A quick and easy way to estimate entropy and mutual information for neuroscience

Mickael Zbili, Sylvain Rama

https://www.biorxiv.org/content/10.1101/2020.08.04.236174v2

Submitted by Sylvain Rama 2020-08-06 13:36
Parallel submission

Abstract

Calculations of entropy of a signal or mutual information between two variables are valuable analytical tools in the field of neuroscience. They can be applied to all types of data, capture nonlinear interactions and are model independent. Yet the limited size and number of recordings one can collect in a series of experiments makes their calculation highly prone to sampling bias. Mathematical methods to overcome this so-called “sampling disaster” exist, but require significant expertise, great time and computational costs. As such, there is a need for a simple, unbiased and computationally efficient tool for estimating the level of entropy and mutual information. In this paper, we propose that application of entropy-encoding compression algorithms widely used in text and image compression fulfill these requirements. By simply saving the signal in PNG picture format and measuring the size of the file on the hard drive, we can estimate entropy changes through different conditions. Furthermore, with some simple modifications of the PNG file, we can also estimate the evolution of mutual information between a stimulus and the observed responses through different conditions. We first demonstrate the applicability of this method using white noise signals. Then, while this method can be used in all kind of experimental conditions, we provide examples of its application in patch-clamp recordings, detection of place cells and histological data. Although this method does not give an absolute value of entropy or mutual information, it is mathematically correct, and its simplicity and broad use make it a powerful tool for their estimation through experiments.

Keywords: Entropy, Mutual Information, Electrophysioogy, Histology

Round #1

---

by Haudur Freyja Olafsdottir, Mahesh Karnani and Fleur Zeldenrust, 2020-11-05 10:46
Manuscript: https://www.biorxiv.org/content/10.1101/2020.08.04.236174v1

Request for revision: 'A quick and easy way to estimate entropy and mutual information for neuroscience'

Dear Sylvain and Mickael,

Thanks for submitting your preprint to PCI C Neuro, and apologies again for the delays - it took us some time to reach outside our usual neuroscientist pool in order to get feedback on the mathematics as well. The reviews of your manuscript "A quick and easy way to estimate entropy and mutual information for neuroscience" are below. The manuscript was evaluated by 2 reviewers and 2 of us (reviewers 3 and 4). Everyone was enthusiastic about the work, but felt there were some crucial edits to be made. Furthermore, the reviewers would like you to make explicit under what experimental conditions the method can be used, which could enhance the use of the presented method. If you could write a response to the reviews and upload a revised preprint, we would be happy to evaluate a revision for recommendation, this time with a much faster turnaround time. We hope this feedback helps in the preparation of a revised manuscript. Thank you for participating in the PCI Circuit Neuroscience initiative. Best regards, Freyja, Mahesh and Fleur

Dear PCI recommenders,

Many thanks for your feedback. After a careful reading of the review, we mostly agree with your comments and remark and we modified the manuscripts in this regard. Moreover, since the submission on BiorXiv we received many comments from physicists using similar methods in their field of work. To our relief they did not find any outrageous mistakes and pointed to previous works showing the pros and flaws of this way of estimating entropy. Therefore, we added many references and added some controls. By taking into account your comments and theirs, we hope this version is improved and will answer your previous concerns.

In short: **i**) we agree that this method is more intended for a private back-to-the-envelope way of estimating the evolution of entropy and mutual information. If the experimenter wants the real value, she/he will have to do the calculations by using for example the quadratic extrapolations method; **ii**) We introduce the notion of PNG Rate, which is the size (in Bytes) of the file as PNG divided by the number of pixels of the image. We show that this PNG Rate converges slowly toward a stable value, which gives an indication about the minimal recording time and minimal number of sweeps needed for optimal entropy and mutual information estimation. It allows to compare files of different sizes as long as this minimal size is respected; **iii**) The assertion that this PNG method is not affected by the sampling disaster was misleading. It is affected but the relationship between entropy and PNG rate stays linear (Figure 2C, right). **iv**) We added a figure about the calculation of Mutual Information in the case of place fields, using simulated data. We hope it is a proof of concept that this method is valid to estimate the Mutual Information for more complex cases. **v**) We discuss ways of normalizing the size values to obtain the real entropy values from the PNG Rate calculation. However, all these methods need to use some programming skill and your own compression algorithm; we do not think they are possible by using the PNG algorithm natively.

We slightly reformatted the different reviews to allow for points-by-points answers.

Reviewer 1:

The manuscript presents a novel method for entropy estimation based on leveraging on optimal **noiseless** compression algorithms, already developed for image processing, and in particular on the PNG format. The idea is that the size of the PNG file obtained by saving a particular set of data will reflect the amount of variability present in the data and will therefore provide with an indirect estimation of the entropy content of the data. The method is based on Shannon's Source Coding Theorem, therefore approaching entropy estimation problem from the field of compressed sensing, something that as the authors rightly state, has not yet found wide use at least in the field of neuroscience. The resulting algorithm is extremely straightforward, essentially consisting in just the PNG saving step. Therefore it provides a useful tool for a fast and computationally efficient evaluation of the complexity of a signal, without having to resort to more math-heavy methods (as the math is done “for free” by the PNG compression software).

The main issue with this method is that in its present form is essentially limited to a private use by the experimenter, when needing to have a on-the-go assessment of the variation of entropy content in some preparation.

We fully agree with the reviewer in this regard, this method is more of a “Back of the envelope” way of measuring the entropy. However, it does have some benefits: as shown in Figure 2, the relationship between Entropy Rate and PNG Rate is always linear, which means we can compare data with few points and a broad range of entropy rates. This is not possible with the direct method, as high entropy rates will be skewed by the sampling disaster (Figure 2B).

Especially, it seems to me that it is particularly suitable for well controlled recordings with multiple trials of fixed length or for continuous imaging of cell cultures (and indeed, these are the examples presented in the manuscript). It would problematic to apply the method in its present form either to experiments with a larger behavioral component or, on the other hand, to compare different recordings This comes from the fact that the PNG conversion method presented here allows only for relative entropy estimation.

We decided to introduce the notion of “PNG Rate” in the figure 2, which is the value of the PNG file size (in Bytes) divided by the number of pixels in the compressed image. This value converges to an optimal, keeping constant for files of different sizes (Figure 2C). This is similar to (Martiniani et al., 2019) and allows comparing files of different sizes. For larger behavioral component, we think it is still possible with basic data modification exemplified in the new place field figure (Figure 5). Basically, we accumulate values from random-walk behavior to progressively build a 2D array. This array can be saved as PNG and the behavior will be the same than Mutual Information.

The authors rightly note how the size of the file has no direct relationship to the value of entropy. And indeed all the examples presented in the manuscript deal with the estimation of entropy or information content in data of fixed size obtained from the same system. While this is no doubt a very handy approach to obtain quick measures of entropy for large dataset, it is hard for me to see how it could be generalized to allow for comparison across different experimental setup, or the same experimental setup in different conditions. Taking the case of mutual information, consider as an example the mutual information between a place cell activity and the position of the animal in space. The conditional entropy component of the mutual information requires to consider different bins in space, for which variable amount of data would be available, given the heterogeneities of the animal behavior. Moreover comparing two neurons recorded in two different sessions will also present the problem of different exploration time and amount of data.

We originally considered about place cells, but we have really little experience in that kind of data. Moreover, it is true the differences in recording length were an issue. We now added a full figure showing the estimation of Mutual Information by PNG size in the case of place cell detection, shortly explained here:

By definition, entropy calculation can be done on any data as an array of number. However, experimental data may need to be adapted to that kind of calculation. Let us take the example of the activity of a place cell from an animal with a changing position in space: we have 2 sets of data as input (X and Y position of the animal) and one set of output (the AP frequency in Hz recorded at this point). We can imagine that all parameters have the same sampling rate. We can transform the 2D input (X, Y) data as a single input with a pairing function traversing the field where the animal is moving (Figure 5). This is similar to (Skaggs et al., 1993). We now have a single array of data with the index representing the position in the 2D space and the value being the AP Frequency. If the animal goes to the same place multiple times, this will yield multiple frequency values with the same index, thus creating multiple sweeps. In the end, with multiple trials, we will obtain a 2D array of values going from 0 to maximum frequency recorded, with the column index indicating the position in space and the row index the number of time the animal went to that position. There will be "holes" as we can imagine the animal will not go to every position the same amount of times. But we can decide to simply fill the gaps with 0. We are now exactly in the same configuration than the Figure 4C, with X value indicating the position in a 2D space. As shown in Figure 5, this follows the really same behavior than the Mutual Information computed by (Skaggs et al., 1993; Souza et al., 2018).

I am not sure whether the authors have already considered cases of this type and whether it would be possible to include a normalization step in their method, so that, for example PNG size is compared to an image of the same size obtained from a random process. As this would greatly extend the applicability of the PNG conversion, the paper would benefit from a discussion of possible solutions to this sort of situation.

We now cite different papers using compression algorithms to estimate absolute value of entropy. However, we do not think they apply to our simple method. The first method needs to have access to the dictionary created by the compression algorithm (Baronchelli et al., 2005) . However, with the PNG algorithm we do not have access to the dictionary itself. The second method needs to first estimate the maximum entropy possible by the model (Avinery et al., 2019). This is a tricky question, as the maximum entropy of the model (the neuron, for example) will be much lower than the maximum possible entropy (we cannot imagine a neuron going straight from -60 mV to +10 mV, for example). We could make a first step, calibrating a curve like the curve in Fig3A, left and then use it as a reference. But this needed to calculate the Entropy Rate by the quadratic extrapolation methods, so it might be better to simply use this method all way long. In shorts, both methods are doable but the user will need either to use its own compression algorithm or to calculate maximum entropy by more conventional mean first.

Reviewer 2:

The article proposes quantifying the entropy and mutual information of neuroscientific data based on file size after compressing the data as a PNG-file. The idea is intriguing and is presented and studied in the article fairly thoroughly. The paper has some merits, but there are multiple issues as well, which are listed below.

The authors correctly note the issue that file size does not correspond to absolute entropy values. Nevertheless, they state, already in the Abstract, that "we can reliably estimate entropy" with the method, which seems a bit like false advertising. Perhaps using "the level of entropy" in place of just "entropy" would be more accurate.

We agree and this is corrected.

The paper contains some repetition, especially in Sections 2.2 and 3.1. However, even the parts that are explained twice are not explained clearly enough; in particular, the quadratic extrapolation method requires some clarification. The terminology should be defined and stated more clearly and rigorously in the beginning; for instance, the authors use the terms "sampling", "bin" and "word" related to the variable T, but the meaning of these terms and the variable remain nevertheless unclear.

We simplified the duplicated parts and added some information, we hope it is clearer now.

The authors state the formula of Shannon entropy in Equation 1, but they do not discuss what is the probability space on which the entropy is considered in the context of the paper. It would be nice to have a discussion about this at least on a descriptive and intuitive level.

We added some clarifications in the Methods to address this point. Basically, the main issue of entropy calculation is that we cannot know which probability space is “the good one”. The entropy calculation has to be done on every possible probability spaces. The direct method does exactly that: we define multiple probability spaces with the variables Size, v and T, we calculate the entropy for all possible combinations and then we extrapolate those values to infinity.

The authors use the terms entropy and entropy rate in an interchangeable manner, although they are two quite different concepts. Which one are the authors actually interested in computing? This especially causes confusion when interpreting Figure 2A.

We agree with the reviewer and this is corrected with the addition of PNG Rate concept. Briefly, the quadratic extrapolations method allows the calculation of the entropy rate. Therefore, to be coherent, we now divide the PNG file size by the number of pixels to obtain a “PNG Size Rate” or “PNG Rate”.

Figure 2B has a couple of issues: First, the authors state that the plot of the PNG file size "follows the same curve than the entropy". However, although they appear similar, the values of the curves differ quite substantially at ~25% white pixels, where the entropy is around 50% of the maximum entropy but the file size is above 60% of the maximum. Secondly, the file size curve seems slightly asymmetric, which is surprising, as the file size should be the same regardless of whether compressing an image containing x percent white pixels or x percent black pixels. Perhaps the file size curve should be averaged over more runs of the compression?

Averaging multiple file sizes will not yield the ideal entropy curve, as values below the entropy curve are impossible (it would mean we had loss of information). What the algorithm does is estimating the entropy by its upper bound. With perfect algorithm and longer and longer recordings, the size curve will get closer and closer to the entropy curve, but always slightly higher. We modified the Figure 2 to illustrate this. It is true the curve should be symmetrical; it might be asymmetrical because of the slow convergence to a stable PNG rate.

The authors should discuss the fact that, in addition to compression, the PNG algorithm also involves filtering for 2D images, (see e.g. http://www.libpng.org/pub/png/book/chapter09.html), which affects the compression size of 2D images. The authors should note that this also affects the results gained with the 2D-images. Currently the 2D images are simply transformed into 1D signals.

This is true, the filtering is part of multiple parameters depending of the software used. Currently, data were not filtered. We added a comment in the Methods and Discussion.

The authors talk about "white noise with amplitude x". Is this a common convention for the use of the term amplitude? Perhaps some other terminology would be more standard?

We corrected it by “white noise with x possible values” or “white noise taking x values”

There are quite a few grammatical errors and typos throughout the text, so I would urge the authors to read through and edit the manuscript with care in this respect.

We hope it has been corrected, do not hesitate to point out any error precisely.

The mathematical notation has some issues. Most crucially, Equation 5 for the conditional entropy is simply incorrect.

The conditional probability should be denoted with a vertical line "|", not a forward slash "/".

The mathematical variables should always be italicized (e.g. on lines 41-42).

The two (unnumbered) Equations between lines 161 and 166 are repeated in Equations 2 and 8 unnecessarily. Also, the Equation below line 161 has the term T in all three limits.

We apologize for these mistakes and typos. They are now corrected.

Reviewer 3:

This preprint presents a png-file compression based method for estimating relative entropy of time series signals and 2-d images. The authors start with a very useful presentation of another entropy estimation method and careful demonstration of better performance of the png method. They also explore the limitations in Fig2. After this, they present the use of the png method for analysing neural data by obtaining a metric similar to mutual information for repetitive trials of electrophysiological data, and analysing dendritic complexity in micrographs of neurons. The method appears useful and simple to apply, though parts of the application should be clarified and it would be more useful to present the questions that can be addressed with the method rather than metrics that can be approximated. Only the last use case seems to be addressing a clear question, 'what is the growth state of a neural culture?'.

I have major and minor suggestions for improvement below.

Major: 1) Line 296 states that conditional entropy of signal X given signal Y can be interpreted as the noise entropy of X across trials. This seems incorrect and needs justification. Conditional entropy should express the entropy that remains in X given the knowledge of Y. It makes some intuitive sense that the noise entropy of X across trials will be affected by the driving signal Y, but to equate it with conditional entropy as defined in eq.5 seems a step too far. The main usefulness of the png method presented in the paper, estimating mutual information, relies on this interpretation.

We respectfully disagree with the reviewer. This case of Mutual Information calculation has been described multiple times in the literature we review. The original paper is (Strong et al., 1998) and this method has been used by others (Juusola and de Polavieja, 2003; de Polavieja et al., 2005; Schneidman et al., 2000).

Perhaps an overly ambitious assumption like this explains the deviations between Fig3C middle and right panels.

We are not certain what the reviewer means by "The deviation in Fig3C middle & right panels". The two curves cannot be absolutely the same for two reasons: first the PNG methods yields values in bytes and second, the algorithm being not perfect, it will get close to the "real" entropy value but will never reach it. However, the PNG curve follows a sigmoidal shape as the entropy curve does.

If the authors could come up with a more convincing method of estimating mutual information with the png approach, that would enhance the usefulness of the paper. However, given that the png method only returns relative entropy of one signal at a time it appears not suited for estimating conditional entropies. Other electrophysiological signal entropy metrics may also be equally useful for the community, such as transfer entropy, and just the relative entropy of signals, but it may be helpful to demonstrate an example of what questions can be answered with these.

Transfer entropy can be inferred from compression algorithms such as Lempel-Ziv, see for example (Kathpalia and Nagaraj, 2018). The goal of this paper was to show that the PNG method can be used in a similar way to other standard methods used in other work. It is true we did not repeat the question addressed by the original papers. For example, the figure 4A-B shows that information transfer rate grows with the number of spikes emitted by the cell (de Polavieja et al., 2005), and figure 4C that a strong synaptic drive increases mutual information as well (London et al., 2002). Both work answered their questions by using regular ways of computing entropy rate or mutual information, but we claim they could have used the PNG method and obtain the same conclusion with less effort. We added this explanation.

2) Entropy of images is presented as another key use of the method in figure 4. The estimation of culture growth stage (Fig4C) is useful. However, the usefulness of the other two presented instances is dubious. Fig4A shows a use in estimating layers in a Cajal drawing. It is unclear why one would need to do this, as the boundaries of layers 2-5 appear to be obscured in the curve, compared to just looking at the drawing.

We meant the Panel A as a simple example relating the entropy to complexity of single columns of pixels. We agree that entropy is not a perfect tool, but it allows a quantification of the complexity while the experimenter eye is a subjective tool. Moreover, it may allow determination of layers in histological preparations that are less obvious than the example of the Panel A.

Fig4B shows a use in estimating a curve similar to Sholl analysis (but different enough to not serve the same purpose). As the Sholl analysis curve describes the number of neural branches as a function of radial distance from soma, it gives us an immediately useful metric which needs no interpretation. The png metric however seems more difficult to interpret. It means something about the uncertainty/complexity of dendritic patterns at a given radius. Can the authors explain more why they believe this is useful? Demonstrating an improved identification of cell ty pes based on the png metric over the Sholl curve would make a very strong case for usefulness.

We respectfully disagree with the reviewer: a Sholl Analysis does not give the number of branches as a function of radial distance. It gives the number of intersections between concentric circles (centered on the soma) and anything on the image. It cannot make the difference between multiple straight branches and a single, convoluted one. As such, our entropy calculation is really similar for estimating branching complexity, like in our figure 4B.

As the authors pointed out, astrobiologists can use png metrics to help distinguish between biogenic and nonbiogenic rocks – could the technique offer an equally useful categorization assay for neuroscience?

We wanted to try the same approach to classify different types of cells: astrocytes, pyramidal neurons, interneurons all have really different arborization and this should be reflected in their PNG file size. However, we could not find any repository with those cells at the same resolution, same image size and same magnification.

Minor: 1) Authors explain on lines 64-70 that the quadratic extrapolation method is not the only alternative for estimating signal entropy. Why was the quadratic extrapolation method singled out as the comparison for the png method? Would the other methods perform as well or better than the png method?

To our knowledge, only 2 methods have been used in neuroscience for accurate calculation of entropy and mutual information: the direct method with quadratic extrapolation (Juusola and de Polavieja, 2003; de Polavieja et al., 2005; Schneidman et al., 2000; Strong et al., 1998) and via compression source theorem (London et al., 2002). The latter one is precisely used in the DEFLATE algorithm used in the PNG method. As we needed a comparison method, we used the quadratic extrapolations method.

Compared to PNG method, the others would be more accurate, converge faster and give the exact value of entropy in bits. But they all need some strong skills in programming and / or mathematics.

2) Could the authors mention what computer was used for the study? This is important because, e.g., line 424, it is mentioned that speed is an advantage of the png method because getting the same metric with quadratic extrapolation took 2h. This depends on the hardware.

Computation time of course depends on the hardware. Everything was calculated on the same laptop computer (intel i5-7200U @ 2.5 GHz, 8Go of RAM, SSD). We did not estimate the computational cost of the direct method in Big-O notation, but to compute hundreds of entropy values from thousands of bins each, involving log computation and then quadratic fits of these data is extremely heavy.

3) Equation between lines 161 and 162 should be the same as eq.2 on line 222.

This is corrected.

Reviewer 4:

Estimating entropy/mutual information is often a difficult and computationally expensive process, and the simple method the authors propose is attractive, because it offers a quick and easy way to compare the entropy between conditions. However, I think there is an issue, namely that the "file size does not correspond to absolute entropy values", possibly because "the PNG algorithm also involves filtering for 2D images, (see e.g. http://www.libpng.org/pub/png/book/chapter09.html), which affects the compression size of 2D images".

Since entropy and mutual information are almost always heavily dependent on the estimation method (and direct measurement is almost never possible), this is not a problem per se, but I think the authors should be extremely clear on when their method does or does not apply. The authors make strong claims in the abstract ("By simply saving the signal in PNG picture format and measuring the size of the file on the hard drive, we can reliably estimate entropy through different conditions") and although they mention the limitations themselves, both in the abstract and in the text, I think they have to be careful here. So I believe this context should be made clearer, and it should be made explicit when to use (and NOT use) the method.

We agree, we corrected the introduction and added some comments in the Discussion (see the “when to use it” paragraph).

The authors claim in the discussion that "PNG files must be all of the same dimensions, of the same dynamic range and saved with the same software" -- maybe they could think of a few typical experimental conditions where this does or does not apply?

As we introduced the notion of PNG Rate, we got rid of the size constraint. It could be possible to do the same with the dynamic range, but we would have to take the same file and increase or decrease the dynamic range until we reach a stable PNG Rate. This is the same than using the quadratic extrapolations method: modifying the dynamic range is exactly like modifying the “v” parameter. The software question is even trickier: we cannot know what every compression software does. When simply comparing Labview and PyPNG, we have different overheads. We can even imagine a “smart” software changing any parameters of the PNG package to obtain the smallest file in the end. This would be efficient in terms of saving space on the hard drive, but our aim was to provide the simplest method possible, usable without programming skills.

Secondly, I also wonder whether the size of a PNG file correlates linearly with the entropy of non-white noise files. The authors do not show this explicitly, only for white nose files, but aren't any signals that we are interested in non-white noise?

This works for any distribution, for example neuronal data in Figures 3-6 do not follow a uniform distribution. We chose uniform white noise as first examples because the distribution of all n values is 1/n, and the entropy of such a distribution is log2(n) which makes the calculations trivial.

Finally, personally I would be quite curious where the differences in png file size come from, if it is not the entropy. Do the authors have an opinion on that (because that could help others to judge when or when not one can use the method)?

From our different tries we see multiple reasons of a size variation, inherent to the use of computers:

- Zero-padding to 8-bits: a chain of characters, a dictionary word or a file size can be of any number of bits, but in the end it will be stored in 8-bits format on the hard drive. The computer will thus add 0-padding to fit the 8-bits format, which will make the size bigger.
- Pointer size: the algorithm builds a dictionary of encountered patterns and if a pattern appears again, it will put a pointer to it instead. But this pointer takes some space on the hard drive. It is even possible that the pointer itself takes more space than the pattern it

refers to. With really big dictionaries, the size of the pointer has to be big as well or to be dynamic to accommodate for all possible addresses.

- File Allocation Table: Operating Systems need to manage files efficiently, and as such have minimal cluster sizes. As an experience on Win10 (using NTFS file system), when checking the properties of the file we always see 2 numbers: ("Size" being the real size of the file and "Size on Disk", with the latter being bigger because it indicates all the used clusters). This study is using only the "Size" information, but there could be other parameters managed by the OS. We could imagine changing the file system for FAT16 for example (it has smaller cluster sizes).