}(\hat{y} = j|y = i) = 1\ , \quad p_{mf}(\hat{y} = j|y = i) \geq 0\ , \quad \forall i, j\ .$$

We examine our model performance on the CoNLL2003 (English) dataset [10]. There are five labels in our NER task: {O, PER, ORG, LOC, MISC}, where each takes the value of 0 to 4 respectively. Following [9] we use a bi-LSTM based model for our task.

First we measure the performance of our model ($F1$) on different noise levels ($\epsilon$). We consider 10 varied noise settings in a range of noise levels from 0.01 to 0.99. 80% of these results serve as the training dataset and the remaining 20% for testing, for our symbolic regression task. The performance of the regressor is evaluated based on the training mean-squared-error (MSE), test MSE, and coefficient of determination $R^2$.

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

| 1. Flip labels 1 & 2 ($f12$) | 2. Flip labels 1 & 4 ($f14$) |
|---|---|
| $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1-\epsilon & \epsilon & 0 & 0 \\ 0 & \epsilon & 1-\epsilon & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$ | $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1-\epsilon & 0 & 0 & \epsilon \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & \epsilon & 0 & 0 & 1-\epsilon \end{pmatrix}$ |
| 3. Consecutive single-flip ($cs$) | 4. Uniform noise ($uni$) |
| $\begin{pmatrix} 1-\epsilon & \epsilon & 0 & 0 & 0 \\ 0 & 1-\epsilon & \epsilon & 0 & 0 \\ 0 & 0 & 1-\epsilon & \epsilon & 0 \\ 0 & 0 & 0 & 1-\epsilon & \epsilon \\ \epsilon & 0 & 0 & 0 & 1-\epsilon \end{pmatrix}$ | $\begin{pmatrix} 1-\epsilon & \epsilon/4 & \epsilon/4 & \epsilon/4 & \epsilon/4 \\ \epsilon/4 & 1-\epsilon & \epsilon/4 & \epsilon/4 & \epsilon/4 \\ \epsilon/4 & \epsilon/4 & 1-\epsilon & \epsilon/4 & \epsilon/4 \\ \epsilon/4 & \epsilon/4 & \epsilon/4 & 1-\epsilon & \epsilon/4 \\ \epsilon/4 & \epsilon/4 & \epsilon/4 & \epsilon/4 & 1-\epsilon \end{pmatrix}$ |
| 5. Label 0 unchanged + others in uniform noise ($0uni$) | 6. Label 0 & 1 unchanged + others in uniform noise ($01uni$) |
| $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1-\epsilon & \epsilon/3 & \epsilon/3 & \epsilon/3 \\ 0 & \epsilon/3 & 1-\epsilon & \epsilon/3 & \epsilon/3 \\ 0 & \epsilon/3 & \epsilon/3 & 1-\epsilon & \epsilon/3 \\ 0 & \epsilon/3 & \epsilon/3 & \epsilon/3 & 1-\epsilon \end{pmatrix}$ | $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1-\epsilon & \epsilon/2 & \epsilon/2 \\ 0 & 0 & \epsilon/2 & 1-\epsilon & \epsilon/2 \\ 0 & 0 & \epsilon/2 & \epsilon/2 & 1-\epsilon \end{pmatrix}$ |
| 7. Alternate single-flip ($alt$) | 8. A multi-flip ($mult$) |
| $\begin{pmatrix} 1-\epsilon & 0 & \epsilon & 0 & 0 \\ 0 & 1-\epsilon & 0 & \epsilon & 0 \\ 0 & 0 & 1-\epsilon & 0 & \epsilon \\ \epsilon & 0 & 0 & 1-\epsilon & 0 \\ 0 & \epsilon & 0 & 0 & 1-\epsilon \end{pmatrix}$ | $\begin{pmatrix} 0.1 & 0.01 & 0.09 & 0.3 & 0.5 \\ 0 & 0.9 & 0 & 0.05 & 0.05 \\ 0 & 0 & 1-\epsilon & 0 & \epsilon \\ 0.02 & 0 & 0 & 0.03 & 0.95 \\ 1-\epsilon & 0 & 0 & \epsilon & 0 \end{pmatrix}$ |
| 9. Flip labels 3 & 4 ($f34$) | 10. Flip labels 1 & 3 ($f13$) |
| $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1-\epsilon & \epsilon \\ 0 & 0 & 0 & \epsilon & 1-\epsilon \end{pmatrix}$ | $\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1-\epsilon & 0 & \epsilon & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & \epsilon & 0 & 1-\epsilon & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$ |

Table 1: Noise matrices studied.

In particular, we consider the following 10 noise representations (Table 1).

Fig. 1 shows an overview of the model behaviour under the various noise representations considered, as described in Table 1. Typically, the performance decays drastically when the noise level is higher than 0.5. Among all the noise settings, the range of $F1$ performance is the smallest for our choice of a multi-flip noise, depicted in Fig. 2.

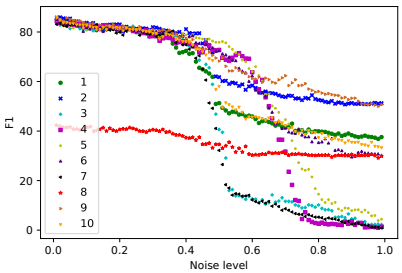


Fig. 1: 10 noise settings as numbered in Table 1.

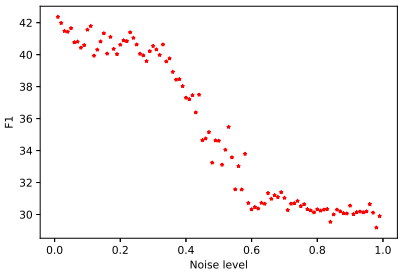


Fig. 2: A multi-flip noise (noise 8 in Fig. 1, shown here for better visibility.)

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

## 3 Regression Results

The performance results are first modelled by the symbolic regressor. Then we refine the results further by manual fitting, where we obtain a single mathematical description.

### 3.1 Symbolic Regressor Model

We simplified the formulas from the regressor as a post processing step. From the symbolic regressor [3], we obtain two types of expressions, depending on the noise setting,

$$F1_{type\,1} = \alpha_1\,\epsilon + \alpha_2 + \alpha_3\tanh(\alpha_4\,\epsilon + \alpha_5)$$

$$F1_{type\,2} = \beta_1\tanh(\beta_2\,\epsilon - \beta_3) + \beta_4\tanh(\beta_5\,\epsilon - \beta_6) + \beta_7$$

with constants $\alpha_m, \beta_n \in \mathbb{R}$, $m = 1, .., 5$, $n = 1, .., 7$. The complete results in Table 2 are shown up to two decimal places for the corresponding $\alpha_m$ and $\beta_n$ values.

| Noise | $\alpha_1$ | $\alpha_2$ | $\alpha_3$ | $\alpha_4$ | $\alpha_5$ | MSE (test) |
|---|---|---|---|---|---|---|
| $f12$ | $-6$ | 63.89 | $-19$ | 12 | $-5.62$ | 3.24 |
| $f14$ | 10 | 61.97 | $-20$ | 7 | $-3.59$ | 2.12 |
| $f13$ | $-6$ | 62.59 | $-20$ | 15 | $-7.59$ | 6.80 |
| $01uni$ | $-13$ | 64.32 | $-19$ | 12 | $-7.30$ | 2.86 |
| $alt$ | 2 | 42.62 | $-40$ | 9 | $-4.37$ | 4.47 |
| $f34$ | 1 | 67.18 | $-20$ | 3 | $-1.62$ | 2.18 |

| Noise | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ | MSE (test) |
|---|---|---|---|---|---|---|---|---|
| $cs$ | $-19$ | 12 | $-6.00$ | $-19$ | 24 | $-11.60$ | 43.71 | 9.03 |
| $uni$ (Fig. 3) | $-20$ | 9 | $-5.66$ | $-20$ | 18 | $-12.63$ | 40.00 | 4.96 |
| $0uni$ | $-19$ | 12 | $-7.30$ | $-19$ | 12 | $-9.11$ | 43.53 | 4.25 |

Table 2: Results of the equation parameters and test MSE from the symbolic regressor.

A multi-flip (noise 8):
$F1_{mult1} = 29.59 - 10\tanh(3\,\epsilon - 0.92) + 10\tanh(3\,\epsilon + 0.58)$
$F1_{mult2} = 35.74 + 2\,\epsilon - 9\tanh(3\,\epsilon - 1.28)$

Notice that there is no linear term predicted in the type 2 expression. Meanwhile, it occurs that for the case of the multi-flip noise setting, the symbolic regressor can produce two equivalently well-fitted formulas: $F1_{mult1}$ with an $R^2$-score $\approx 0.943$ (test MSE $\approx 1.37$), and $F1_{mult2}$ with an $R^2$-score $\approx 0.935$ (test MSE $\approx 1.55$), where one of them comes with a linear term and the other without.

### 3.2 A Manual Fit

Having learned of an occurring pattern in the expressions from the symbolic regressor, let us now fit our model performance by using only

$$F1 = -\alpha\,\epsilon + \beta - \gamma\tanh(\lambda\,\epsilon - \kappa)\ ,$$

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

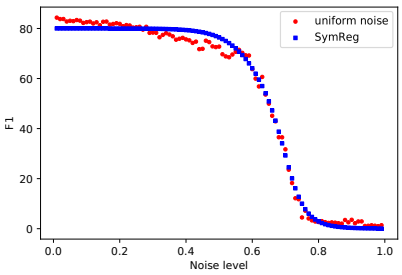


Fig. 3: A fit for noise *uni* by the symbolic regressor.

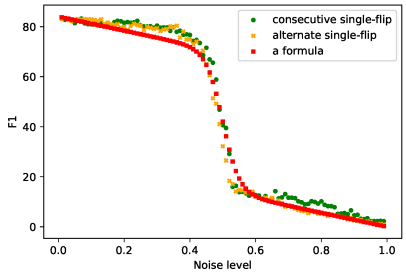


Fig. 4: Noise *cs* and *alt* under our manual fitting.

with positive constants $\alpha, \beta, \gamma, \lambda, \kappa \in \mathbb{R}_{>0}$. We have the freedom to tailor these parameters. The complete results are given in Table 3.

| Noise | $\alpha$ | $\beta$ | $\gamma$ | $\lambda$ | $\kappa$ | MSE |
|---|---|---|---|---|---|---|
| $f12$ | 7 | 63 | 19 | 8 | 4 | 6.06 |
| $f13$ | 7 | 63 | 19 | 8 | 4 | 6.4 |
| $f14$ | 11 | 73 | 11 | 14 | 7 | 2.76 |
| $cs$ | 28 | 56 | 28 | 20 | 10 | 11.27 |
| $alt$ | 28 | 56 | 28 | 20 | 10 | 10.19 |
| $uni$ | 30 | 56 | 30 | 19 | 13 | 5.58 |
| $0uni$ | 18 | 54 | 30 | 13 | 9 | 5.42 |
| $01uni$ | 13 | 65 | 19 | 12 | 7 | 6.76 |
| $f34$ | 1 | 69 | 18 | 3 | 1.5 | 2.91 |
| $mult$ | 6 | 38 | 4 | 8 | 4 | 1.06 |

Table 3: Results of the equation parameters and MSE from our manual fitting.

We found a good single description for the two cases of flips denoted by $f12$ and $f13$, and also one for the consecutive and alternate single-flips. We remark that there are two reasonable solutions found for the alternate single-flip case. Besides the one shared with the consecutive single-flip, $F1_{alt2} = -17\,\epsilon + 51 - 34\tanh(16\,\epsilon - 8)$ presents a good fit as well with a $\text{MSE}_{alt2} \approx 10.26$. Note the differences in the constants. $F1_{alt2}$ will require more parameters to be specified than $F1_{cs,alt}$.

An important finding here is that the parameters in the argument of the *tanh* term are related by a factor of 2 when the model performance looks symmetrical (Fig. 4). On the contrary, results from the uniform noise, "label 0 unchanged + others in uniform noise", and "label 0 & 1 unchanged + others in uniform noise" do not possess such property.

Finally, depending on the noise, the expressions are parametrized by one to five parameters. For example,
$F1_{01uni} = -\alpha\,\epsilon + \beta - \gamma\tanh(\lambda\,\epsilon - \kappa)$, where $\alpha = 13, \beta = 19, \gamma = 12, \lambda = 7, \kappa = 65$;
$F1_{f12,f13} = -\alpha\,\epsilon + 9\alpha - \gamma\tanh(2\kappa\,\epsilon - \kappa)$, where $\alpha = 7, \gamma = 19, \kappa = 4$;
$F1_{0uni} = 2\,(-3\alpha\,\epsilon + 9\alpha - 5\alpha\tanh(\lambda\,\epsilon - 3\alpha))$, where $\alpha = 3, \lambda = 13$;

ESANN 2021 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 6-8 October 2021, i6doc.com publ., ISBN 978287587082-7.
Available from http://www.i6doc.com/en/.

$F1_{cs,alt} = 14(-\alpha\,\epsilon + 2\alpha - \alpha\tanh(10\alpha\,\epsilon - 5\alpha))$, where $\alpha = 2$.

## 4 Conclusions

For the CoNLL03 dataset under the 10 chosen noise settings, our model performance can be satisfactorily described by 8 expressions (with the corresponding values for the parameters) mapping noise level $\epsilon$ to $F1$-score, in the form of

$$F1 = -\alpha\,\epsilon + \beta - \gamma\tanh(\lambda\,\epsilon - \kappa)\ .$$

It is composed of a bounded function ($tanh$) as well as a monotonic function (the linear component). The performance declines gradually in the beginning then substantially. Led by the results from the symbolic regressor, we are able to find such a universal relation. Furthermore, when the model performance presents a mirror symmetry around the mid range of the noise level, the general expression is simplified with $\lambda = 2\kappa$. Therefore, if this special property is fulfilled, one can rule out the existence of a uniform noise-like contamination in the data.

The applicability of our current findings to other datasets and tasks is left for future work. More importantly this formula should help us in developing a theory for the relation of noise level and system performance.

## References

[1] A. Khetan, Z. C. Lipton and A. Anandkumar, Learning From Noisy Singly-labeled Data. *CoRR*, abs/1712.04577, 2017.

[2] M. A. Hedderich, L. Lange, H. Adel, J. Strötgen and D. Klakow, A Survey on Recent Approaches for Natural Language Processing in Low-Resource Scenarios. In *Proceedings of NAACL 2021*.

[3] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT Press, Cambridge, MA, USA, 1992.

[4] G. Bettgenhäuser, M. A. Hedderich and D. Klakow, Learning Functions to Study the Benefit of Multitask Learning. arXiv:2006.05561 [cs.LG], 2020.

[5] N. Natarajan, I. S. Dhillon, P. Ravikumar and A. Tewari, Learning with Noisy Labels. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013*, 1196-1204, 2013.

[6] J. Larsen, L. Nonboe, M. Hintz-Madsen and L. K. Hansen, Design of robust neural network classifiers. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, 1205-1208, 1998.

[7] S. E. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan and A. Rabinovich, Training Deep Neural Networks on Noisy Labels with Bootstrapping. In *ICLR 2015, Workshop Track Proceedings*, 2015.

[8] A. J. Bekker and J. Goldberger, Training deep neural-networks based on unreliable labels. In *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, doi:10.1109/ICASSP.2016.7472164, 2016.

[9] M. A. Hedderich, D. Zhu and D. Klakow, Analysing the Noise Model Error for Realistic Noisy Label Data. In *Proceedings of AAAI 2021*.

[10] E. F. T. K. Sang and F. De Meulder, Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 142-147, 2003.