Memories of the Future: Systems, Human, and Cybernetic Aspects of the Emerging Post-AI World*

Vladik Kreinovich[1], Miroslav Svitek[2], Julio C. Urenda[3], and Olga Kosheleva[4]

Abstract— **While current machine-learning-based AI techniques have been spectacularly successful, their present applications still leaves many important open questions – for example, how to make their results more reliable or, at least, how to gauge how reliable is each AI recommendation. In this paper, we argue that to fully answer these questions, we need to go beyond the current AI techniques, and that in this development, systems-, human-, and cybernetics-based ideas not only naturally appear, they seem to provide a way to the desired answers.**

I. WE ARE IN THE AGE OF AI, BUT . . .

We are in the age of AI. Spectacular successes of machine learning techniques have made everyone in awe of AI:

- Starting with ChatGPT, Large Language Models (LLMs) help us in many creative tasks like writing poems and coming up with class syllabi.
- Computers are world champions in Go – probably the most difficult human-invented game.
- Self-driving cars are a common sight in many cities.
- AI-based system solve practically important partial differential equations faster than all known algorithms, etc.

To many people, AI is all we need – and if the current AI is not yet perfect, all we need is to train it some more. Computer Science department and programs all over the world are renamed into AI departments and programs, most faculty openings in Computer Science are now AI-related, all other research directions are almost frozen – to the extent that several member of the Program Committee of one of the major fuzzy conferences seriously proposed to give the conference Best Paper Award to a machine learning paper that has practically nothing to do with fuzzy.

But. The recent success of AI was very fast and very unexpected – in terms of successes, it is a clear example of a steep exponential growth. It is natural to expect that this spectacular exponential growth will continue for some time. Of course, it is not possible that this exponential growth will last forever: there are many real-life examples where this growth stopped:

- an economic boom leads to overheated economy and even to a crisis,
- exponential population growth that worried people in the 1960s has drastically slowed down, etc.

But with AI, many people expect the boom to continue for some time.

However, our opinion is that the boom is about – if not to end, but at least to slow down. Indeed, while LLMs are well known to be not perfect, the hope was that, just like training on more examples made machine learning results better, feeding even more examples to LLMs will make them less imperfect.

What we see now is that the current LLMs have already devoured practically everything that is there – to the extent that several authors of copyrighted texts are suing the LLM-producing companies for a supposedly illegal use of their works.

In spite of all the training – that took practically all possible available examples:

- self-driving cars are still, in many situations, several times less safe than an average human driver; see, e.g., [1], and
- even the Go championship is now in doubt – while the AI indeed did easily beat Go professionals, it has been shown to be defenseless against a skilled amateur; see, e.g., [17].

Maybe we can get better results if instead of the current, mostly empirically determined aspects of neural network such as activation function etc., we find a better one? Unfortunately, there is little hope that this will leads to significantly better results: most such empirically determined aspects turned to be provably optimal in some reasonable sense [10] and references therein.

In short, what we observe is a drastic slowdown of AI progress.

Comment.

- We are *not* saying that AI progress has stopped: there are still many potential useful applications of modern AI, of current LLMs, applications that need to be developed and exploited.
- What we *are* saying is the techniques themselves are not growing as fast as we expected – to continue growing,

*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), HRD-1834620 and HRD-2034030 (CAHSI Includes), EAR-2225395 (Center for Collective Impact in Earthquake Science C-CIES), by the AT&T Fellowship in Information Technology, by a grant from the Hungarian National Research, Development and Innovation Office (NRDI), by the Institute for Risk and Reliability, Leibniz Universitaet Hannover, Germany, and by the European Union under the project ROBOPROX (reg. no. CZ.02.01.01/00/22 008/0004590)

[1]Vladik Kreinovich is with the Department of Computer Science, Unuiversity of Texas at El Paso, El Paso, TX 79968, USA, vladik@utep.edu

[2]Miroslav Svitek is with the Faculty of Transportation Sciences, Czech Technical University in Prague, Konviktska 20, 110 00 Praha 1, Czech Republic, miroslav.svitek@cvut.cz

[3]Julio C. Urenda is with the Department of Mathematical Sciences, Unuiversity of Texas at El Paso, El Paso, TX 79968, USA, jcurenda@utep.edu

[4]Olga Kosheleva is with the Department of Teacher Education, Unuiversity of Texas at El Paso, El Paso, TX 79968, USA, olgak@utep.edu

they need a boost.

What we do in this paper. In this paper, we analyze where the desired boost can come from. Our conclusion is that it will come if we better take into account systems, human, and cybernetic aspects of the problems – i.e., the aspects that are largely within the scope of our IEEE Systems, Man, and Cybernetics society.

We titled this paper "Memories of the Future" because:

- while we always appreciate new ideas,
- we believe that (and we argue that) many problems can be solved if we go back to the previous AI (and other) ideas, ideas that currently mostly frozen, and work to combine them with the current machine-learning-based AI techniques.

Most of our related ideas are raw, but some of them have started to becomes somewhat more technical. This paper is largely a position paper, so in this paper, we do not provide the corresponding technical details; instead, we provide references to more technical papers where these details are described.

II. SO WHAT ARE SYSTEMS, HUMAN, AND CYBERNETIC ASPECTS?

Let us start by describing what we mean by systems, human, and cybernetic aspects. *Systems* means that we consider processes as a whole, including software, hardware, humans, and other objects involved in the corresponding process. For example, a systems approach to manufacturing means to consider all aspects from providing raw materials to human satisfaction of the final products to the effects on environment and health.

Human means taking into account that the ultimate goal of all the human activity is to benefit the humankind. So we need to take into account not only objective characteristics, it is also important to take into account what humans want, and how they perceive what we offer them.

A classical example of the difference between objective characteristics and human perception is economy. Objectively, we should aim at the largest and more fairly distributed Gross Domestic Product (GDP). However, the ultimate goal is human happiness, and it is also well-known that while, in general, GDP is correlated with happiness, there are many exceptions. For example:

- in Mexico, people are, on average, happier that what a correlation model predicts, while
- in several countries of Northern Europe, they are less happy than they should be based on their GDP.

Finally, by definition (see, e.g., the original book [18]), *cybernetics* is a study of processes that are common to human, living beings, and machines. Studying such processes has led to many useful developments, and this usefulness makes sense. For example, if we want to make a machine that flies, it is useful to look at creatures that fly – birds and insects – do it. As a result, we have modern airplanes with wings. Of course, these wings are different from the birds' ones – for example, they do not flap, but still the main idea is largely the same.

Similarly, if we want to make a machine that thinks and makes decisions, a natural idea to look at how we humans think and make decisions.

- This can be on the level of statements and logics, as in the historical logical approach to AI.
- This can also be on the level of neurons – and this is, by the way, where the modern neural network-based spectacular AI models come from.

III. WHAT DO WE WANT?

What do we want? For all our problems – be it transportation, medicine, manufacturing – we want a solution:

- that would be satisfactory to us,
- that would provide accurate and reliable results while maintaining privacy and security,

and we want these solutions to be generated as fast as needed (which, in many cases, means as fast as possible).

And ideally, the corresponding computations should not spend as much energy as they do now, when computers take more than 5% of the world's energy consumption – and this proportion is growing. After all, our brains perform their computations and their reasoning at the energy level several orders of magnitude lower than the computers.

What we do in the following sections of this paper. We will go over all these requirements one by one. For each of them, we will explain:

- why we think that the current machine learning techniques are not sufficient, and
- how systems-, human-, and cybernetics-based ideas can help.

IV. HOW DO WE MAKE SURE THAT THE SOLUTION IS SATISFACTORY?

What do we want? First, we want to find out what we humans want, be it transportation, be it eating out, etc.

This is not easy to formulate in precise computer-understandable terms. For example, for transportation, in the first approximation, it makes sense to assume that we want to get from point A to point B as fast as possible. But this is true only in the first approximation: we also need to take comfort into consideration. It is largely because of the comfort that in many big cities, many people prefer to drive their own cars, while public transportation is often much faster.

How can we learn what people want? How can we take into account this *human* aspect of the corresponding problem?

Why cannot we just use current machine-learning-based AI techniques? At first glance, why not use machine learning (ML)? In many other learning tasks, they have been very successful.

But these ML techniques are only very successful when they are fed thousands and millions of examples. The need for so many examples can be naturally explained by simple statistics. Indeed, based on N noisy samples, we can find the value of the corresponding parameters with accuracy proportional to $1/\sqrt{N}$; see, e.g., [15]. So, to get accuracy of 0.1%, we need to have $N = 10^6$ – i.e., we need a million of examples. This, by the way, is the reason why learned self-driving cars are not yet perfect: even with billions of examples $N \approx 10^9$, they have accuracy of $1/\sqrt{N} \approx 3 \cdot 10^{-5}$, which is still, in many situations, much worse than the performance of a human driver [1].

Back to human aspects: it is easy to come up with millions and billions examples of values of the corresponding physical process – sensors are very affordable now, we can make many measurements – but it is difficult to come up with millions of survey results describing how people feel.

What can we do with this human-related aspect? Good news is that in coming up with ideas, we do not need to start from scratch. People have been thinking of how to describe human reasoning for quite some time (see, e.g., [13]) – this was one of the main direction in traditional AI, with neural networks being a poor cousin. It may be time to revive that research and combine machine learning with more traditional reasoning-based AI techniques.

Also, people have been thinking of how to formalize imprecise ("fuzzy") human sentiments for quite some time. One of the successful techniques – that led to many successful applications in the 1980s and continues to be useful – is fuzzy techniques; see, e.g., [5], [6], [11], [12], [14], [21]. So a reasonable idea is to also combine machine learning techniques with fuzzy (see, e.g. [8]). Such attempts are already going on, as judging by many presentations on fuzzy conferences and publications in fuzzy journals. We just need to take this activity more seriously:

- not, as some people in ML community perceive it – as attempts of somewhat obscure fuzzy community to compete,
- but as an attempt to further boost AI techniques.

Comments.

- Why should we combine ML with anything else? Many researchers like the saying "if all you have a hammer, everything starts looking like a nail". Hammer is just one of the tools, a very important one, no doubt: every household has a hammer, even those that do now own any other tools. Similarly, ML is just one of the tools, maybe one of the most important ones – but we often also need other tools to succeed.
- There is an additional advantage of combining machine learning with more traditional AI techniques and fuzzy techniques:
  - in contrast to usual machine learning – which, to a human user, is just a black box,
  - traditional-AI and fuzzy techniques use language.

  Human users can understand how these techniques come to their conclusions – and this understanding makes these technique more reliable to human users.
- For those who are not very familiar with modern fuzzy techniques, it is necessary to emphasize that they have gone way beyond the original Zadeh's simple ideas of using values from the interval $[0, 1]$ and min and max instead of "and" and "'or". Modern fuzzy techniques are much more complex now. And while they do not have as many spectacular applications as in the peak of their usage, in the 1980s and 1990s, they are still successfully used:
  - e.g., in many cars' automatic transmissions – where one of the main objectives is an imprecise objective to make the ride comfortable for the driver and for the passengers – and
  - in many other similar human-related applications – like a rice cooker – where the quality is largely subjective.

V. HOW DO WE MAKE THE SOLUTIONS MORE ACCURATE AND RELIABLE – AND AT LEAST HOW TO GAUGE THEIR ACCURACY AND RELIABILITY?

What is a problem? Many machine learning results are way off; they are usually called *hallucinations*. For example, a very effective image understanding algorithm may suddenly interpret a cat picture (a clearly cat picture) as a dog or even as a train.

Of course, there are cases when an algorithm makes a mistake: no one is perfect, and most algorithms are not perfect either. What many algorithms – and many humans – do in such situation is provide not only the answer itself, but also an estimate for the accuracy and/or reliability of this answer.

For example, when my doctor says that I have hernia that needs to be operated upon, all I can do is agree to this operation. However, if my doctor says that most probably it is hernia, but an additional ultrasound test may be helpful, then I would probably first undergo this test.

From this viewpoint, it is desirable to supplement machine learning results with such estimates. In other words, we need to apply methods of Uncertainty Quantification (UQ) to such situations.

And this is where further work is clearly needed: many machine learning algorithms already produce estimates of their answer's reliability and accuracy, but these estimates are way off. In examples when a system recognizes a cat as a train, its estimated confidence in this identification is sometimes 99%.

What can we do about it?

Why cannot we just use current machine-learning-based AI techniques? Since ML is so good in learning, why cannot we use it to provide the desired estimates? At first glance, we can rather easily do it. We can come up with groups of images with different degree of clarity. For each of these groups, we can apply the trained ML tool (whose accuracy we are estimating) to each of these images. Based on the results, we can compute the percentage of correct answers. Then, we train another ML tool on the pairs (image, percentage-of-the corresponding group). After this training, we expect that for every new image, thus trained tool will provide a good estimate for the first tool's reliability – and we can use similar techniques to gauge the accuracy of neural network's computations.

This seems like a natural idea, so does not this idea work? Why, when we train the neural network on pairs like (image, cat), (image, train), etc., it learns, but when try to train it on pairs (image, percentage), it does not? The answer goes back to the same statistical arguments that we used in the previous section.

To be well trained, we need many examples. If we have a sufficiently large number N of examples, the system will be well trained. However, to get an example of a probability, we need a group of such examples – and if we want to estimate this probability with accuracy 10%, we need – by the same statistical argument – to have at least 100 examples in each group.

Since each group has 100 examples, we have $N/100$ groups – which means that the resulting accuracy of estimating the degree of confidence is 10 times smaller than the accuracy of classification itself.

So what can we do? One of the main reasons why detection is not perfect is that we do not have a pure picture of a standard average cat – in this ideal case, we would have a perfect (or at least an almost perfect) recognition. In reality, we have a picture of an individual cat that is different from the average cat, and we have other objects in the picture. From this viewpoint, both factors – the difference between an individual cat and an average cat and the background – act as noise. So, the question become: how to gauge the effect of this notise on the system's decision?

Out of three aspects that we mentioned in Section II, here the most appropriate is the systems aspects. Indeed, after all, the trained neural network can be viewed as a system, and for systems, noise is a usual feature, we know how to gauge the effect of noise. It is therefore desirable to use the general system techniques to gauge the effect of this specific noise on the result. Traditional methods of this estimation – methods of sensitivity analysis – require that we estimate the effect of each noise components – i.e., in effect, that we estimate the partial derivation of the computation result with respect to each input.

In general, this requires calling the algorithm as many times as there are inputs – which, for ML models with many inputs, would be prohibitively long. Good news is that since training of a neural network is based on computing partial derivatives, we can use this built-in feature to compute the derivatives practically for free, at the expense of just one back-propagation step; see, e.g., [7]. Hopefully, other systems-related techniques can also be designed.

VI. HOW CAN WE MANAGE SECURITY AND PRIVACY?

Security and privacy are very important. However, every security and privacy professional will tell you that there is no such thing as perfect security and perfect privacy. What we really need is to make sure that the probabilities of security and privacy violations should be small. For this purpose, we need to be able to estimate this probability.

Similarly to the uncertainty quantification case, the traditional machine learning tools may not be very helpful here, since to estimate a probability, we need a group – and this drastically decreases the number of training examples and thus, drastically decreases the estimation accuracy. So, for these estimations, we need to supplement machine learning with more traditional security and privacy techniques.

For example, one way to preserve privacy – that is often used in surveys – is to replace exact numbers with ranges. For example, a survey mask whether the age is between 20 or 30, or between 30 and 40 years old, etc. – instead of asking for the exact age. Of course, if we only have, as inputs, ranges (and not the exact values), the computation results lose some accuracy. In this case, an important question is how to select ranges for different inputs so that, within the constraint of preserving privacy, the result will be as accurate as possible. It is known that once we know all the partial derivatives, we can then reasonably easily solve this problem; see, e.g., [20]. So, we can use the above-mentioned uncertainty quantification solution for machine learning-based data processing.

VII. HOW CAN WE MAKE COMPUTATIONS FASTER AND LESS ENERGY-CONSUMING?

What can we do? Training a neural network takes some time. For example, for the first version of ChatGPT, training took over a year. And we are talking computations on very fast high-performance computers. How can we make training faster? And how can we make it less energy-consuming?

Three factors cause the training of machine learning models to be rather long (and rather energy-consuming):

- first, we have a large amount of examples to process; each example includes many features, and inputting and processing all the features from all the examples requires significant time and energy;
- second, the training algorithms that we use require many computational steps, which, in their turn, require a lot of time and a lot of energy; and
- finally, the hardware on which the algorithms run takes time and energy to perform each operation.

So, to speed up computations, it is desirable to see if we can decrease the effect of these three causes.

How can we decrease the number of features. A natural idea is to only use the features that effect the result the most. Detecting such features is a known problem in system analysis, and there are many statistical techniques to detect the most important features. One of these techniques – actively use in machine learning – is the use of the Shapley value, a concept that was originally designed to find the collaborators who were most important for the success of a project. However, the use of the Shapley value does not always find the truly most important features. There are two ways in which we can improve the situation:

- First, the original Shapley value implicitly assumes that we know the exact effect of each combination of factors. In practice, especially when the effect is measured in probabilities – as in machine learning – these probabilities can only be determined approximately. It is therefore desirable to use modifications of the Shapley value that take this uncertainty into account; see, e.g., [2].
- Second, the use of Shapley values means, in effect, that we consider a *linear* approximation to the effectiveness, when the predicted value of using only m out of all n features is approximated by the sum of the values corresponding to each feature. Of course, most real-life systems are more complex than the linear ones. So, a natural idea is to use *non-linear* generalizations of the Shapley value concept; see, e.g., [3], [16], [19].

How can we decrease the number of needed examples? We humans do not need thousands and millions of examples to learn a new concert. For us, dozen (or so) examples is usually enough – and a few hundreds is definitely enough. How do we do it? It is often said that the best teacher is not the one who best teaches the class material, the best teacher is the one who teaches students how to learn. This is why we humans are so good (in comparison with the current neural networks) in learning new concepts and new ideas: because we have learned how to learn.

So, instead of training a neural network for each specific case, let us teach the neural network how to learn. Specifically, lrt us give, to a neural network, many examples, from different domains, of pairs in which:

- the input consists of several examples of the desired problem-solution pairs, plus a new problem of similar type, and
- the output is the solution to the last of the problems.

And:

- just like when we feed a neural network pairs (image, animal name), it learns to recognize an animal from an image,
- with this new training, we will feed it examples of a new concept, it will learn how to use this concept – just like we learn how to use it; see, e.g., [4].

For example:

- we give it many examples of addition triples, like $2+2=4$, $3+5=8$, etc., and a new similar example, e.g., $3+6$, and
- the trained neural network will (hopefully) return 9.

This idea comes from emulating how we humans do it – so it related to the cybernetic aspects.

Comment. This may also be related to the fact that many of us boost their productivity when attending face-to-face conferences: during several days, we hear a lot of pairs (problem, solution) – and this boosts our own ability to find solutions to complex problems. This is just like in elementary school: the process of going through many examples of addition that the teacher showed on the board helped us to better perform addition (or multiplication, or whatever concept it was).

What can we do on the hardware level? In modern computers, the only think that is moving is electrons. They move for two reasons:

- when we need to communicate a bit (or a sequence of bit) from one location to another, and
- when we need to change the state of the basic cell from 0 to 1 (or from 1 to 0).

The time needed for each such movement is equal to the distance divided by speed. Electrons already move with a speed close to the speed of light, so the only way to decrease time is to decrease the distance. Similarly, the energy needed for each movement is proportional to the distance. So, the only way to decrease the distance is also to decrease the distance.

Already now, within a cell, the distance that electrons need to travel when the state changes from 0 to 1 is the size of a few thousand molecules. If we decrease this distance even more, we will eventually get the size of a few atoms. Moving electrons between atoms is what is happening during chemical reactions – so we arrive at the idea of *chemical computing*. Chemical processes is the main way processes are happening in our brain, hence this falls under *cybernetic* aspects. So, it is desirable to pursue chemical computing.

In this pursuit, it is desirable to take into account that the larger the concentrations, the faster the reaction – and even the simulation of high-concentration chemical reactions can indeed speed up computations, in particular, neural computations; see, e.g., [9].

If we go even further in decreasing size, we get to the micro-objects, for describing whose behavior Newtonian physics is not longer a good approximation – we need to use quantum physics. Computing on this level will thus be, in effect, *quantum computing.*

VIII. LET US SUMMARIZE

Let us summarize the above analysis of how systems-, human-, and cybernetics-related ideas and techniques can help modern AI to become even more effective (and less flawed):

- We need to combine traditional AI techniques and fuzzy techniques with machine learning; this is related to human aspects.
- We need to further develop uncertainty quantification, security, and privacy techniques to machine learning models; this is related to systems aspects.
- To more adequately find the most important input features, we need to further explore the use of non-linear and/or uncertainty-affected generalizations of the Shapley value; this is also related to systems aspects.
- We need to use the ability of a neural network to learn not only to make it learn a specific material, but also to learn how to learn by itself; this is related to cybernetic aspects.

- We need to pursue the possibility of using chemical (and quantum) computing; this is also related to cybernetic aspects.

Let us do it. This is a position paper, not a paper describing results. We cannot guarantee that all these directions would lead to successes – and, in many of these recommendations, we do not have a clear idea of what exactly to do – but let us try! As a Russian poet Mayakovsky said (it sounds better when rhymed): *The future will not come by itself, we need to do something about it!*

REFERENCES

[1] M. Abdel-Aty and S. Ding, "A matched case-control analysis of autonomous vs human-driven vehicle accidents", *Nature Communications*, 2024, Vol. 15, Paper 4931.

[2] K. Autchariyapanikul, O. Kosheleva, and V. Kreinovich, "Shapley value under interval uncertainty and partial information", In: V. Kreinovich, W. Yamaka, and S. Leurcharusmee (eds.), *Data Science for Econometrics and Related Topics*, Springer, Cham, Switzerland, to appear.

[3] L. Bokati, O. Kosheleva, V. Kreinovich, and N. N. Thach, "Why Shapley value and its variants are useful in machine learning (and in other applications)", In: V. Kreinovich, S. Sriboonchitta, and W. Yamaka (eds.), *Machine Learning for Econometrics and Related Topics*, Springer, Cham, Switzerland, 2024, pp. 169–174.

[4] C. Baral and V. Kreinovich, "How to make a neural network learn from a small number of examples – and learn fast: an idea"; short version in *Abstracts of the 2023 IEEE International Conference on Systems, Man, and Cybernetics SMC 2023*, Honolulu, Hawaii, October 1–4, 2023; detailed version in: *Proceedings of the 9th World Conference on Soft Computing*, Baku, Azerbaijan, September 24–27, 2024.

[5] R. Belohlavek, J. W. Dauben, and G. J. Klir, *Fuzzy Logic and Mathematics: A Historical Perspective*, Oxford University Press, New York, 2017.

[6] G. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, Upper Saddle River, New Jersey, 1995.

[7] O. Kosheleva and V. Kreinovich, "How to propagate uncertainty via AI algorithms", *Proceedings of the 10th International Workshop on Reliable Engineering Computing REC'2024*, Beijing, China, October 26–27, 2024, pp. 5–12.

[8] O. Kosheleva and V. Kreinovich, "All we (and LLMs) need is fuzzy: an argument", *Proceedings of the IEEE International Conference on Fuzzy Systems FUZZ-IEEE 2025*, Reims, France, July 6–9, 2025, to appear.

[9] V. Kreinovich and O. Fuentes, "High-concentration chemical computing techniques for solving hard-to-solve problems, and their relation to numerical optimization, neural computing, reasoning under uncertainty, and freedom of choice", In: E. Katz (ed.), *Molecular and Supramolecular Information Processing: From Molecular Switches to Logical Systems*, Wiley-VCH, Wienheim, Germany, 2012, pp. 209–235.

[10] V. Kreinovich and O. Kosheleva, "Optimization under uncertainty explains empirical success of deep learning heuristics", In: P. Pardalos, V. Rasskazova, and M. N. Vrahatis (eds.), *Black Box Optimization, Machine Learning and No-Free Lunch Theorems*, Springer, Cham, Switzerland, 2021, pp. 195-220.

[11] J. M. Mendel, *Explainable Uncertain Rule-Based Fuzzy Systems*, Springer, Cham, Switzerland, 2024.

[12] H. T. Nguyen, C. L. Walker, and E. A. Walker, *A First Course in Fuzzy Logic*, Chapman and Hall/CRC, Boca Raton, Florida, 2019.

[13] P. Norvig and S. Russell, *Artificial Intelligence: A Modern Approach*, Pearson, New York, 2021.

[14] V. Novák, I. Perfilieva, and J. Močkoř, *Mathematical Principles of Fuzzy Logic*, Kluwer, Boston, Dordrecht, 1999.

[15] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.

[16] M. Svitek, N. Winnewisser, M. Beer, O. Kosheleva, and V. Kreinovich, "Why Shapley value and its generalizations are effective in economics and finance, machine learning, and systems engineering", In: H. T. T. Nguyen, H. H. Phan, and V. N. Huynh (eds.), *Data Science in Finance and Accounting*, Springer, Cham, Switzerland, to appear.

[17] R. Waters, "Man beats machine at Go in human victory over AI: Amateur exploited weakness in systems that have otherwise dominated grandmasters", *Financial Times*, February 19, 2023, https://arstechnica.com/information-technology/2023/02/man-beatsmachine-at-go-in-human-victory-over-ai/

[18] N. Wiener, *Cybernetics: Or Control and Communication in the Animal and the Machine*, MIT Press, Cambridge, Massachisetts, 1948.

[19] N. Winnewisser, M. Beer, O. Kosheleva, and V. Kreinovich, In: V.-N. Huynh, K. Honda, B. H. Le, M. Inuiguchi, and H. T. Huynh (eds.), "How Shapley value and its generalizations can help in the analysis of complex engineering systems and what next", *Integrated Uncertainty in Knowledge Modelling and Decision Making. Proceedings of the 11th International Symposium, IUKM 2025, Ho Chi Minh City, Vietnam, March 17–19, 2025*, Part II, Springer Lecture Notes in Computer Science, 2025, Vol. 15586, pp. 85–97, https://doi.org/10.1007/978-981-96-4603-6_8

[20] G. Xiang and V. Kreinovich, "Data anonymization that leads to the most accurate estimates of statistical characteristics", *Proceedings of the IEEE Symposium on Computational Intelligence for Engineering Solutions CIES'2013*, Singapore, April 16–19, 2013, pp. 163–170.

[21] L. A. Zadeh, "Fuzzy sets", *Information and Control*, 1965, Vol. 8, pp. 338–353.