11-1-2022

Anomaly Detection in Crowdsourcing: Why Midpoints in Interval-Valued Approach

Alejandra De La Pena
The University of Texas at El Paso, adelapena5@miners.utep.edu

Damian L. Gallegos Espinoza
The University of Texas at El Paso, dlgallegose@miners.utep.edu

Vladik Kreinovich
The University of Texas at El Paso, vladik@utep.edu

Comments:
Technical Report: UTEP-CS-22-112

Anomaly Detection in Crowdsourcing: Why Midpoints in Interval-Valued Approach

Alejandra De La Peña, Damian L. Gallegos Espinoza, and Vladik Kreinovich

Abstract In many practical situations – e.g., when preparing examples for a machine learning algorithm – we need to label a large number of images or speech recordings. One way to do it is to pay people around the world to perform this labeling; this is known as crowdsourcing. In many cases, crowd-workers generate not only answers, but also their degrees of confidence that the answer is correct. Some crowd-workers cheat: they produce almost random answers without bothering to spend time analyzing the corresponding image. Algorithms have been developed to detect such cheaters. The problem is that many crowd-workers cannot describe their degree of confidence by a single number, they are more comfortable providing an interval $[\underline{x}, \overline{x}]$ of possible degrees. To apply anomaly-detecting algorithms to such interval data, we need to select a single number from each such interval. Empirical studies have shown that the most efficient selection is when we select the arithmetic average. In this paper, we explain this empirical result by showing that arithmetic average is the only selection that satisfies natural invariance requirements.

1 Formulation of the Problem

What is crowdsourcing: a brief reminder. In many practical situations, we need to perform a large number of reasonably simple tasks, tasks that do not require high qualifications. For example, deep learning requires that a large number of labeled examples by available (see, e.g., [1]). In many cases, we do not have that many labeled examples, so we need someone to label a large number of photos, or a large number of speech recordings. One way to perform these tasks is *crowdsourcing*,

Alejandra De La Peña, Damian L. Gallegos Espinoza, and Vladik Kreinovich
Department of Computer Science, University of Texas at El Paso, 500 W. University
El Paso, Texas 79968, USA, e-mail: adelapena5@miners.utep.edu,
dlgallegose@miners.utep.edu, vladik@utep.edu

when people all over the world are paid to solve the corresponding tasks – e.g., to label pictures that will be used for training a machine learning algorithm.

Need to detect anomalies. Most crowd-workers work conscientiously. However, since the payment is proportional to the number of answers, there are also many cases when crowd-workers do a sloppy job, not spending enough time on analyzing the corresponding picture and therefore producing answers that are often wrong. Such wrong answers prevent machine learning algorithms from getting high quality results. It is therefore important to be able to detect such anomalous crowd-workers and dismiss their answers.

A natural way to do it is to include examples with known labels into the list of tasks. Then, we can gauge the quality of a crowd-worker by the number of wrong answers that he/she has on these examples. If this number is unusually high, then all the answers provided by this crowd-worker should be dismissed.

Need to take into account degrees of confidence. Crowd-workers are often not 100% confident in their answers. To help machine learning, it is therefore desirable to collect not only the answers, but also the degrees indicating how confident is the crowd-worker in each answer. This way, the neural network will be able to weigh these answers with different weights: if its answer is different from the confident answer of a crowd-worker, then the algorithm should continue training, but if the difference is only with not very confident crowd-workers, then maybe there is no need to adjust.

Because of this, some crowdsourcing algorithms require the crowd-worker to submit not only the answer, but also his/her degree of confidence in this answer – as expressed by a number on some scale $[\underline{X},\overline{X}]$, e.g., from 0 to 10, or from 0 to 1. Usually, larger numbers correspond to larger degrees of confidence.

Usually, linear transformations are used to transform between different scales. For example, the value 7 on a scale from 0 to 10 is transformed into 7/10 on the scale from 0 to 1. Similarly, the value 0 on the scale $[-1,1]$ is transformed into the value 0.5 on the scale $[0,1]$.

These degrees of confidence are used to detect anomalies: if the answer is wrong but the crowd-worker is not very confident about it, this may be an honest mistake, but if there are many wrong answers with high degrees of confidence, this indicates an anomaly. Sometimes, these degrees also affect the amount of payment: the higher degree of confidence, the higher the pay – since one way to gain more confidence is to spend more time analyzing the corresponding picture or recording.

Interval-valued degrees of confidence. Crowd-workers are usually unable to describe their degree of confidence by a single number: in general, people cannot meaningfully distinguish, e.g., between degrees of confidence 0.70 and 0.71 on a scale from 0 to 1. So, it makes sense to allow the crowd-workers to mark their confidence by selecting an interval $[\underline{x},\overline{x}]$ of possible degrees: e.g., an interval $[0.7,0.8]$.

How to detect anomalies based on interval-valued degrees: formulation of the problem. A natural idea is to utilize formulas that have been successful in detecting anomalies based on numerical degrees. To apply these formulas, we need to select a single value x from the corresponding interval $[\underline{x}, \overline{x}]$. In other words, we need an algorithm $x = f(\underline{x}, \overline{x})$ that generates a number based on the bounds of the worker-generated interval.

Which algorithm $f(\underline{x}, \overline{x})$ should we select? We can take arithmetic average, we can take geometric average $\sqrt{\underline{x} \cdot \overline{x}}$, we can have many other choices.

An empirical analysis described in [2] has shown that the more accurate anomaly detection happens when we use arithmetic average

$$\frac{\underline{x} + \overline{x}}{2}.$$

How can we explain this empirical result?

What we do in this paper. In this paper, we use natural invariances to explain this empirical result.

2 Our Explanation

Invariance. We can have different scales, so it is reasonable to require that the desired algorithm $x(\underline{x}, \overline{x})$ should not change if we apply some linear transformation to a different scale. Thus, we arrive at the following definition.

Definition 1. *We say that a function $f(\underline{x}, \overline{x})$ is* scale-invariant *if for every linear transformation $x \mapsto a + b \cdot x$ with $b > 0$, and for all possible values $\underline{x} < \overline{x}$, once we have $x = f(\underline{x}, \overline{x})$, then we should also have $y = f(\underline{y}, \overline{y})$, where $y = a + b \cdot x$, $\underline{y} = a + b \cdot \underline{x}$, and $\overline{y} = a + b \cdot \overline{x}$.*

Additional requirement related to negation. In situations where there are only two possible choices A and B, if we use the scale from 0 to 1, then one way to interpret the degree of confidence x is as the probability that the correct choice is A. This same situation can be interpreted as the probability $1-x$ that the correct choice is B.

If, instead of the exact probability x, we have an interval $[\underline{x}, \overline{x}]$ of possible values of A-probability, then the corresponding values of B-probability $1-x$ form an interval $[1 - \overline{x}, 1 - \underline{x}]$.

- We can apply the desired function to the original interval $[\underline{x}, \overline{x}]$ and thus get some probability x.
- Alternatively, we can apply the same function to the negation-related interval $[1 - \overline{x}, 1 - \underline{x}]$ and get some probability y – in which case, for the probability of A, we get the value $1 - y$.

Since these are two ways to describe the same situation, it is reasonable to require that we should get the same probability, i.e., that we should have $x = 1 - y$. Thus, we arrive at the following definition.

Definition 2. *We say that a function* $f(\underline{x},\overline{x})$ *is* negation-invariant *if for all possible values* $0 \le \underline{x} < \overline{x} \le 1$*, once we have* $x = f(\underline{x},\overline{x})$*, then we should also have* $y = f(\underline{y},\overline{y})$*, where* $y = 1 - x$*,* $\underline{y} = 1 - \overline{x}$*, and* $\overline{y} = 1 - \underline{x}$*.*

Proposition. *The only scale-invariant and negation-invariant function* $f(\underline{x},\overline{x})$ *is arithmetic average*

$$f(\underline{x},\overline{x}) = \frac{\underline{x}+\overline{x}}{2}.$$

Proof. It is easy to check that arithmetic average is scale-invariant and negation-invariant. Let us prove that, vice versa, every scale-invariant and negation-invariant function $f(\underline{x},\overline{x})$ is arithmetic average.

Indeed, let us denote $f(0,1)$ by α. Here, $\underline{x} = 0$, $\overline{x} = 1$, and $x = \alpha$. Then, for every two numbers $x_1 < x_2$, we can take $a = x_1$ and $b = x_2 - x_1$. In this case,

$$\underline{y} = a + b \cdot \underline{x} = x_1,$$

$$\overline{y} = a + b \cdot \overline{x} = x_1 + (x_2 - x_1) = x_2, \text{ and}$$

$$y = a + b \cdot x = x_1 + \alpha \cdot (x_2 - x_1) = \alpha \cdot x_2 + (1-\alpha) \cdot x_1.$$

Thus, due to scale-invariance, we conclude that

$$f(x_1,x_2) = \alpha \cdot x_2 + (1-\alpha) \cdot x_1. \quad (1)$$

To find α, let us now use negation invariance. According to this property, we should have $f(1-x_2, 1-x_1) = 1 - f(x_1,x_2)$. Substituting the expression (1) into this formula, we conclude that

$$\alpha \cdot (1-x_1) + (1-\alpha) \cdot (1-x_2) = 1 - \alpha \cdot x_2 - (1-\alpha) \cdot x_1.$$

If we open parentheses, we conclude that

$$1 - \alpha \cdot x_1 - (1-\alpha) \cdot x_2 = 1 - \alpha \cdot x_2 - (1-\alpha) \cdot x_1.$$

The two linear functions on both sides of this formula should be equal to all $x_1 < x_2$. Thus, the coefficients at x_1 must coincide, so $\alpha = 1 - \alpha$ and thus, $\alpha = 1/2$ – and therefore, the formula (1) becomes arithmetic average.

The proposition is proven.

Conclusion. We have explained why arithmetic average works well: it is the only function that satisfies natural invariance requirements.

Acknowledgments

This work was supported in part by the National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science), and HRD-1834620 and HRD-2034030 (CAHSI Includes), and by the AT&T Fellowship in Information Technology.

It was also supported by the program of the development of the Scientific-Educational Mathematical Center of Volga Federal District No. 075-02-2020-1478, and by a grant from the Hungarian National Research, Development and Innovation Office (NRDI).

One of the authors (VK) is thankful to all the participants of the 19th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'2022 (Milan, Italy, July 11–15, 2022), especially to Chenyi Hu, for valuable discussions.

References

1. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016.
2. M. Spurling, C. Hu, H. Zhan, and V. S. Sheng, "Anomaly detection in crowdsourced work with interval-valued labels", *Proceedings of the 19th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU'2022*, Milan, Italy, July 11–15, 2022, pp. 504–516.