Generalized Information

A Straightforward Method for Judging Machine Learning Models

Jonathan Bartlett and Eric Holloway

DOI: 10.33014/issn.2640-5652.1.2.bartlett.1

Abstract

Generalized Information (GI) is a measurement of the degree to which a program can be said to generalize a dataset. It is calculated by creating a program to model the data set, measuring the Active Information in the model, and subtracting out the size of the model. Active Information allows GI to be usable with both exact and inexact models.

1 Introduction

In Machine Learning and other forms of statistical inference, the goal is to create a model that matches the data. A model is essentially a function which takes a certain number of inputs and generates an output value. The input is whatever parameters the statistical model is allowing, and the output is the prediction, classification, or whatever the model is meant to identify.

As a simplified example, let's say that you are a realtor and you want to know the impact of square footage and the year a home was built on its selling price. Given a large amount of data, a machine learning package might find a way to model that data, so that, if you give it an input which is not in the set, the system will spit out for you what it thinks the selling price will be.

2 The Problem of Overfitting

Exact fits of models to data are not necessarily preferable in machine learning. Such models are often said to be *overfit*. Overfitting occurs because not all data is actually signal.

Nearly any dataset will contain some amount of noise. If your model makes an exact fit to the data, that means that much of your model is actually *modeling the noise*. Modeling the noise actually causes poor performance as the model is extended out to new data points. For the purposes of this paper, noise can either be statistically random events (variations around a mean) or even non-noise features, but whose predictive inputs are not included in the set of inputs being modeled.

In the realtor example above, let's say that most homes in the 1,000-1,200 square feet area that were built in 1975 were selling for $150,000, but one house, which was 1,175 square feet, sold for $75,000, because the homeowner desperately needed to sell it quickly. If the model attempted to have an exact model, that particular data point would cause bad predictions for square footages for that year (and possibly surrounding years). Thus, overfitting a model means that both data and noise are included in the model.

The goal, therefore, is to find a way to tell if a given model matches the data in the correct way. To do this, we will explore the question of models from a philosophical standpoint, and use those results to come up with a mathematical definition of a good model.

3 What is a Model?

What is the goal of modeling?

For most people, the goal of making a model is to enable *prediction* of points that we don't have. For instance, in the real estate model example, the goal is to be able to determine, as best we can, what the unknown price points will look like. We already know what the existing points are. If all we wanted to do was know what different square footages in different years sold for in the past, we don't need a model, we only need a lookup table.

Now, obviously, machines can't predict the future. We do not expect our predictions to work if the very basis of what is happening in our dataset changes. For instance, we would not expect a model to continue to function if a community implemented price controls for homes. Therefore, a model has an inherent presumption that future data will have the same essential patterns as current data.

4 Picking Models

There are innumerable ways to pick models. Given any discrete dataset, there are literally infinite models that can be made to match them. Therefore, given an infinite selection of choices, how does one decide which model is the best model for a given set of data?

As an example, Figure 1 shows a set of points. These points can be given by the following list of data pairs:

$$(42, 21), (40, 20), (30, 15), (24, 12), (36, 18), (14, 7), (12, 6) \quad (1)$$

Now, as mentioned, if we want to generate a model for these points, there are actually an infinite number of models to choose from. One model (shown in Figure 2) can be given by the equation

$$\begin{aligned} x^7 - 198x^6 + 16364x^5 - 729288x^4 + 18855360x^3 - \\ 281625984x^2 + 2241146880x - y^7 + 99y^6 - 4091y^5 + \\ 91161y^4 - 1178460y^3 + 8800812y^2 - 35017920y - \\ 7258507200 = 0. \end{aligned} \quad (2)$$

Another model (shown in Figure 3) can be given by the equation,

$$y = \frac{x}{2}. \quad (3)$$

Yet another model (shown in Figure 4) can be given by the equation

$$\begin{aligned} x^7 - 198x^6 + 16364x^5 - 729288x^4 + 18855360x^3 - \\ 281625984x^2 + 2241146880x - y^9 + 100y^8 - 4190y^7 + \\ 95252y^6 - 1269621y^5 + 9979272y^4 - 43818732y^3 + \\ 92171520y^2 - 57153600y - 7315660800 = 0. \end{aligned} \quad (4)$$

Keep in mind that all of these models are *perfect* fits to the original dataset, and there are infinitely more models available to choose from. While it may seem intuitively obvious which of these models should be chosen, more difficult cases make intuition more problematic. Therefore, we have to establish specific criteria for choosing models.

The one we intuitively think of as the best fit is (3) (Figure 3). Notice that this is also the *shortest* description of the data.

One consistent theme of the theory of inductive inference through the ages is that, when deciding between two equally-explanatory theories, the shortest one is the best. This has been expressed by Aristotle ("the more limited, if adequate, is always preferable"), Ptolemy ("we consider it a good principle to explain the phenomena by the simplest hypothesis possible"), Occam ("plurality must never

Figure 1: Example Set of Data Points

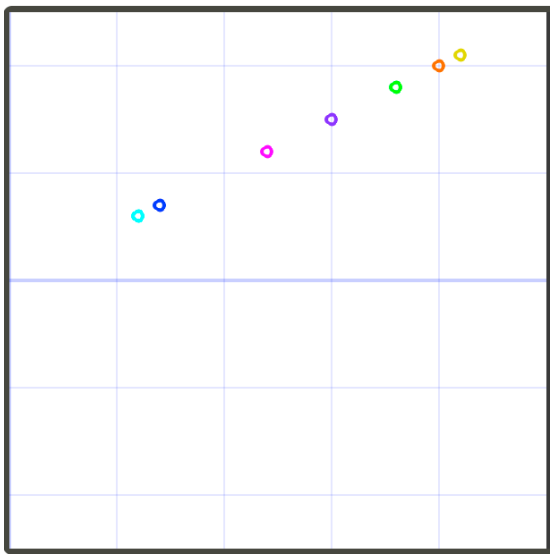


Figure 2: A Basic Curve Model

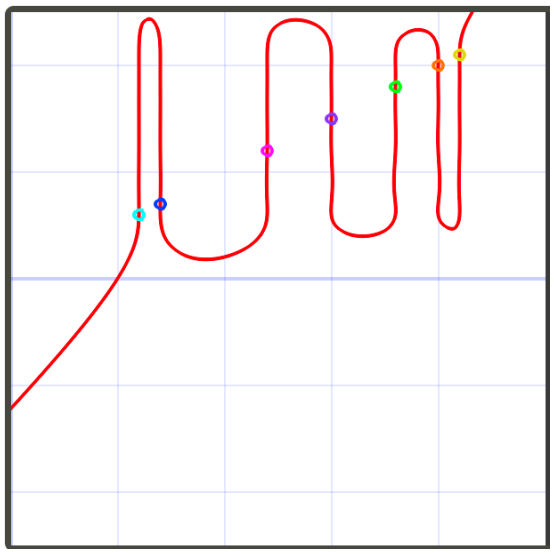


Figure 3: A Line Model

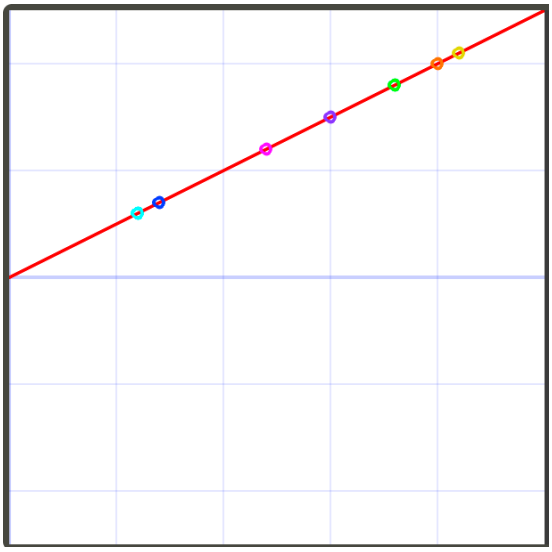
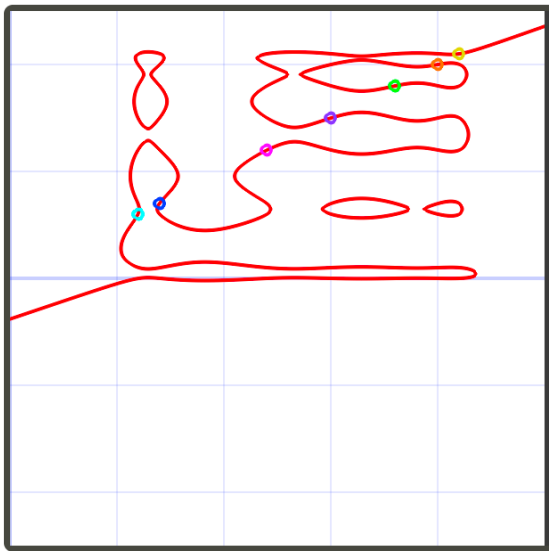


Figure 4: A Complex Curve Model



be posited without necessity"), and Newton ("we are to admit no more causes of natural things than such as are both true and sufficient to explain their appearances"). This principle is generally known by the name "Occam's Razor" (Borowski, 2012).

The underlying strand in all of these is to never make something more complicated than necessary. While it may be difficult to judge between two models that explain different subsets of data (though we will tackle this question in Section 7), if two theories are exactly equivalent in explaining known data, this principle definitely prefers the one that is simpler. Keas (2018) provides a detailed account of how this criteria has worked in the history and philosophy of science.

Since computer models are encoded in bits, we actually have an objective way of measuring the size (and, correspondingly, the simplicity) of the model. Therefore, if two models are equivalent in prediction, the one which can be encoded in the fewest bits is to be preferred. This ability to connect model size to Occam's razor was first identified by Solomonoff (Solomonoff, 1964a; Solomonoff, 1964b). Solomonoff induction has been used as the basis for a number of machine learning induction techniques, including PAC learning, Occam Learning, and others (a comparison of the present approach with these is given in Section 12).

5 Finding Generalizations

The methods described in Section 4 allows us to distinguish between two possible models. However, while it does allow us to determine if one model is better than another, it does not give any information about whether or not a model is a "good" model in an absolute sense.

Look again at Figures 2 and 4. Even if we didn't have the model in Figure 3, both of these seem ridiculously overcomplex for the given data. That is, neither of them is a good model for the given data, even if they are being compared against each other.

To understand why this is the case, take a look at the data points given in (1). Now, take a look at (2) and (4). In both of these cases, the equations are *longer* than the original data set.[1] However, the equation that seems to be a better model is (3). In this case, at least from an initial look, the model is shorter than the data that is modeled.

[1]While we haven't specified a specific mechanism for measuring these size of equations, for these particular equations, pretty much every means available for measuring their size will be longer than the original data set.

We will call a model that is shorter than the data that it is modeling a *generalization*. Generalizations are important because only generalizations represent real learning. This can be understood just from thinking about the problem itself. For any given set of data points, the data points could be treated as a mapping of input to output. Therefore, the data points themselves act as a model for themselves. Since Occam's Razor measures models comparatively, we already have the data points themselves as a one possible model for the data. Therefore, any proposed model must be smaller than the data points themselves.

We will call a model that is smaller than the data points themselves a *generalization*.

This is the first iteration of the concept we will construct. Generalized Information (I_G) is the difference between the size of the data points and the size of the model. Therefore, in this first formulation, if D is the data and $|D|$ is the size of the data in bits, and M is the model and $|M|$ is the size of the model in bits, then

$$I_G = |D| - |M|. \tag{5}$$

This measures the amount of generalization that a model provides for its data.[2] In this formulation, when I_G is positive, M provides a generalization of D. When I_G is zero or negative, M does not provide a generalization of D.

6 Output Prediction Dimensionality

The one thing to consider about the model given in the previous section is that, since it uses data points, it is ambivalent as to which direction the prediction occurs in. That is, given n-dimensional data, one can use the data points to arrive at results from any $n-1$ dimensions given. So, if our data consisted of homes with the year that it was built, the square footage, and the price it sold at, we presumably would want to know the price based on the year it was built and the square footage. However, there is nothing preventing us from looking up the square footage based on the price and the year it was built, or looking up the year it was built from the price and the square footage.

However, many machine learning models are non-reversible. That is, if the goal is to determine the final price, and the model is trained to look for a final price, the model cannot be used to take a final price and square footage and solve for the year. In many machine learning systems, you would have to build *separate* models for each direction of the data.

Therefore, for an n-dimensional system, you would need n models of the training data in order to match the original success of the data points. This means that we need to modify (5) in order to account for this. For a simple version, you can simply divide $|D|$ by the number of dimensions, which would lead to

$$I_G = \frac{|D|}{n} - |M|. \tag{6}$$

However, more specifically, you can think of the dataset itself divided by dimensions, where each dimension has its own size specification. Therefore, in most machine learning systems, the model is only able to output a single dimension, which we can consider the "output" dimension. Therefore, we can be even more explicit about the model size based on this dimension, yielding

$$I_G = |D_{\text{out}}| - |M|. \tag{7}$$

7 Dealing with Fuzzier Models

Not every model is an exact fit for data. However, not every model needs to be an exact fit. If the goal is to avoid modeling noise, then some amount of discrepancy between model and data needs to be allowed for. The goal, then is to transform the ideas present in (5) and (7) so that they continue to apply to noisy data.

Active Information is a simple and straightforward way to measure the amount of information that a model models (Dembski and Marks II, 2009). First we will understand Active Information on its own, original terms, and then we will apply this to the study of Generalized Information.

7.1 Active Information Basics

Imagine that we are looking for a particular card in a standard deck—we'll use the King of Diamonds for this example. Active Information says that the *endogenous information* (I_Ω) is the probability that we will find the card by random guessing, expressed in bits.[3] The probability for finding the King of Diamonds in a deck in a single random guess is $\frac{1}{52}$, which is approximately 5.7 bits.

[2] There are some important caveats here, but the goal is to focus on the philosophical underpinnings rather than technical minutiae. In any case, for this to be sound, all data should be given in prefix-free formats, and $|D|$ will also have to include some constant number of bits to convert D from pure data into a model.

[3] If p is the probability, $-\log_2(p)$ is the probability expressed in bits.

Now, let's say that someone has outside knowledge of how this particular card deck is organized. They tell you (correctly) that the King of Diamonds is one of the first four cards. Now, we can create a directed search that picks one of the first four cards at random. This new search probability, termed *exogenous information* (I_S) is $\frac{1}{4}$, which is 2 bits.

The *Active Information* (I_+) is the amount of information that my search strategy applies to the problem at hand. Active information is given simply as

$$I_+ = I_\Omega - I_S \tag{8}$$

If Active Information is positive, then the search strategy is helping you, but if Active Information is negative, then the search strategy is hurting you.

7.2 Bitwise Active Information

Active Information is normally applied to complete results—that is, the chance of guessing a number correctly, or guessing a number within a boundary of error, etc. However, Active Information can also be applied in a bitwise manner by simply applying Active Information to the probability of guessing each bit correctly. Therefore, if we have a target bitstring of 01001100101, we can measure the amount of Active Information in a generated bitstring 01001100111. The endogenous information present in the first bitstring is simply the number of bits—11 bits (i.e., $-\log_2\left(\left(\frac{1}{2}\right)^{11}\right)$). The exogenous information present in the algorithm that generated the second bitstring can be found by first looking at the probability. The second bitstring hit the target $\frac{10}{11}$ times. Therefore, the exogenous information is $-\log_2\left(\left(\frac{10}{11}\right)^{11}\right) = 1.5$ bits. Therefore, the Active Information in the algorithm that generated the second bitstring is $11 - 1.5 = 9.5$ bits.

7.3 Applying Active Information to Fuzzy Models

If we are going to allow for models which have a fuzzy relationship to the actual output, we need a mechanism of also discounting the allowed model size. Active Information can be applied by reducing the measured size of our output dimension based on the Active Information in the result.

Active Information can be applied to models in the following way—it is the ability for a model to improve the guessing on data outcomes. Imagine that the outcomes of the known dataset were guessed at.[4] What is the probability of guessing the results at random? This represents I_Ω. Now, imagine that we use model M to improve our guessing. What is the new probability of guessing the result? This represents I_S. Therefore, the amount of data that is modeled by our model is the Active Information, I_+.

When using this for a generalization, we only care about the size of the *actual* modeled information, I_+. Therefore, we can use this idea to transform (7) into a more nuanced equation,

$$I_G = I_+ - |M| \tag{9}$$

We can see that, for the case where we have an exact model, (7) and (9) are equivalent.[5]

If our search allows for exact guessing, then there is no exogenous information in the search. In other words, $I_S = 0$. Therefore, all that is left is I_Ω, which, with only guessing, is the size in bits of the data. Therefore, I_Ω will be the size of the data itself (at least in the output dimension).

For reversible models of n dimensions, the Active Information from each dimension can be summed up for a total model size, yielding

$$I_G = \left(\sum_{x \in n} I_+(x)\right) - |M|. \tag{10}$$

8 A Simplified Example and Application

To see how this works, imagine a dataset of 400 entries. Each data point will be a simple boolean true/false bit, with the input being simply the index of the bit. For this dataset, random guessing will achieve a 50% probability of a correct answer for each bit, yielding an endogenous information of 400 bits for the whole dataset. Program Z_1 reproduces the 400 bits exactly, and $|Z_1|$ is 260 bits. Program Z_2 reproduces the 400 bits with 99% accuracy, and $|Z_2|$ is 190 bits. Both of these programs generalize the data ($|Z_1| < 400$ and $|Z_2| < 400$), but which one generalizes the data better?

[4] Note that we can use either regular Active Information or bitwise Active Information for this.

[5] To see this more explicitly, remember $I_+ = I_\Omega - I_S$. I_Ω will be the size of the result (the output dimension) in bits, which is simply $|D_{\text{out}}|$. If the result is exact, then $I_S = 0$. Thus, $I_+ = |D_{\text{out}}| - 0 = |D_{\text{out}}|$.

For Z_1, $I_G = 400 - 260 = 140$ bits. For Z_2, since Z_2 is not an exact match to the data, we need to calculate its Active Information. I_Ω will be the same. Since the model moves the probability for each bit to 99% accurate, the exogenous information (I_S) is $-\log_2\left(0.99^{400}\right) \approx 5.8$ bits. Therefore, the Active Information (I_+) in this model will be

$$I_+ = 400 - 5.8 = 394.2 \approx 394 \text{ bits.} \tag{11}$$

Therefore, since $|Z_2| = 190$, the Generalized Information will be

$$I_G = 394 - 190 = 204 \text{ bits.} \tag{12}$$

Because the I_G of Z_2 is greater than the I_G of Z_1, this means that Z_2 is a better generalization of the data, even though it is less accurate. Z_1 is at risk of slight overfitting, because the gain in accuracy is more than offset by the increase in the complexity of the model.

Thus, Generalized Information allows weighing between generalization and accuracy in models. It provides a philosophically coherent scoring system which weighs together accuracy and model size to determine which models are to be preferred over others, and which models should even count as generalizations at all.

9 Generalized Information and Knowledge

Let us write $I_G(M, D)$ as the amount of generalization a model M has about dataset D. In this framework, knowledge can be represented as the following limit:

$$\lim_{|D_{\text{out}}|\to\infty} I_G(M, D) = \infty \tag{13}$$

In other words, if a generalization can be applied to a theoretical infinite number of data points, it is knowledge. Note that this definition of knowledge does not require exactness, since Generalized Information does not require exactness. It merely requires that increasing the amount of data without bound also increases the amount of generalization of the model.

Under this system, classical physics, despite it not being an exact description of reality, is considered knowledge, because the same model continues to generalize more and more points as the dataset gets larger.

Additionally, we can be certain that if *knowledge* about a topic can be found, then, with a large enough sample size, it can be generalized through generalized information. That is, if we are able to form a model of an item of knowledge, then it is covered by a fixed size program. If knowledge is defined to be the continued applicability of a model to an infinite size of data, and the model is of a fixed size, then that means that we will have generalized information available in the limit. Therefore, there is some quantity of data for which a knowledge-oriented model provides generalization.

Interestingly, this also means that, given enough data, the most size-efficient model is not even necessary. That is, if the ideal model is M_i, but a poorly implemented model M_p has the same accuracy but is inefficiently coded (i.e., it is coded using three times amount of code), given sufficient data, M_p will also be a generalization.

In short, if knowledge can be had through a model, it will show generalized information on a sufficiently large dataset.

10 Why Generalization Works

Montañez (2017) points out that all machine learning systems work only because of the existence of a bias. He paints compression-based learning systems with skepticism because, in theory, for any particular arrangement of codes, the desired ordering of codes may put the longer ones first.

What makes the present model different is that it does not approach machine learning with the assumption that the hypothesis space is adequate. Instead, it presents a way of testing if a given hypothesis can be defensibly considered to have generalized the training data. For any particular mappings of codes to functions, the training data may not be sufficient to generalize. Thus, the ability to find an appropriate hypothesis under Generalized Information may not be available.

However, in Section 9, we pointed out that, if an appropriate model exists at all, there will be some size of training data for which the model will provide compression. Since Montañez focused on finite sets of data,[6] this would not be true for his system.

A way of understand the relationship between Generalized Information and the results of Montañez is to say that Generalized Information provides a way of knowing whether or not your dataset is sufficiently large to provide enough information to the hypothesis space to be confident that the hypothesis is doing its job. In other words, Generalized In-

[6] Section 3.1 of Montañez (2017) says, "We limit ourselves to finite, discrete search spaces, which entails little loss of generality when considering search spaces fully representable on physical computer hardware within a finite time."

formation is able to detect whether or not the bias in the hypothesis space has sufficient mutual information with the training set in order to have confidence that a given hypothesis likely also shares some amount of mutual information with the underlying data set. It does not say anything about whether or not a given hypothesis space is sufficiently biased in order to do this with a particular size of training data, or even if the underlying structure to the system can be modeled with a finite model in the hypothesis space.

Generalized Information does not say that a model that does not exhibit generalized information is incorrect. Rather, it says that there is not sufficient data to know, based on the data itself, that it is true. For instance, if we are given only a single data point, since it is presumably based on some real phenomena, there is some model about that data point that is true. If a good model is selected, then that model will continue to be true for future data points. However, it is impossible to tell *from that single point* whether or not the model matches the data sufficiently. A given modeller may know from *other information they know about the problem* whether or not the model is correct, but not from the data itself. Generalized Information tells you the cutoff point for when you can know, based on the data itself, when generalization is occurring.

11 Benefits of Generalized Information

Generalized Information offers several benefits over other inferential models. As noted in Section 4, Generalized Information is not the first inferential system to utilize Occam's Razor as a foundation stone. However, Generalized Information offers several benefits, including:

- It provides a minimum threshold for establishing whether a model is a valid generalization of the dataset.
- The techniques are grounded philosophically—each step is the result of philosophical analysis of the goals we are trying to achieve.
- The techniques are straightforward—only the most basic information theory mathematics are required to perform them.
- The techniques are adaptable—it is not dependent on any particular type of model being used.
- Generalized Information allows use of more data—generalizations can be made using the entire dataset.[7]

Additionally, while a thoroughly rigorous application of these ideas in software may require a good amount of programming effort,[8] a "good enough" approach is fairly straightforward to implement. The amount of Active Information can be determined statistically, and the data size and model size can both be evaluated simply by checking storage size inside the program itself.

12 Comparison with Other Machine Learning Systems

The first thing to note about Generalized Information is that it is not necessarily in competition with other machine learning systems. In fact, Generalized Information does not specify either (a) the nature of training algorithm, or even (b) the nature of the model being used. Generalized Information, as such, is fully compatible with any training algorithm or any model. Generalized Information is only a *test* of the outcomes of a machine learning system. For instance, for a Decision Tree, Generalized Information would take into account the number and size of nodes (i.e., the model size) and compare it to its accuracy on the training data. However, some machine learning systems do provide opportunities for comparison.

For instance, take the k-Nearest-Neighbor (kNN) algorithm. For this algorithm, the model *is* the training data. Therefore, according to Generalized Information, a naive implementation of kNN *cannot* exhibit Generalized Information. However, kNN could be tweaked to do so. For instance, one could establish a *subset* of training data that performs sufficiently well in order to be considered a generalization.

Many statistical tests have been established to compare two alternative models, and decide which one is more likely to be the true model, even based on model size. The Vuong closeness test, for instance, compares two models, taking into account the number of parameters in each model. Many other similar criteria are available, including the Akaike information criterion, the Bayesian information criterion, and others (Sayyareh, Obeidi, and Bar-Hen, 2011). In the Machine Learning field, the principle of Solomonoff induction inspired several systems including PAC-learning (Valiant, 1984) and Minimum Description Length (Grün-

[7] In typical machine learning techniques, some of the data has to be held back in order to test the model on the data. Since this technique focuses on generalization rather than prediction, all of the data can be used.

[8] Here, I am considering such problems as converting all data values to prefix-free formats, identifying a fixed language for a model, establishing the program size for converting a dataset into a model to determine $|\boldsymbol{D}|$, etc.

wald, 2005). These have very similar characteristics to the statistical criteria listed above, in that they base their model selection on some combination of model size and accuracy.

There are three primary advantages to Generalized Information against all of these other criteria.

1. These criteria only work for parameterized models, while Generalized Information can operate with any type of model, as long as it can be evaluated for a size (which means any computer-implementable model).
2. These criteria are only relative comparisons. That is, they compare models against each other, not against some absolute standard. Generalized Information establishes a minimum criteria which must be achieved—the size of the training data itself. Additionally, using Active Information, this size can be adjusted based on the degree of accuracy which is attained.
3. Generalized Information is a much more straightforward test. It is understandable by nearly anyone with the slightest background in information theory or computer programming, while the other tests require much more advanced knowledge of statistics. Additionally, knowing *why* Generalized Information works can be done without hardly any mathematics, as this present paper demonstrates.

Interestingly, since Generalized Information is model-agnostic, all of these criteria could be extended with at least part of Generalized Information by constraining models to a maximum model size based on the size of the training data available and the accuracy of the model.

13 Potential Problems

Several potential problems exist with this framework. The first and most obvious one is whether or not limiting the size of the model to the size of the output dimension as discussed in Section 6 is the correct procedure. While it appears to be correct from a variety of angles (i.e., reconstructing all output dimensions allows use of the entirety of the training data size), it does seem that the allowable model size should include some amount of size from the input dimensions, since, after all, it will be *using* those dimensions in calculating the output dimension.

Another one is calculating accuracy. That is, with a naive approach, all failures are equivalently failing. This is not an essential problem with the model, as it is based on the output encoding. A simple naive output encoding could be squished into a binary output, but this prevents one from knowing "how far" off the algorithm is. An alternative output encoding could put more weight on more significant figures, and less weight on less significant figures, such that results that are "near" can match more bits than those that are "far."

Additionally, this model needs to be subject to empirical verification. Currently, it exists only as an idea, and needs to be applied to specific problems to demonstrate that it can indeed prevent overfitting and demonstrate generalizations as claimed.

14 Future Applications of Generalized Information

Generalized Information can be used anytime someone wants to judge between actual fits between model and data and post-hoc curve fitting. The primary target considered here is for machine learning models, however, it can be applied to a number of similar situations.

For instance, this technique could be used as a replacement for p-values in statistical inference. p-values do not take into consideration the model size used to establish the inference. Thus, Generalized Information can prevent the problem of p-value hacking in many statistical applications. That is, imagine that someone pulls in a huge number of datasets in order to hack a false statistical correlation. With Generalized Information, there would need to be sufficient information within the model to choose which statistics are to be used for correlation, thus depressing the p-value of the result. Statistically, $p < 0.05$ should be roughly equivalent to $I_G > 4.322$.

Finally, work needs to be done on establishing a baseline language for this type of modeling. In the limit (see Section 9) it doesn't matter what language is used for the model, as long as the language selection is independent from problem selection. However, for practical purposes, it would be helpful to have a unified, efficient language that could be used for general comparisons.

Acknowledgements

The author would like to acknowledge Robert Marks, Winston Ewert, George Montañez, Andrew Jones, George

Hunter, William Dembski, and Mike Keas for feedback on early versions of this manuscript.

References

Borowski, S (2012). "The origin and popular use of Occam's razor". In: *Sciencia*. URL: https://www.aaas.org/origin-and-popular-use-occams-razor.

Dembski, W A and R J Marks II (2009). "Conservation of Information in Search: Measuring the Cost of Success". In: *IEEE Transactions on Systems, Man and Cybernetics A, Systems and Humans* 5.5, pp. 1051–1061.

Grünwald, P (2005). "Introducing the Minimum Description Length Principle". In: *Advances in Minimum Description Length: Theory and Applications*. Ed. by P Grünwald, J Myung, and M A Pitt. MIT Press, pp. 3–22. URL: https://arxiv.org/pdf/math/0406077.pdf.

Keas, M N (2018). "Systematizing the Theoretical Virtues". In: *Synthese* 195 (6). URL: https://link.springer.com/article/10.1007/s11229-017-1355-6.

Montañez, G D (2017). "Why Machine Learning Works". PhD thesis. Carnegie Mellon University.

Sayyareh, A, R Obeidi, and A Bar-Hen (2011). "Empirical Comparison between Some Model Selection Criteria". In: *Communications in Statistics—Simulation and Computation* 40, pp. 72–86.

Solomonoff, R (1964a). "A Formal Theory of Inductive Inference, Part 1". In: *Information and Control* 7.1, pp. 1–22. URL: http://raysolomonoff.com/publications/1964pt1.pdf.

Solomonoff, R (1964b). "A Formal Theory of Inductive Inference, Part 2". In: *Information and Control* 7.2, pp. 224–254. URL: http://raysolomonoff.com/publications/1964pt2.pdf.

Valiant, L G (1984). "A Theory of the Learnable". In: *Communications of the ACM* 27.11, pp. 1134–1142. URL: http://web.mit.edu/6.435/www/Valiant84.pdf.