# Measuring Dynamical Uncertainty with Revealed Dynamics Markov Models Supplementary Information

Aaron Bramson[1 2 3 *] Adrien Baland[1] Atsushi Iriki[1 4]

January 29, 2019

- Appendix A: Method Details
- Appendix B: Simulation Data and Analysis Details
- Appendix C: Weather Data and Analysis Details
- Appendix D: Exchange Rate Data and Analysis Details
- Appendix E: Extended Discussion and Conclusions

# A Appendix: Method Details

Here we describe in greater detail each component of the methods used in the paper. This includes: building the Revealed Dynamics Markov Model (RDMM), the proposed measures of uncertainty on that formalism, and further details on our implementations of the existing measures of uncertainty/complexity from the literature. Later appendices cover results from the individual data analyses.

## A.1 The Revealed Dynamics Markov Model

The variant of Markov modeling utilized here is called a Revealed Dynamics Markov Model (RDMM), and its representation of time series datasets is straightforward. It is similar to generating Markov chains from categorical data such as gene sequences [1], computer security profiles [2], or protein configurations [3]. It is thus distinct in both its construction and its useful applications from hidden Markov models [4, 5, 6, 7] or other Markov chain approaches [8, 9, 10]. Other network representations of time-series data exist as well: recurrence networks [11], networks of interacting dynamical units [12], networks of temporal correlations of data features (e.g., cycles [13]), and others (see [14] for a partial comparison). RDMMs are a simple formalism, but they suit the current purpose of describing the dynamics as observable from the data.

To build an RDMM first specify the number of bins $B_j$ for each dimension $j$ of the dataset to coarse-grain it into $n$ discrete states (some bins may be empty, so $n \leq \prod_j B_j$). These observed states become the nodes of the Markov model. The frequency of each observed state transition is recorded across the time series (or time window) and these frequencies are normalized by the sum of the frequencies to generate conditional probability distributions for each node. That is, for each state we know the proportions of transitions over next states, and these proportions comprise the maximum likelihood estimates for the transition probabilities of the Markov model. After closing any loose ends with a self-edge of weight 1, the result is a Markov model of the observed transitions through the dataset's phase space. You can see a simplified example of this process in Figure A.1.

The analyses presented in this paper use the simplest binning technique: divide the observed data range into a chosen number of equal-width bins. For a dataset $\mathbb{D}_i$ with entries $x_t$ (with $1 \leq t \leq T$ as the time index), the bin width

[1]Laboratory for Symbolic Cognitive Development, Center for Biosystems Dynamics Research, 2-1 Hirosawa, Wakoshi Saitama 351-0198, Japan
[2]Department of General Economics, Gent University, Tweekerkenstraat 2, 9000 Gent, Belgium
[3]Department of Software and Information Systems, UNC Charlotte, Woodward 341, 9201 University City Blvd., Charlotte, NC 28223, USA
[4]RIKEN-NTU Research Centre for Human Biology, Lee Kong Chian School of Medicine, Nanyang Technological University, 50 Nanyang Avenue 639798, Singapore
*Correspondence to: aaron.bramson@riken.jp

$W_j$ for a number $B_j$ of bins is therefore determined by:

$$W_j = \frac{\max_x \mathbb{D}_i - \min_x \mathbb{D}_i}{B_j - 1} \,. \tag{1}$$

Note that we place $x_t$ in bin $b_k$ if its value is $b_k \leq x_t < b_{k+1}$ except for the maximum point which is included in the highest bin.

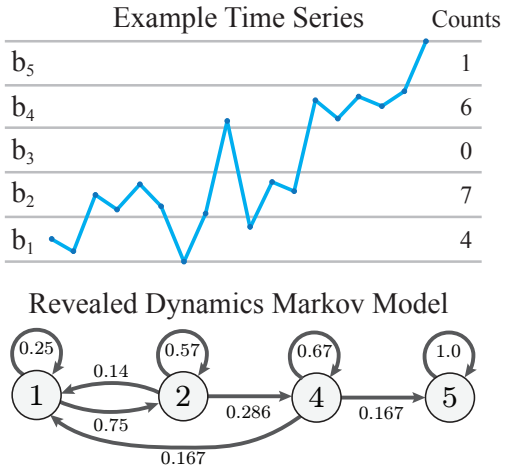



Figure A.1: Demonstration of applying the RDMM method to a small sequence of data. Note that there are no observations in bin 3, so no node is created for it. The probabilities for each exit transition reflect the proportions of bin-to-bin changes except for bin 5's terminating self-loop.

In our robustness testing we explore changes in each of the uncertainty measures across a range of binning values to establish each measure's sensitivity to binning. There is also a question about binning and cross-data normalization. For example, when we use weather data from four U.S. cities to demonstrate the technique, it is necessary to used a shared binning and normalization. If we bin them separately, then all cities' data will get the same number of bins, and (potentially) nodes, and (potentially) similar levels of uncertainty. However, if use a shared binning, then (for example) San Diego's temperature range occupies a small subset of the range of New York, which more accurately reflects their relative phase space. As we will see in that section, making a comparison of uncertainty on the same scale requires shared binning and a shared normalization — this consideration resurfaces when we analyze RDMMs of sliding time windows in future work.

Although here we mainly apply the technique to single instances of 1-variable time series data, the RDMM approach naturally generalizes to multidimensional time-synced data and to a simultaneous analysis of multiple trials into a single RDMM. We demonstrate the multidimensional application using exchange rates below. Combining multiple trials simply requires a shared binning and combining the transition frequency counts all the trials before determining the conditional probability distributions for each state.

When using one or multiple dimensions, equal-width bins are the easiest and most intuitive; however, other binning approaches can been used as well: equal-contents bins, agglomerative binning, clustering, etc. These other binning approaches can soften the curse of dimensionality because the RDMM only includes nodes for *observed combinations* of binned values. In high-dimensional applications (outside the scope of this paper) we perform adaptive binning to ensure sufficiently high confidence on the distribution of transition probabilities of each node, but this complication is unnecessary for our focus on testing/comparing uncertainty measures here. regardless of the binning method, nodes of the RDMM represent equivalence classes of values (i.e., those data values within the same bin) that define the states of the translated time series. This coarse-graining mimics a loss of resolution and/or down-sampling that is commonly required of large-scale analyses in the natural and life sciences (e.g. neurological or sensor web data).

In an RDMM, node $s_i$ transitions to node $s_j$ with probability $p_{ij}$. A specified set of nodes is written as $\mathbf{N}$, and the set of transition edges within $\mathbf{N}$ is denoted $\mathbf{E}$. Because Markov models are directed, reflexive networks, a fully connected set of $|\mathbf{N}| = n$ nodes has $n^2$ edges. No measure here is sensitive to a node's in-degree, so $k_i$ will be used to refer solely to a node's *out-degree*, and $\mathbf{k}_i$ to the set of nodes $\cup_{s_j \in \mathbf{s}}\, p_{ij} > 0$ (thus $k_i = |\mathbf{k}_i|$). Self-loops count in the out-degree of a node.

## A.2 RDMM Measures of Dynamical Uncertainty

This section presents multiple measures that quantify the magnitude of dynamical uncertainty in an RDMM. The Markov model representation enables techniques that combine structural measures of weighted directed graphs and probabilistic measures of stochastic processes. The mathematics necessary for the measures here are rudimentary: the technique takes common and well-understood mathematical ingredients and blends them together in a novel way to produce a new analytical window on system dynamics.

The formal measures need to satisfy the following criteria:

1. Uncertainty is maximized when all states are equally likely to be transitioned into from every state;

2. Uncertainty is minimized when every state has exactly one exit transition (including reflexive transitions); and

3. The measure increases monotonically with increases in uncertainty.

The first two conditions are specific to a Markov model representation. The first one requires not just randomness, but a uniform distribution across all sequential pairs of data. The second condition requires not just a deterministic process, but one in which revisits to the same block of the phase space always transition into the same block. Thus using an RDMM one can compare the regularity of different random or deterministic processes. That is, although both normal and uniform noise are fully random processes, given the same bounds an RDMM analysis will rate the normal distribution as less uncertain.

From an information-content interpretation these measures are similar to that of Kolmogorov complexity in which the "program language" is a Markov model, and what these measures tell us is how complicated a model has to be in order to capture the time series data. RDMMs are intended to be descriptive (rather than predictive) models of the time series, so what is important is that the measures presented below track the uncertainty properties of the data, rather than that the data-generating process actually be approximately Markovian. So the weather may be perfectly predictable by some non-Markovian generative weather model, and although the RDMM will never capture that model, these measures on the RDMM will tell us how complicated that generative model needs to be. This interpretation is not the only concept of time series uncertainty, but it is a common one.

### A.2.1 Entropy

Across the sciences, the default measure of uncertainty is entropy, and more specifically Shannon entropy. Though Shannon entropy is commonly applied to Markov processes (including in Shannon's original description [15]), those Markov models are constructed in a distinct way from the RDMMs employed here. For example, RDMMs can be non-ergodic, may have anywhere between zero and $T$ source nodes, have multiple components, and differ in other ways because they are built/used in distinct ways (especially when used to combine multiple times series into one RDMM). These differences notwithstanding, the Shannon entropy measure is applicable and potentially useful here upon appropriate reinterpretation.

Entropy can be applied to individual nodes or to the whole system. For the local measure we calculate the entropy of the edge weights of the existing out-going transitions, but this doesn't adjust for the number of possible edges. For this application one must realize that missing links in the Markov model also provide information about the uncertainty of the system. As such, the appropriate measure of uncertainty normalizes over all $n$ possible connections. The maximum entropy value occurs when a node's exit transitions are all equally weighted at $p_{ij} = 1/k_i = 1/n$. The equation for calculating normalized local entropy for each node is

$$h(s_i) = \frac{1}{\ln \frac{1}{n}} \sum_{s_j \in \mathbf{k_i}} p_{ij} \ln p_{ij} \ . \tag{2}$$

The local measure is useful in contexts where one is choosing states or paths for future operation of the modeled system, or in which system uncertainty should be weighted by some node property (e.g. frequency, centrality, or a context-specific state variable). The distribution of node-specific entropy values can also provide another layer of uncertainty assessment. We don't use the local measure in this paper

We also recommend taking the square root of the result from the equation in order to scale up the values. Because the measures (this one and all of the RDMM measures presented here) are normalized by $n$ and the systems tend to be sparse rather than dense, the measure values are often very low. Taking the square root is a common approach to better differentiate small values when measures range from 0 to 1 because it is a monotonic transformation that preserves the ordering and the normalized range. The rooting step is optional and additional, so throughout we define the measures without that step and state it explicitly when applied in our results section.

System entropy, $H(\mathbf{N})$, is equal to the mean of the node's individual measures (before being square rooted), but the system's entropy can be calculated by either averaging over all the individual node transition entropies or by calculating the normalized entropy of all the edges directly.

$$H(\mathbf{S}) = \frac{1}{n \ln \frac{1}{n}} \sum_{i \in \mathbf{E}} p_i \log p_i \ . \tag{3}$$

The normalization factor matches a uniform complete graph where each edge has probability $1/n$. Because all of the $n^2$ edges in $\mathbf{E}$ have this weight, the summation results in $n \ln 1/n$ so the maximum value for $H$ is 1. In a minimum entropy system there are exactly $n$ edges, each with probability 1 (the rest are 0). $\ln 1 = 0$, so $H$ yields 0 for the minimum uncertainty case.

Note that the square root of the average local entropies is not generally equal to the average of the squared local entropies: $\sqrt{\frac{\sum h(s_i)}{n}} \neq \frac{\sum \sqrt{h(s_i)}}{n}$. In our analyses below we only report the square root of the global measures, but we make use of the pure versions in discussing their relative scaling properties.

### A.2.2 Uniformity

As an alternative to entropy we can calculate the uniformity of the nodes' exit probabilities: a measure of the edge weights' divergence from a uniform distribution of $p_{ij} = 1/k_i = 1/n$. The calculation used here is similar to calculating the $\chi^2$ test statistic for a discrete uniform dis-

tribution. The local uniformity of a node $s_i$ is

$$U(s_i) = 1 - \frac{n}{2n-2} \sum_{j \in \mathbf{k}_i} \left| p_{ij} - \frac{1}{n} \right|^q \tag{4}$$

in which we set $q = 1$.

We can directly calculate the global uniformity that compares deviations from a complete graph with uniform edge weights. This calculation is also largely analogous to the $\chi^2$ test statistic because direct global uniformity is equivalent to the mean of the nodes' local (pre-square rooted) uniformities.

$$GU(\mathbf{S}) = 1 - \frac{1}{2n-2} \sum_{i,j \in \mathbf{S}} \left| p_{ij} - \frac{1}{n} \right|^q . \tag{5}$$

Both measures range within $[0, 1]$ using the normalization factor $\frac{n}{2n-2}$ and are subtracted from 1 so that zero deviation from uniform random yields an uncertainty measure of one. In the applications below we also take the square root of uniformity when reporting the results for aesthetic purposes (the numbers are often very small and this helps differentiate them).

Two small technical notes: First, a measure called 'homogeneity' can be similarly built using uniformity (or entropy) as the base by using $\frac{1}{k}$ instead of $\frac{1}{n}$ as the normalization factor, but its interpretation is distinct from system uncertainty and is not covered here. Second, the uniformity measure here, though distinct in calculation from Laakso & Taagepera's measure of the effective number of political parties (used below), fulfills all their desiderata for a generalized expression of the effective number of components [16].

### A.2.3 Turbulence

The crux of the turbulence measure of uncertainty is edge density, specifically normalized weighted edge density. Basic edge density is a common measure of networks and the definition here is the same as the usual one: the percentage of possible transitions that are actually observed. Here we normalize it to the RDMM case in which each node must have at least one out-edge:

$$\frac{\frac{1}{n^2}|\mathbf{E}| - \frac{1}{n}}{1 - \frac{1}{n}} = \frac{|\mathbf{E}| - n}{n^2 - n} . \tag{6}$$

Normalized edge density already acts as an unrefined measure of dynamical uncertainty under the following interpretation: the fewer transition edges in the system, the fewer possible paths through the system dynamics, and hence the less dynamically uncertain it is. If each state can transition into many others, then (like with the common usage of 'turbulent'), there is a great deal of uncertainty regarding how the system's dynamics will unfold. The measure is unrefined because it treats all possible transitions as equally likely; i.e., ignoring their relative probabilities.

**Effective Degree.** Transition probabilities clearly play a role in determining dynamical uncertainty. For example, if all but one of each state's transitions have very small probabilities, then that system is considerably less uncertain than if all the transitions were equally probable. We clearly must use the transition probabilities to determine a weighted version of edge density.

To account for the probability distribution of exit transitions we cannot simply sum the edge weights in a Markov model (as is typical in network theory; see [17] or [18]). Instead we need a measure such that the minimal value occurs when one edge dominates and the maximal value occurs when all the edge weights are equal. One such measure is the inverse Simpson index [19]. We use the name "effective degree" to match Laakso & Taagepera's use of this measure for the effective number of political parties using proportions of representatives [16]. The effective degree of state $s_i$ equals

$$\kappa_i = \frac{1}{\sum_{j \in \mathbf{k}_i} {p_{ij}}^q} . \tag{7}$$

Below we explore different values of the parameter $q$ keeping in mind that the standard value is 2. If state $s_i$'s transitions all have the same probability, then the effective degree is equal to the out-degree $k_i$. As the proportions become more focused on one edge, the effective degree converges to 1. The variation in $\kappa_i$ values across nodes tells us whether some regions of the phase space are considerably more uncertain than others, but here we focus on system-wide assessments of uncertainty.

**Effective Edge Density.** To calculate the effective edge density one can simply replace degree with effective degree in the edge density measure. Our measure of turbulence is the effective edge density normalized between minimum effective edge density (still $1/n$) and the maximum edge density, which now depends on the parameter $q$. The resulting turbulence measure becomes

$$\frac{\frac{1}{n^2} \sum_{s_i \in \mathbf{S}} \kappa_i - \frac{1}{n}}{\left(\frac{1}{n}\right)^{2-q} - \frac{1}{n}} = \frac{\sum_{s_i \in \mathbf{S}} \kappa_i - n}{n^q - n} . \tag{8}$$

Note that when applying effective edge density to a subset of the system $\mathbf{S}'$ (as one might do to test differences of uncertainty for distinct behavioral regimes), one must set $\mathbf{k}_i$ in the effective degree calculations to sum over only those transitions within $\mathbf{S}'$ to ensure the normalization bounds are satisfied.

When $q > 2$ a node's effective out-degree is always less than its raw out-degree, and so turbulence$_3$ is always less than or equal to normalized edge density. They are equal if and only if $q = 2$ and all the exit transitions in-

cident on each node are homogeneous (and in this case turbulence$_2$ is also equal to uniformity). The difference between unweighted and weighted edges therefore corresponds to an aggregate measure of transition probability dispersion. We take the square root of the raw edge density and turbulence calculations in the empirical results presented below, but for the rest of the methods section we are using the pure measures.

### A.2.4 Normalization and Minimum Sample Size

The normalization of the measures above are based on maximum uncertainty occurring when the network is fully connected with homogeneous weights, but for a Markov model with $n$ states, there would need to be at least $T = n^2 + 1$ data points in the time series to produce that output. From this constraint we can formulate a general equation for the maximum number of equal-width bins $B_m$ capable of supporting the maximum level of uncertainty:

$$B_m = \lfloor \sqrt[2D]{T-1} \rfloor, \tag{9}$$

where $D$ is again the number of dimensions (i.e., distinct variables). For high-dimensional data the total number of bins (voxels, cells) is the same, but then divided into each dimension, so this severely limits the number of bins *per variable*. Note that in many cases there will be (possibly a large number of) empty bins, so by setting the binning parameter using this general equation we are guaranteed to reach at least the specified level for the actual number of nodes. Of course it is also possible to adaptively bin the data so that the number of nodes (instead of bins) satisfied this criteria (we do this, but not here).

Yet, this does not sufficiently address the problem because even if the data were generated from a uniform random distribution, we would not expect to see each of the $n^2$ transitions exactly once before seeing any repeats. In order for a system with $n^2$ states to reveal itself as uniform random we would need a large number of samples per state. Although theoretically we would need an infinite number, realistically we may be satisfied with $30n$ or $100n$ samples. If $\eta$ is the desired number of points per bin, then to satisfy this criterion we would need to find a number of bins so that each one had at least that number of samples in it. That could be approximated with a number of equal width bins $B_S = T/\eta$, but it would be better to use equal-contents bins set to $\eta$ points per bin. This is a confidence level consideration, and it pulls the number bins/samples supported even lower in most cases. A more sophisticated approach would start with smaller equal-contents bins (e.g., $\eta/10$) and agglomerate them until they each have $\eta$ points or the multinomial confidence level at every node exceeds some threshold; these methods are what we generally use, but it is beyond the scope of this paper.

Another potential solution is to alter the uncertainty definitions so that the uncertainty assessment is the best possible approximation given the data available. So for numbers of bins greater than $B_m$, it is possible to further normalize the measures by the maximum achievable level for a given $T$. That is not enough, as we said, because we would expect to need multiple observations per state to see the real distribution. If we need more bins than $B_S$ allows, then we would need a different normalization factor to approximate how much uncertainty is added by having fewer points than recommended. Although we do not present these normalizations here, these worries are analogous to worries of the other approaches to time series uncertainty regarding minimum series length and window sizes. And like those approaches, we first focus on presenting the method and demonstrating its use and usefulness, and then turn to the optimization of its meta-parameters in future research.

### A.2.5 RDMM Uncertainty Measure Scaling

The plots in Fig. A.2 show the pre-square root values of each RDMM measure with increasing numbers of edges per node. In the upper diagram the edge weights are homogeneous, and this shows the differences in scaling in "ideal" circumstances. In this case we can see that normalized edge density (black diagonal line), uniformity, and turbulence$_2$ are all equal and have a linear relationship between the number of edges per node and the uncertainty. Entropy, turbulence$_{0.5}$, and turbulence$_1$ exaggerate the uncertainty in sparse RDMMs while turbulence$_3$ depresses it.

The relationships in Fig. A.2 confirm that RDMM measures of uncertainty satisfy the criteria laid out for them. Although all variations are monotonic functions, the differences in scaling alters their sensitivity across the range of exit transition distributions. The bottom plot of Fig. A.2 shows an example of the values of the measures when the probability weights are heterogeneous. Importantly, uniformity and turbulence$_2$ split away from normalized edge density as the density of the RDMM increases (this will be important in our empirical analyses later).

Interestingly, when $q = 1$ in the turbulence measure it is undefined because the sum of the probabilities is always exactly 1, and hence the minimum and maximum value of the measure are both one. But as $q \to 1$, turbulence becomes equivalent to entropy. The reason for this equivalence is that for small values of $x$, the Maclaurin series for $ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \ldots$; and so, when $q = 1 + 10^{-10}$ the sum of the node probabilities differ slightly from being exactly 1, and when normalized the scaling of these values is indistinguishable from the scaling of the natural logarithm. For this reason we abandon further analyses of turbulence$_1$.

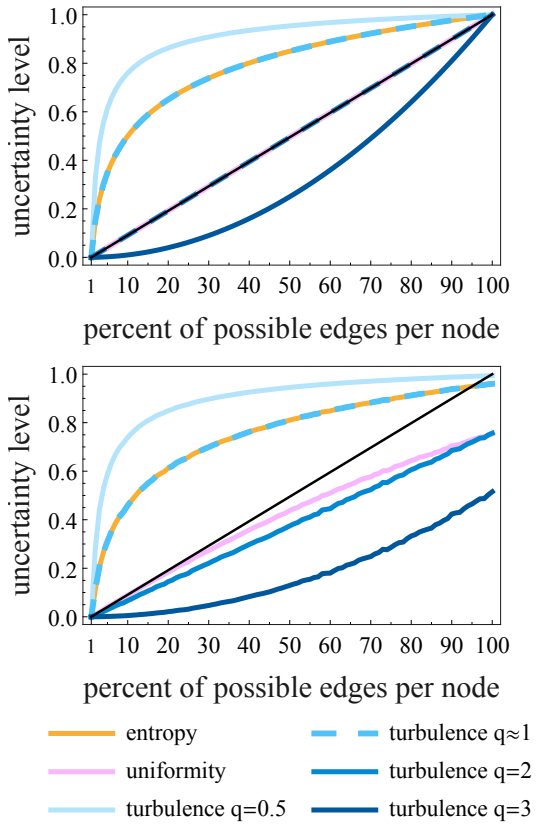



Figure A.2: Scaling comparison of the RDMM measures of uncertainty. Each one is normalized onto the $[0, 1]$ range, but has varying sensitivity to differences in the exit probability distributions within that range. Comparison made with 100 nodes and varying the number of edges per node with uniform weight $= 1/k$ (top), or assigning random proportions to each edge (bottom). Increasing the number of nodes increases the curvature (deviation away from edge density) of all measures.

## A.3 Comparison Measures

The measures offered above based on the RDMM are compared to other measures of time series uncertainty/complexity. The measures we consider here are all calculated directly from the times series, although there are others that use an intermediate model (e.g., neural network, discrete wavelet transform). For now we exclude comparisons to methods that use an intermediate model. For a time series of length $T$ (i.e., with $T$ discrete observations regardless of the time scale) we calculate each of the following measures.

**Variance.** The most basic measure of uncertainty for time series is the variance. The sample variance ignores the time ordering to measure the dispersion in the individual data points within a time-window (Eq 10). Although large random changes will produce a large variance, so will a monotonic increase in values, so variance does a poor job in differentiating these cases.

$$\frac{1}{T}\sum_{t=1}^{T}(x_t - \bar{x})^2 \tag{10}$$

**Volatility.** Volatility is a measure of variation common in finance calculated from the log returns of the time series. First we generate the time series of returns from the value data using $R_t = \ln \frac{x_t}{x_{t-1}}$. Then we find the mean log return $\bar{R}$ to calculate the volatility:

$$\frac{1}{T-1}\sum_{t=2}^{T}(R_t - \bar{R})^2 \tag{11}$$

This measure is only defined when all values in the time series are non-zero; a restriction that is safe for financial instruments but not satisfied in general. It is therefore excluded from comparisons below as appropriate.

**Jaggedness.** In order to capture the variation between adjacent increments of the time series we also calculate the jaggedness (or roughness) of the time series. Although high jaggedness means the time series is far from monotonic behavior, and hence it is distinct from variance and volatility, jaggedness is not a measure of uncertainty in the sense that regular oscillatory patterns are likely more jagged than random ones.

$$\frac{1}{T-1}\sum_{t=2}^{T}|x_t - x_{t-1}| \,. \tag{12}$$

**Approximate Entropy.** ApEn is a simplified and more practical version of the Kolmogorov-Sinai measure of uncertainty [20]. The method first uses overlapping windows of length $m = 2$ and filtering level $r = 0.2\sigma$ with $\sigma$ the time series' standard deviation, to calculate for each window the proportion of the $T - m + 1$ windows that are within distance $r$. ApEn equals the average over all windows over these proportion logarithms, minus the sum of repeating this process for $m + 1 = 3$. In addition to being highly sensitive to its two parameters [21, 22], it also suffers from problems of inconsistent relative measurements [23] and bias. These limitations have led to refinements and alternatives, but ApEn itself is still frequently used in both comparisons and applications.

**Sample Entropy.** The most popular alternative to ApEn is SampEn [23] which removes the auto-correlation bias by excluding same-window comparison when computing the proportions of within distance $r$ windows. Again using $m = 2$ and $r = 0.2\sigma$, let $A$ and $B$ be the number of segment comparisons with distance less than $r$ and lengths $m + 1$

and $m$ respectively, then SampEn equals

$$-\ln \frac{A}{B} \,. \tag{13}$$

**Arctan Normalization.** Each of the previous five measures is bounded below by zero, but not bounded above by one. To ease comparison we further apply the function

$$\frac{2 \arctan x}{\pi} \tag{14}$$

to normalize each measure value $x$ to within the range $[0, 1]$. Although this affects the values reported (e.g., in comparison to previous studies), this does not effect the relative values nor the correlations among the measures.

**Permutation Entropy.** Another measure that directly assesses the uncertainty (regularity, chaos, randomness) of time series data is based on the reoccurrence rates of short-term regularities [24]. Permutation entropy (PermEn) cuts the time series into $m$-tuples of chosen $m$ and amount of overlap; we chose $m = 5$ and an overlap of 4 (i.e., sliding the window 1 step for each tuple). For every collection of 5 points, assign them a rank by value (ties are broken by order of appearance). For each of the $m!$ types of permutations of order ranks $\pi_i$, calculate its observed proportion

$$p(\pi_i) = \sum_{j}^{T-m+1} \frac{\delta_{j\pi_i}}{T - m + 1} \tag{15}$$

in which $\delta_{j\pi_i}$ is the Kronecker delta. One then calculates the entropy from these proportions of permutations types, which we normalize to achieve a value between 0 and 1:

$$-\frac{1}{\ln m!} \sum_{i=1}^{m!} p(\pi_i) \ln p(\pi_i) \tag{16}$$

**Incremental Entropy** Following refinements on permutation entropy to handle the magnitudes of the changes [25], [26] added the capability to account for both the sign and amount of change. These two features are mapped into a two-letter "words" for each incremental pair. The incremental entropy (IncrEn) is calculated as the Shanon entropy of these words. In this way IncrEn is sensitive to patterns that ApEn, SampEn, and PermEn are not, including periods of constant values. We apply this approach using window length parameter $m = 3$, resolution (the number of bins into which the magnitudes of changes are encoded) $R = 3$, and normalize it by $\log(2R + 1)^m$.

**Permutation Test.** Similar to the permutation entropy, the permutation test determines the proportions of each of the $m!$ possible kinds of permutations, but then calculates the $\chi^2$ test statistic based on those proportions [27]. In this case the regions are non-overlapping, and we again choose $m = 5$. Normally the test statistic is used to calculate a $p$-value for rejecting the null hypothesis that the series is different from random, but we instead normalize it to create another measure of uncertainty in the $[0, 1]$ range:

$$1 - \frac{1}{m! - 1} \sum_{i=1}^{m!} \frac{(p(\pi_i) - \frac{1}{m!})^2}{\frac{1}{m!}} \tag{17}$$

**Runs Test.** The runs test is a common method for checking non-randomness in time-series data [28, 27, 29]. The first step establishes a threshold for classifying the data into binary values; the mean and median of the time series are common, so is whether sequential pairs are increasing or decreasing — here we use the mean. This translates the time series into a sequence of 0s and 1s according to whether the data point $x_i$ is above or below $\bar{x}$. A contiguous sequence of 0s or 1s is called a 'run', and $R$ is the number of runs. $\bar{R}$ is the number of runs expected according to a binomial distribution for a given number of values above ($n_0$) and below ($n_1$) the threshold:

$$\bar{R} = \frac{2 m_0 m_1}{m_0 + m_1} + 1 \,. \tag{18}$$

The expected standard deviation of the number of runs is

$$s_R = \sqrt{\frac{2 m_0 m_1 (2 m_0 m_1 - m_0 - m_1)}{(m_0 + m_1)^2 (m_0 + m_1 - 1)}} \,, \tag{19}$$

which allows one to compute the test statistic $Z = (R - \bar{R})/s_R$. Here we are interested in measures of levels of uncertainty instead of hypothesis testing, so instead of determining the $p$-value for that test statistics, we convert it into a measure that is maximal when the runs are most random:

$$Z = -\left| \frac{R - \bar{R}}{s_R} \right| \tag{20}$$

The $Z$ score here has a maximum value of 0 and scales monotonically negative for lower uncertainty. To enhance its comparability to the other measures we further normalize it using a normalized Softplus function,

$$\frac{\ln(1 + e^Z)}{\ln 2} \,, \tag{21}$$

so that it is bound on $[0, 1]$. Thus the final measure will report close to one if the data is assessed as random, and close to zero for any pattern of non-random data.

# B Appendix: Generated Data and Analysis Details

In this appendix we provide additional details of both the simulated data used to test the robustness of the uncertainty measures and the results of those tests. These details are provided to enhance transparency and reproducibility of our analysis in light of the reality that the main article is restricted in length and many readers do not need such details to appreciate the key results.

## B.1 Generated Data

We generated seven time series with $T = 1200$ points using functions chosen to systematically explore the relative behaviors of our suite of measures. Later we explore longer time series. Plots of the first 600 points of each time series analyzed here are shown below to get a sense of their uncertainty characteristics.

The first three generated time series are fully deterministic based on the sine wave at different frequencies, but these frequency changes can dramatically effect the "perceived" uncertainty according to people and the measures. These functions are

1. a low frequency sine wave: $\sin \frac{t}{4\pi}$,
2. a basic sine wave: $\sin \frac{t}{\pi}$, and
3. a high frequency sine wave: $\sin \frac{8t}{\pi}$.

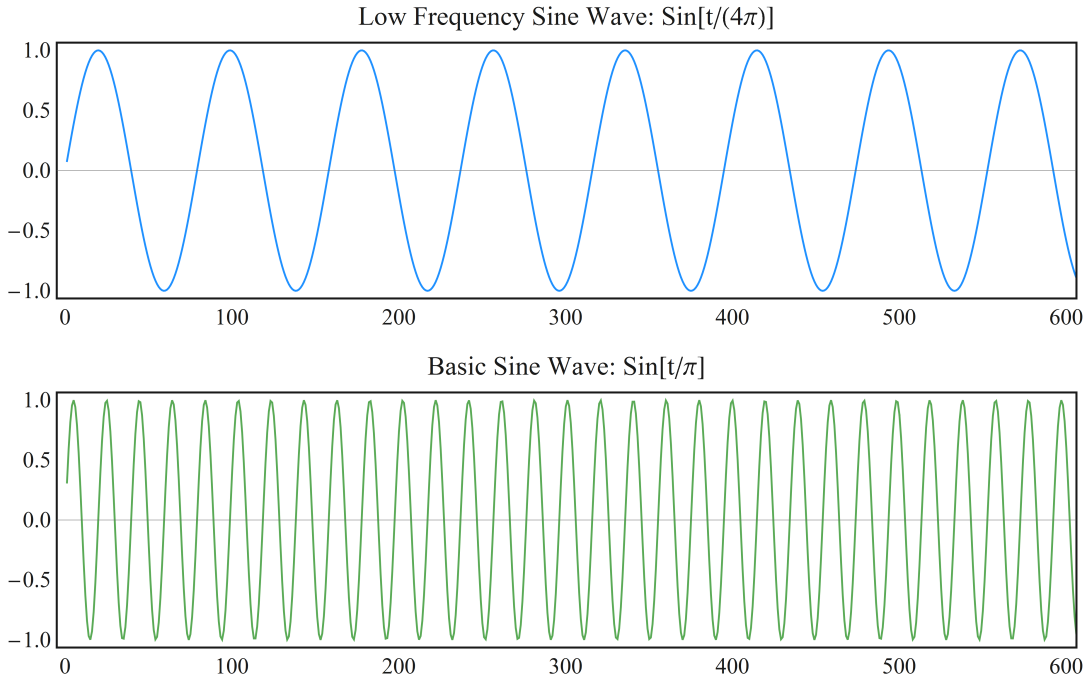
For (3), by sampling at a rate higher than the Nyquist-Shannon Theorem recommends we generate a highly oscillatory pattern that is erratic in the short-term, but regularly repeating in the medium-term.

Using $\mathcal{N}_t$ to represent a single pull from a normal distribution with $\mu = 0$ and $\sigma = 0.4$, the last four time series incorporate this random data in different ways. As

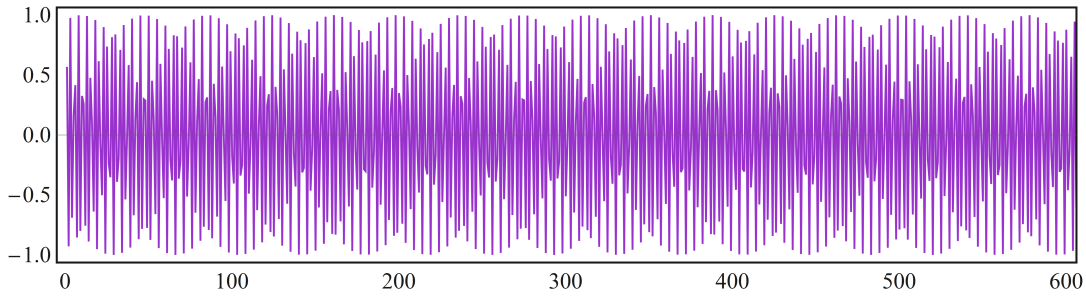
4. random noise: $\mathcal{N}_t$,
5. a random walk: $\sum_{i=1}^{t} \mathcal{N}_i$,
6. a noisy sine wave: $\sin \frac{t}{\pi} + \mathcal{N}_t$, and
7. a randomly walking sine wave: $\sin \frac{t}{\pi} + \sum_{i=1}^{t} \mathcal{N}_i$.

Although all four are equally random in the sense that the only nondeterministic element is the same series of random numbers, they each follow different behaviors.

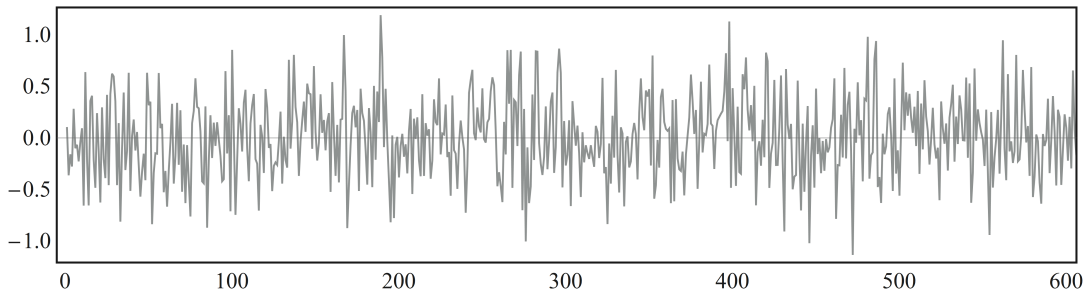
For each of these it is important to remember that in constructing the RDMM the data's $y$-values are binned into 5-100 discrete values. So, for example, at very low numbers of bins the sine part of the randomly walking sine wave would not be detectable; and the noisy sine wave may look nearly identical to the random noise. But at higher resolutions the time series very closely resemble the non discretized time series values.



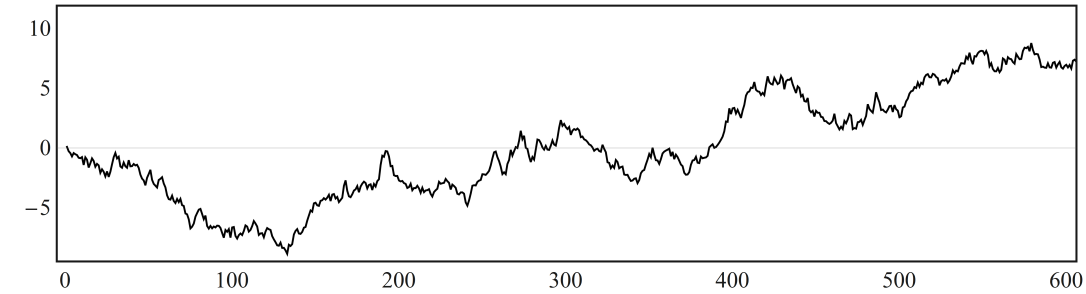
High Frequency Sine Wave: Sin[(8t)/$\pi$]



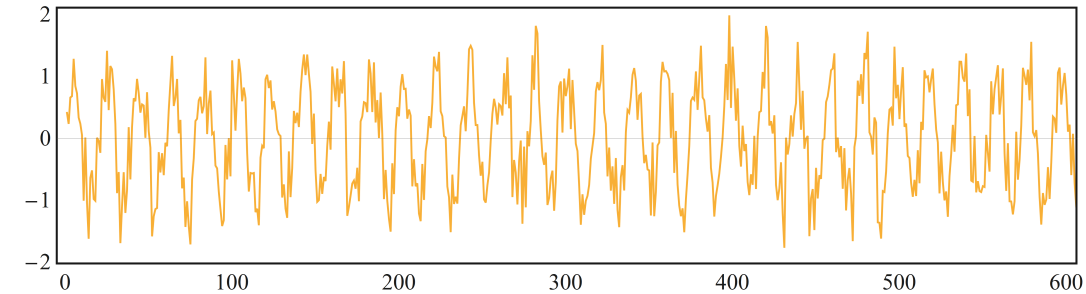

Noise: Normal[0,0.4]



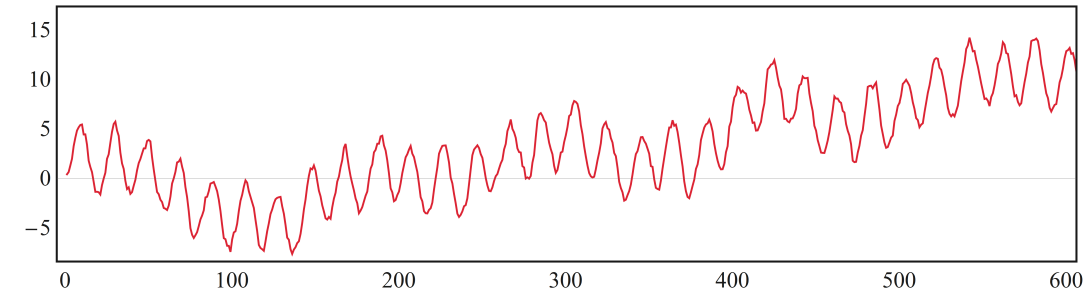

Random Walk: $\sum_{i=1}^{t}$ Normal[0,0.4]




Noisy Sine Wave: Sin t/$\pi$ + Normal[0,0.4]




Random Walking Sine Wave: Sin t/$\pi$ + $\sum_{i=1}^{t}$ Normal[0,0.4]



## B.2 RDMMs of the Generated Data

Below you can see the revealed dynamics Markov model produced from each of the simulated datasets using 35 bins per variable. Given the nature of these time series datasets, each bin has at least one exemplar except random noise (one empty bin, see Table B.1), and so the RDMMs have as many nodes as there are possible bins (35). This is not generally the case, and one advantage the of RDMM approach applied to high-dimensional data is that there are never more nodes than actually observed combinations of values (which is always $\leq T$). The use of binning further reduces the number of nodes by combining nearby value combinations (coarse-graining) as a kind of noise-reduction. This is appropriate for the use of uncertainty measures here because the concept of uncertainty we invoke is about system-wide patterns in the dynamics rather than measurements of short-term and small scale difference (as is the case with permutation entropy, for example).

| | Number of Bins | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 | 60 | 65 | 70 | 75 | 80 | 85 | 90 | 95 | 100 |
| Low Frequency Sine Wave | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Basic Sine Wave | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| High Frequency Sine Wave | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Noise | 1 | 1 | 1 | 1 | 1 | 0.967 | 0.971 | 1 | 0.978 | 0.98 | 0.964 | 0.967 | 0.938 | 0.943 | 0.947 | 0.95 | 0.941 | 0.922 | 0.916 | 0.93 |
| Random Walk | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.99 |
| Noisy Sine Wave | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.98 | 0.982 | 0.983 | 0.969 | 0.971 | 0.933 | 0.963 | 0.953 | 0.944 | 0.937 | 0.95 |
| Random Walking Sine Wave | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.988 | 1 | 1 | 0.99 |

Table B.1: Percent of bins used for each number of bins for each dataset.

The RDMM network diagrams below were generated using our web-based visualization engine which includes many options for displaying features of the RDMMs (node measures, edge measures, group measures, structural measures, dynamical properties, graphlet distributions, etc. — please contact the authors for more information). The **layout** here is based on a spring-force-type algorithm adjusted by the endogenously discovered groups of states with similar dynamical properties. The method for determining the node communities (yellow convex hulls) is based on the similarity of the distribution over future states originating at each node (the details are beyond the scope of this paper, but described in separate work). It is just worth noting that they indicate groups of nodes with similar transition patterns rather than specifically groups with higher internal than external transition rates (it's not any standard community detection algorithm).

In these examples, the **node colors** range from light blue to dark blue indicating the value of the centroid of the points within each bin they represent. So the darkest blue doesn't represent the smallest data point value, it represents the mean of the datapoints in the lowest bin. **Node size** reflects the number of observations within that bin. **Edge darkness** reflects the number of observations of that transition.

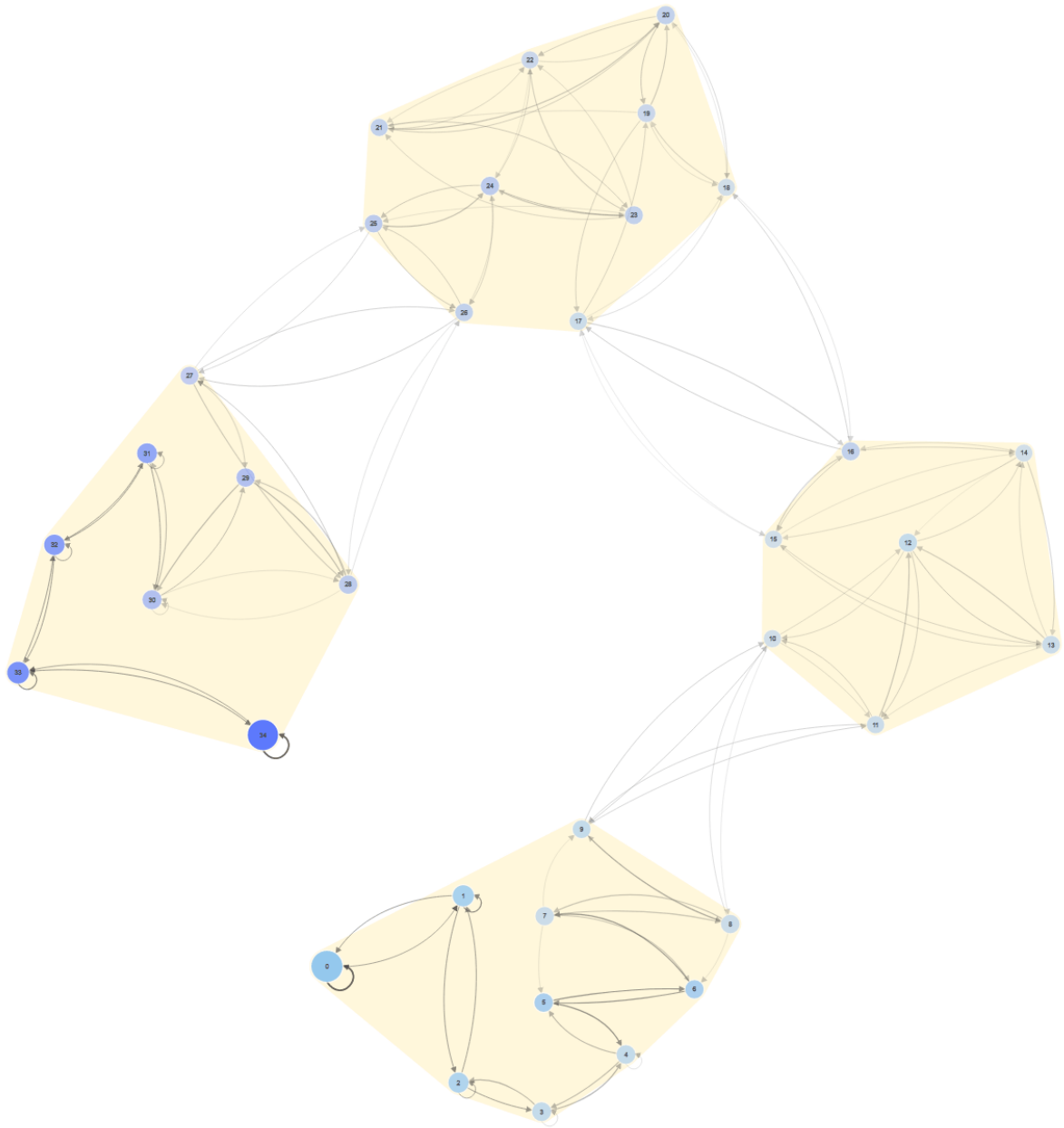


Figure B.1: **Low Frequency Sine Wave.** The Markov model clearly reveals some obvious facts about this dynamical pattern: (1) the highest and lowest bins have the greatest frequency, (2) transitions are quite local without large jumps, and (3) the four group splits occur basically where expected. Because this is a one-step Markov model, given the value of (say) 0.2, it could be increasing or decreasing, and so nearly all connections are two-way links. A two-step Markov model would distinguish these dynamics and reveal the truly deterministic nature of the sequence, but can also explode the number of nodes. It suffices that the network reveals a *highly* deterministic low level of uncertainty.

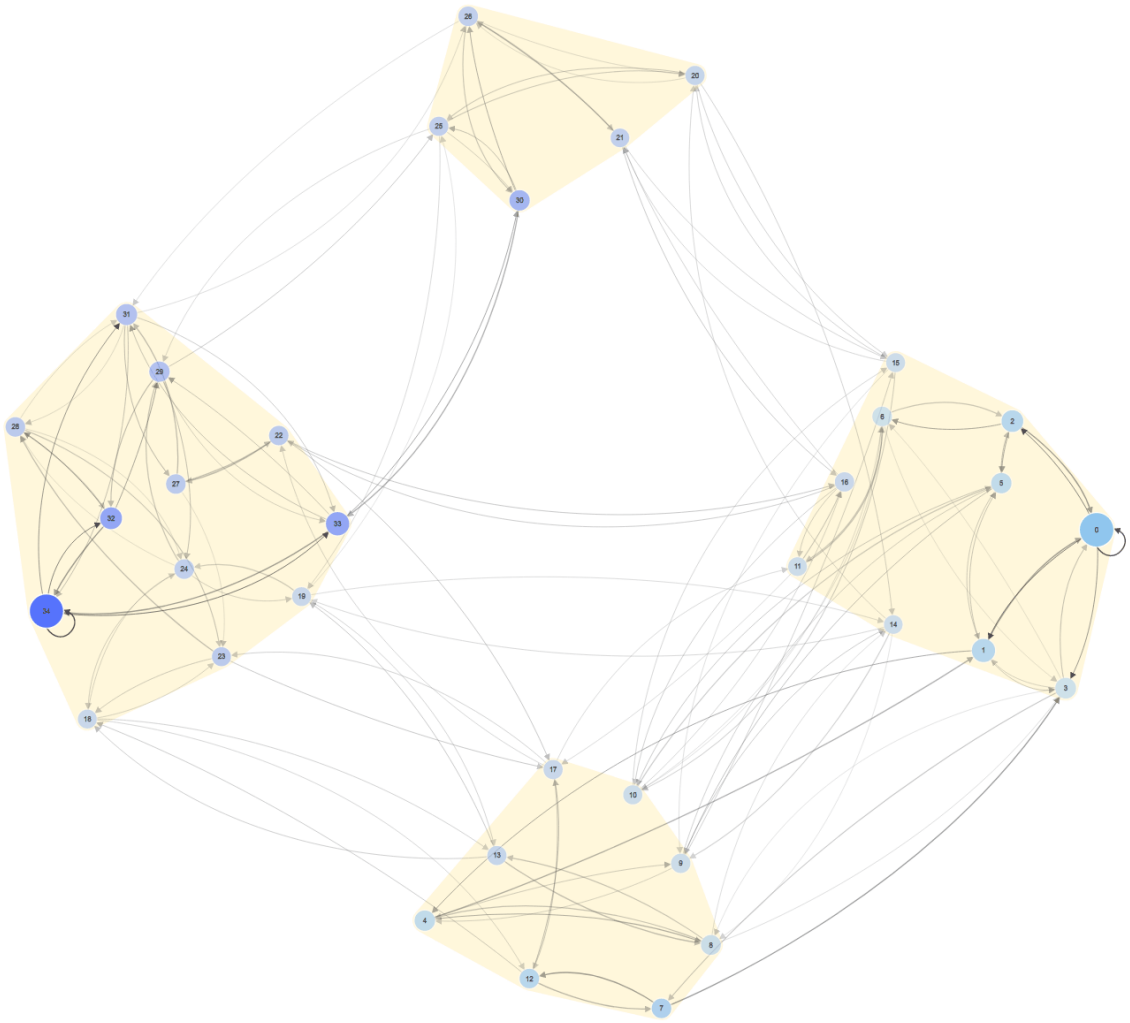


Figure B.2: **Basic Sine Wave.** With 35 bins the basic sine wave does not show up as deterministic as one may expect. This is because the bin boundaries intersect the wave form at different points in the cycle, creating many transition from nodes in the middle values. For example, node 19 is in the left group and represents the value 0.11, and it is connected to nodes representing 0.4, 0.45, −0.1, and −0.2 in each of the groups (starting in the left, going clockwise). It still has the expected heavy bins at the minimum and maximum values and four basic groups. Clearly binning effects the details of these edges, but the changes in the RDMM measures are never (here) drastic for small changes in binning. That said, it is wise to utilize binning approaches that are optimized to datasets for the best results when using RDMMs.

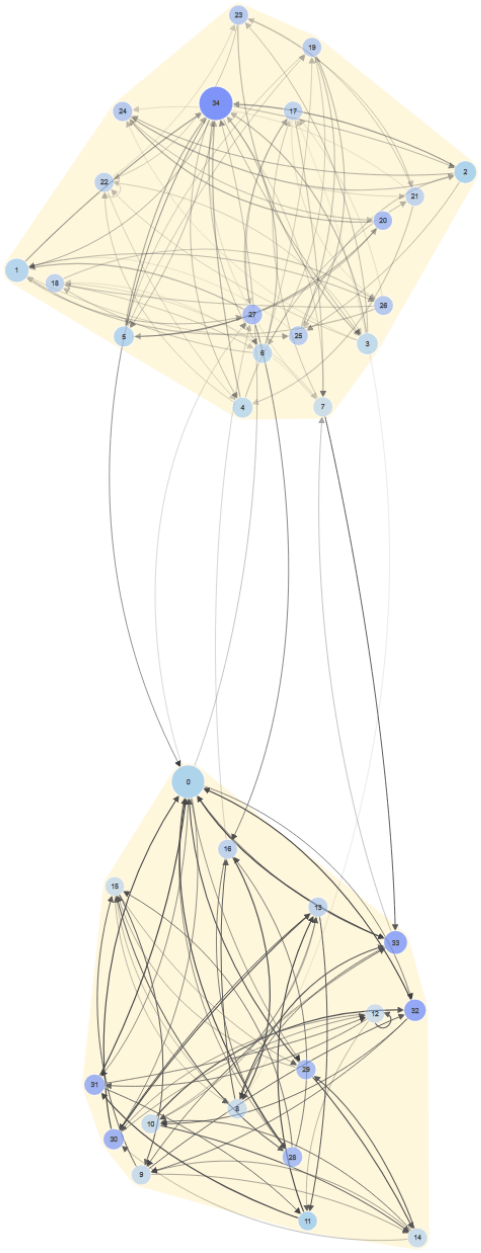


Figure B.3: **High Frequency Sine Wave.** Although this data is generated in a completely deterministic way, and the oscillations clearly follow a regular pattern, their high rate of fluctuation and different values of changing direction indicate, at least, a high level of activity. This is a case in which a Fourier transform would be the best model, but the RDMM is still capable of capturing it as a largely deterministic system. The mean raw degree is 3.51 and the mean effective degree is 2.83 which together indicate that most transitions are regularly occurring (again, due to where the bin break the cycles). Changing the bins changes which nodes appear, and which edges go between them, but the uncertainty measure values over exit transition distributions is largely preserved.

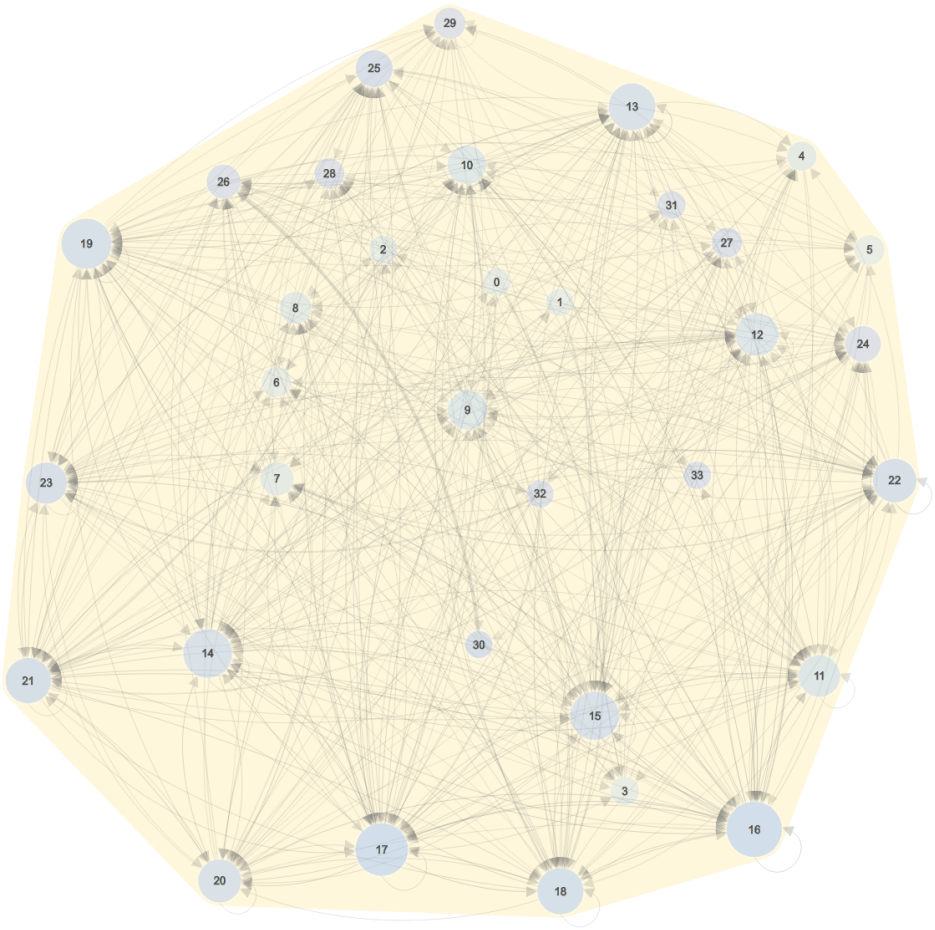



Figure B.4: **Random Noise.** The Markov model for the normally distributed random noise sequence looks about as you would expect: large numbers of edges to and from the nodes in the mid range (near zero) that have lots of observations (large nodes) and few edges to and from the rare $3\sigma$ values. Because each point is a fresh random draw from the distribution, there is no pattern to the sequence, and the states cannot be naturally organized into similarity groups.

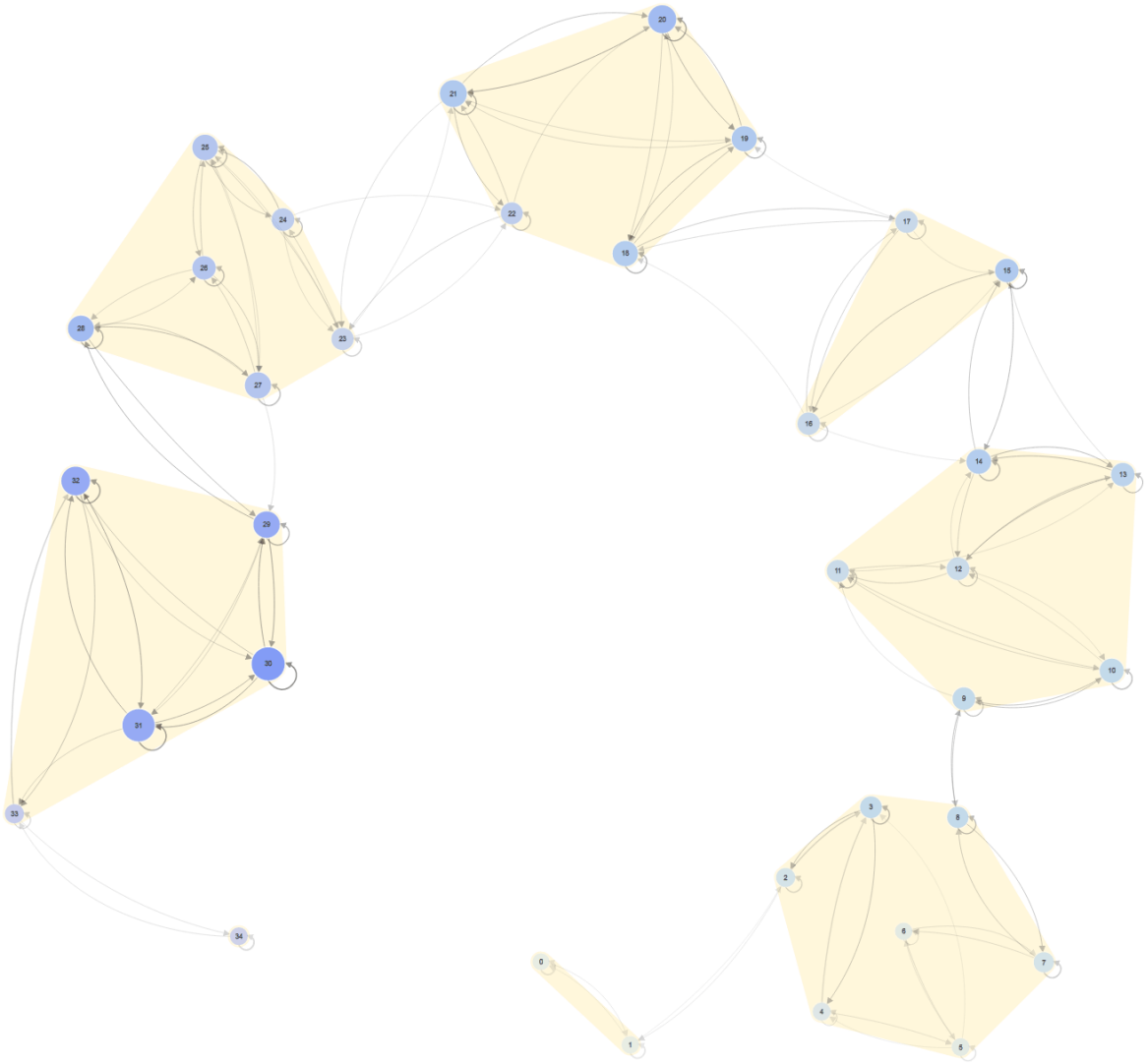


Figure B.5: **Random Walk.** The random walk shows a completely different, and quite coherent, behavior pattern even though it is generated from the same sequence of random numbers as the noise dataset. Each random walk is different, but in this case there is a clear, though not monotonic, overall trajectory of increasing value after an early dip (dark to lighter, back to darker, then lighter and lighter). The plot of the first 600 steps above confirms this story for this particular run. The size of each jump is small compared to the full range, so random walks tend to spend a lot of time within the same bin and/or bouncing between bins. This pattern is also evident from the Markov diagram through the clustered groups that clearly capture a sequence of punctuated equilibria as the dynamics unfold.

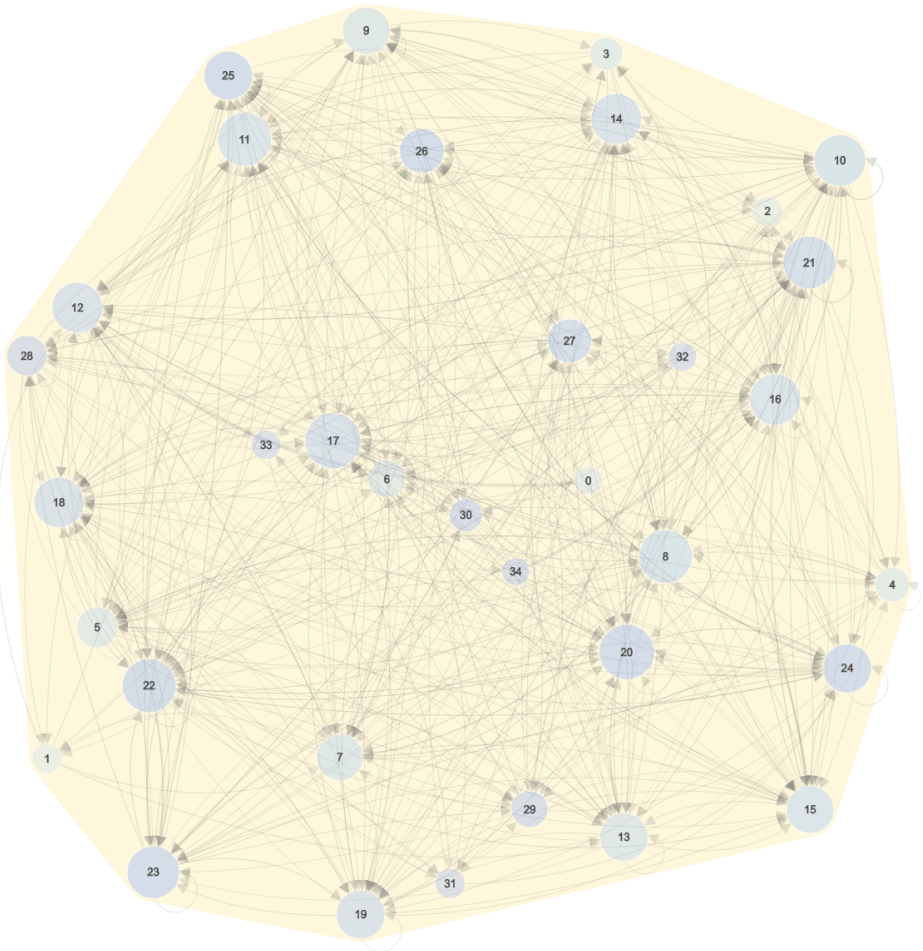



Figure B.6: **Noisy Basic Sine Wave.** Although the specific edge distribution is different, and different nodes have different frequencies of occurrence due to the underlying sine wave, the sine wave is not enough to produce a clear signal through the noise. Using structural measures from network theory besides degree and density (such as graphlets) can clearly distinguish the distributions, as can standard Markov probability estimates. Of course we already saw that the random noise and noisy sine wave have similar, but noticeably different, uncertainty profiles.

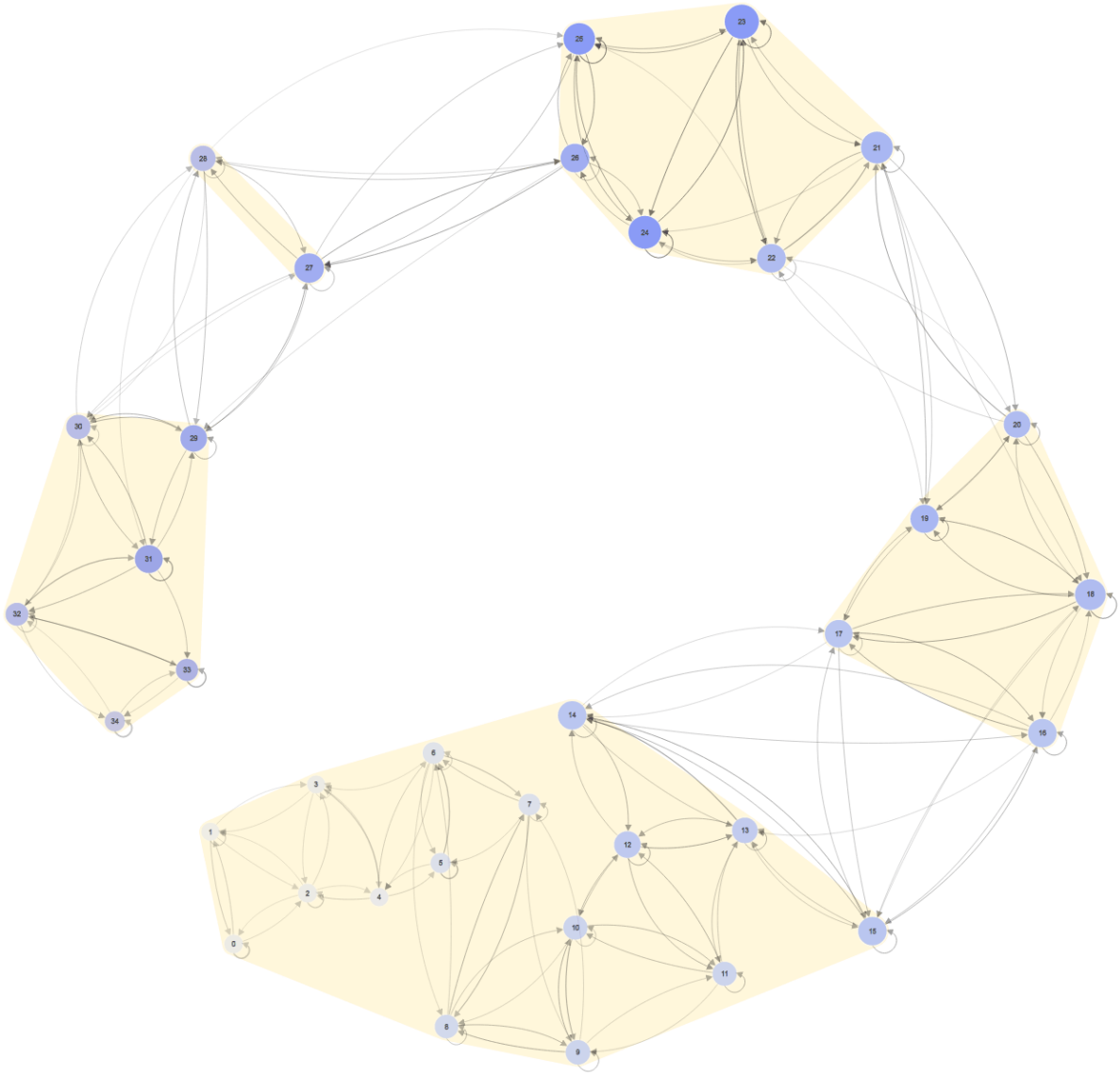


Figure B.7: **Random Walking Sine Wave.** This diagram looks unsurprisingly similar to the random walk, but with a larger occurrence of reciprocal edges due to the underlying sine function. This particular walk also goes through phases of punctuated equilibrium as it walks (because it is accumulating the same sequence of random numbers), but they are perturbed by the sine wave enough that different bins are grouped together. Thus the RDMMs are clearly distinguishable via their structural properties despite similar uncertainty profiles. One should note that although the layout and data imply a progression of points, this Markov model is actually ergodic, and this accurately reflects the standard feature of a random walk: although the pattern looks coherent, it is still possible to reach any state from any state given enough time.

## B.3 Plots of Measure Values per Measure across Different Numbers of Bins

The following plots are useful for understanding how each measure performs on the different time series; in particular, we want to judge the order of the uncertainly levels, and the sensitivity of those levels to increased resolution. Note that although all the measures range from 0 to 1 (after normalization and squashing), the *y*-axes only cover the occupied range in order to show more detail.

### B.3.1 Non-RDMM Measures

Although the first eight measures (previous ones from the literature) were not developed with binned data in mind, binning the data in this way replicates the scenario of converting an underlying analog signal into digital data of a reduced resolution — a common feature of data for which we want to assess the uncertainty. With 100 bins, the signals of these datasets are very close to their analog signal. We also applied the non-RDMM measures to the non-binned data and those values are presented as dots at $\infty$ bins in the plots below. You can see that the measure values have converged to their non-binned values by 100 bins (except for Increment Entropy).

The near constant levels for variance and jaggedness for all datasets accurately reflects those features of the data, but cannot be considered to reflect the uncertainly levels both because they do not decrease with increasing resolution and because they do not rank the datasets properly. Clearly the high frequency sine wave is the most jagged, but just as clearly the random noise is more uncertain/complex.

ApEn and SampEn have similar, but not identical, profiles for all time series. The deterministic sine waves are rated low uncertainty, but so is the random walk. The two time series with non-accumulated noise are rated as the most uncertain. For both of these measures the reported uncertainty levels are erratic across all the bin numbers in a sawtooth pattern, but mostly level after 25 bins per variable. The inability to detect the randomness of the random walk (which is usually rated less uncertain than the low-frequency sine wave) and the sawtooth sensitivity to changes in resolution reduces their viability as robust measures of uncertainty across reduced resolution. However, of the two, SampEn rates the deterministic series as more equivalent, and except for that "mistake" regarding the random walk, SampEn seems to do a better job at measuring the uncertainty of these time series. This makes sense in light of SampEn being a refinement of ApEn that removes a bias.

Increment entropy performs inconsistently, showing increasing uncertainty on some data and stable/decreasing levels on others. It rates all the deterministic ones as roughly equivalently uncertain at high resolutions, and all the stochastic ones as higher, so that matches expectations. Increment entropy is very sensitive to the magnitudes of the changes between time steps, and so there is potentially a large gap between the uncertainty measured at a 100 bin resolution and the original data stream. For these datasets the gap is small except for the low frequency sine wave. This level of sensitivity to the resolution of the data may be a benefit in some scenarios; after all, it was purportedly designed to overcome certain insensitivities in previous measures. Although I do not count it as a benefit here, it is still the case that the ranking matches the relative uncertainty of the datasets both at 100 bins and on the full series. Although the ranking is correct after 40 bins, the scores for both random walks consistently gain in uncertainty with increasing resolution. Because this is counter to what should be expected, it belies the claims that increment entropy is capturing the uncertainty or lack of information in the time series. However, if we are considering it as an alternative to the Permutation Entropy (discussed next), then based on this test it might legitimately be an improvement.

Permutation entropy and the permutation test measures are highly correlated on all datasets considered (more below in the correlation analysis), but they yield different rankings for the uncertainty of the datasets. In both cases the random walking sine wave is less uncertain than the high frequency sine wave, but for permutation entropy they converge, while for the permutation test the high frequency since wave is consistently higher. This is the result of the permutations capturing the generally regular aspect of the sine wave on the short-term, but not detecting the medium-term pattern of the high frequency sine wave. One important difference between the two is that permutation entropy uses overlapping time windows of length 5 (shift of 1), while the permutation test partitions the time interval into windows of length 5. They also both assign increasing ranks to ties based on the order of appearance, so runs of equal numbers result in minimal uncertainty. This plays out significantly in the binning results (especially the random walk) in which increasing bins reduces the amount ties, which tends to increase the uncertainly value reported. The values (and ranks) are mostly settled by 50 bins, but the incorrect assignment of high uncertainty to the deterministic sign wave counts strongly against their candidacy.

The runs test results, when interpreted in its original statistical form, is that only the random noise data stream can be accepted as being random (i.e., with runs of values above and below the mean being binomially distributed). That seems true in the sense that all the others certainly have more of a pattern with respect to being above or below the mean value. All the other datasets' results were different enough from binomial that they become are effectively 0 after the squash function is applied (they are *Z*-scores less than -10). This result is sufficient to stop

considering the runs test as a viable measure for time series uncertainty, although we present a few other results below for thoroughness. This doesn't count against the runs test as a test for randomness though.

### B.3.2 RDMM Measures.

Moving on to the six RDMM measures, there are some clear similarities among them, but important differences as well. First, recall that entropy and turbulence with $q \approx 1$ are indistinguishable (so we omit the latter). Recall also from Figure A.2 that both of these and turbulence with $q = 0.5$ have concave curves with respect to increasing uncertainty (even before applying the square root), meaning they are more sensitive to variations in sparse Markov models than variations in dense Markov models. For high numbers of bins, all the RDMMs on this data will be sparse. Unweighed edge density and uniformity have a linear relationship (before applying the square root). Turbulence with $q = 2$ is close to uniformity, but slightly convex on heterogeneous data, while turbulence with $q = 3$ is always convex, and therefore more sensitive to edge-weight variations in dense graphs. None of these scaling patterns are wrong/bad from purely theoretical considerations, so here we are interested in assessing their usefulness.

When looking at the plots below we must also keep in mind that all of the RDMM measures are normalized by the maximum possible uncertainty for a given number of bins, even if that is impossible for the number of data points. For example, because the time series here all have 1200 points, they have 1199 intervals, and hence at most 1199 edges. That is true regardless of the number of bins, which means that after $\lfloor \sqrt{1199} \rfloor = 34$ bins it is impossible for the maximum level of uncertainty to be reached. As the number of bins increases beyond this point, increasing the number of bins will *likely* decrease the measure value because it will *likely* produce an RDMM with more nodes (though it may not if the added bins are empty). So the measures desrease through this normalization in combination with the exit transition distribution becoming more focused (deterministic) through highesr resolution. For example, given enough data points (at minimum $T = B^2 + 1$, but many more are needed in expectation) the normally distributed random noise dataset would eventually produce a fully connected RDMM (though not a uniform one). By these measures, there is less uncertainty in a shorter random time series than a longer one. As described in Section A.2.4, in the future we will evaluate the use of measures normalized by the maximum possible uncertainty for a given sample size, but for now we present the results with the simple normalization.

Both entropy and turbulence with $q = 0.5$ have (non-monotonically) decreasing curves for everything except the two random walkers. The two random walking datasets produce increasing entropy up to 50 bins, and after that the leveling off may be best explained by the lack of data points just explained. The same is true for turbulence with $q = 0.5$, but it steadily increases until 65 bins. Strangely, it is these two datasets for which increasing the number of bins most clearly increases the information about the system, and for these measures to so clearly get that wrong bodes poorly for them as measures of time series uncertainty. For the other datasets the uncertainty assessments look correct, and these other datasets produce more dense RDMMs, implying that these measures are in fact *overly* sensitive to edge-weight variations in sparse Markov models.

So the reason that uncertainty increases with increasing bins (more refined information) according to these measures on these datasets is because of the concavity of their sensitivity response. All of the other datasets continue to visit the same set of nodes throughout the time series, but for the random walks there are parts of the range (i.e. subset of RDMM nodes) that are very infrequently visited. Although for a given number of bins the number of potential nodes and the number of potential transitions is the same for all measures and all datasets, the difference in sensitivity produces these divergent results in Markov models with highly heterogeneous densities.

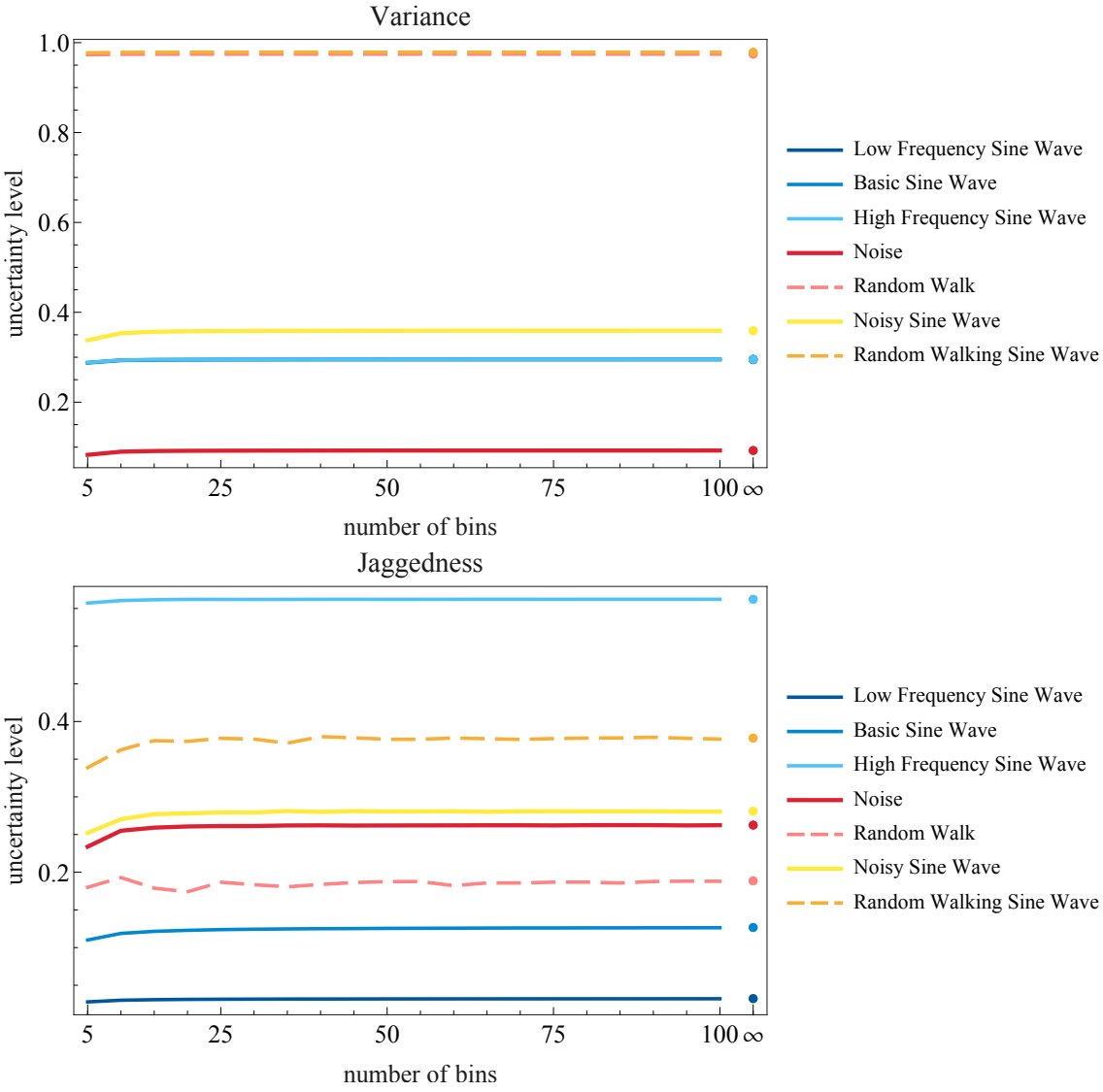
Now considering the remaining RDMM measures (unweighted edge density, uniformity, and turbulence with $q = 2$ or 3) they all have a similar overall pattern. The two random walks appear as less uncertain than the sine waves at very low number of bins, and although their assessment declines, the sine waves' uncertainty sharply declines so that they are less uncertain after about 25 bins. This is useful for determining the "sweet spot" for the number of bins to use in empirical analyses (if not based on the criteria outlined in Section A.2.4). For these datasets that sweet spot appears to be around 35, and it turns out that this is both (1) the minimum number of bins for the deterministic datasets to be revealed as less uncertain than the random ones and (2) the maximum number of bins for the maximum possible uncertainty to be reached (actually 34).

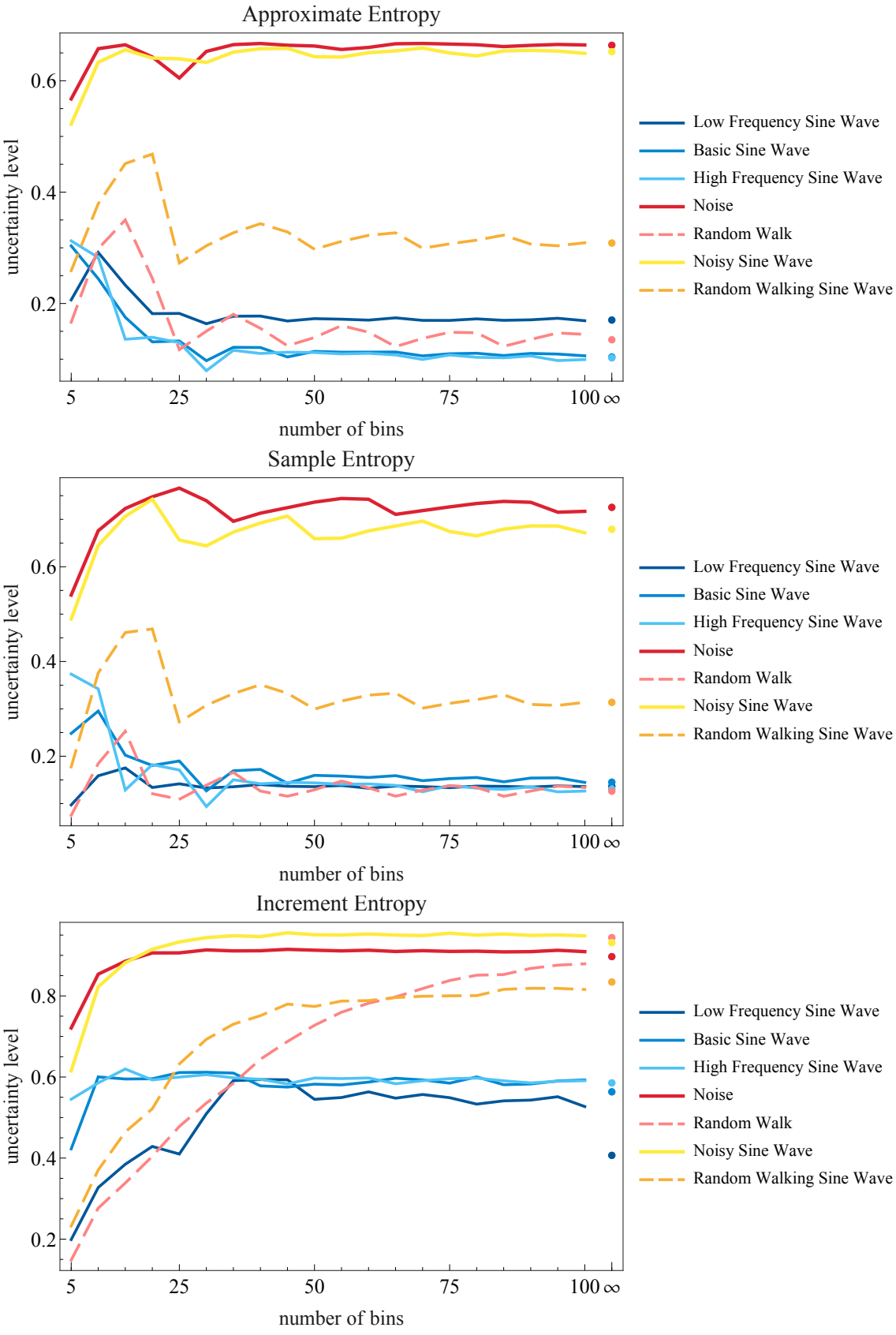
It is also the case that at 35 bins the data is captured with sufficient fidelity to track the tick-by-tick variations, and so uniformity, edge density and turbulence with $q = 2$ are all in agreement on the order of uncertainty. When $q = 3$, however, turbulence lacks sensitivity to variations in the data; i.e., the value is so close to zero for all five non-noise data streams that the differences cannot be reliably discerned or relied upon. From this we can eliminate turbulence with $q = 3$ as a reliable measure because it is overly insensitive to important variations in the data.

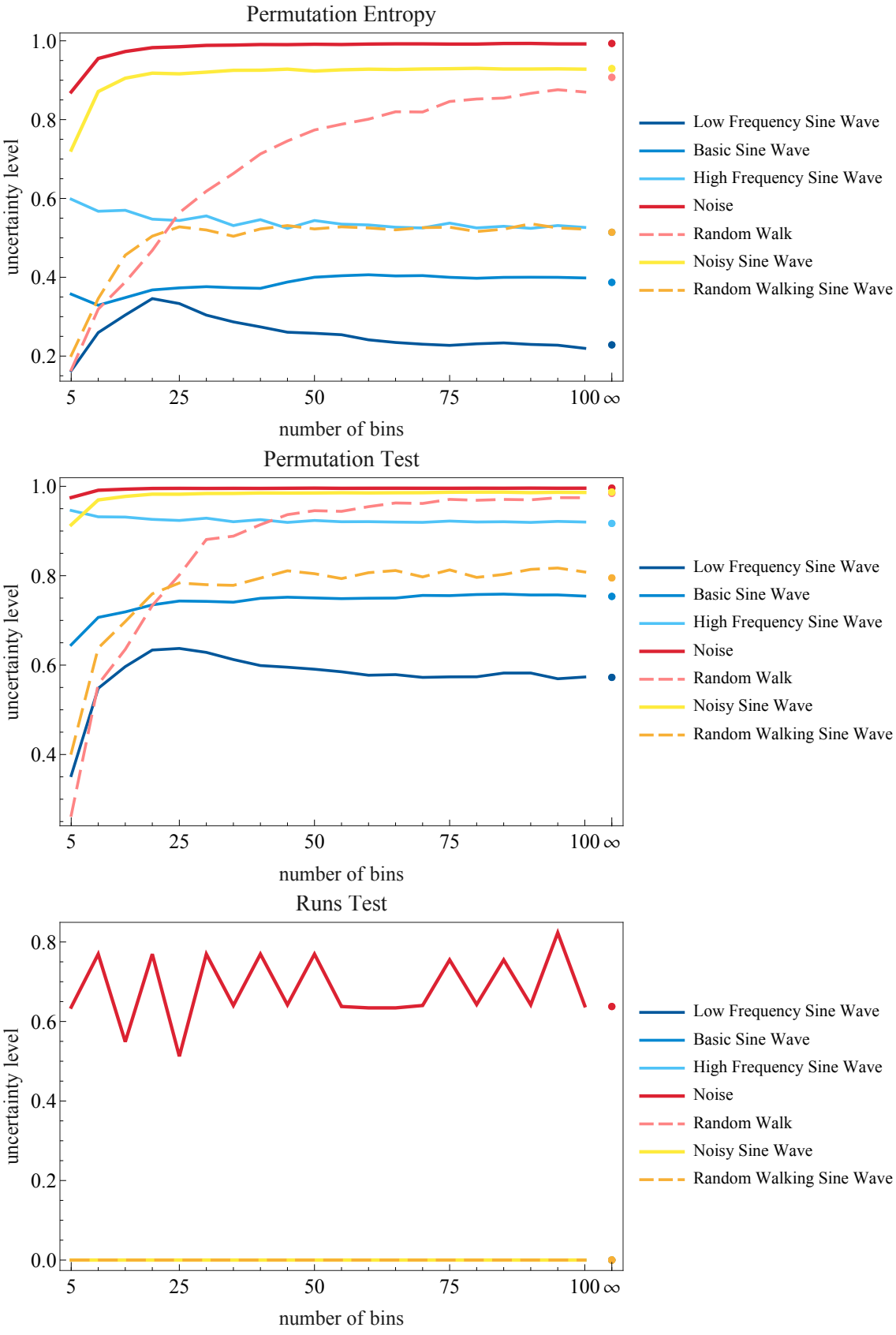
Based on this first comparison we conclude that among the non-RDMM measures, none of them match the desiderata for a measurement of uncertainty, but ApEn and SampEn are the best with respect to non-increasing uncertainty and increment entropy does the best at getting the order correct (after 40 bins). Uniformity and turbulence with $q = 2$ come out as the overall winners, but only when

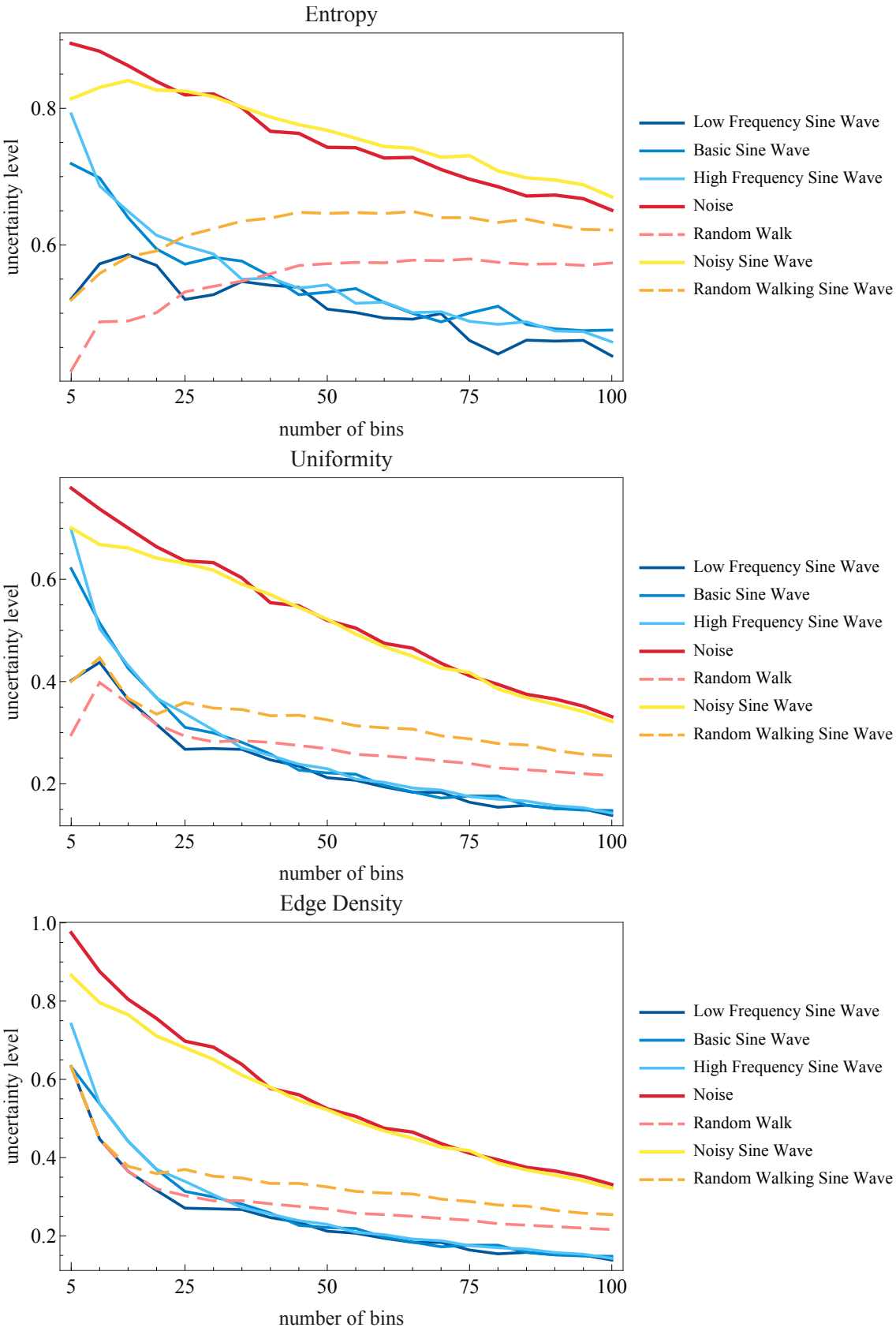
the number of bins is greater than 35. This is also true for edge density, but we can exclude that measure because we know it is insensitive to rates of transitions; its similarity here is largely due to the regular fluctuations of the sine-wave based datasets.

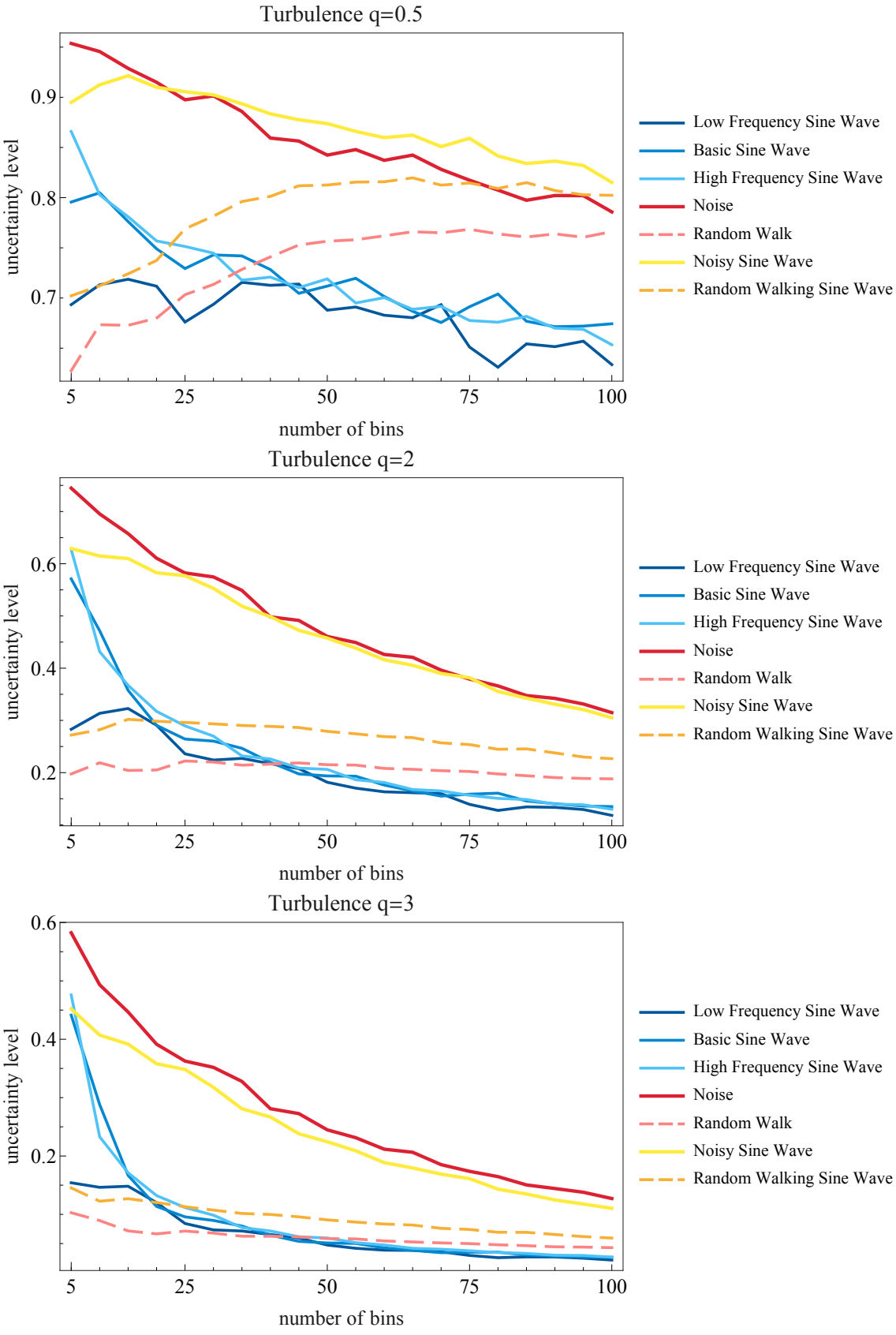
Although these results favor two of the RDMM measures, these above considerations are not the only ones we need to consider.




Approximate Entropy
uncertainty level
0.6
0.4
0.2
5
25
50
75
100
∞
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
Sample Entropy
uncertainty level
0.6
0.4
0.2
5
25
50
75
100
∞
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
Increment Entropy
uncertainty level
0.8
0.6
0.4
0.2
5
25
50
75
100
∞
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave


Permutation Entropy
uncertainty level
1.0
0.8
0.6
0.4
0.2
5
25
50
75
100
∞
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
Permutation Test
uncertainty level
1.0
0.8
0.6
0.4
5
25
50
75
100
∞
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
Runs Test
uncertainty level
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave


Entropy
uncertainty level
0.8
0.6
5
25
50
75
100
number of bins
Uniformity
uncertainty level
0.6
0.4
0.2
5
25
50
75
100
number of bins
Edge Density
uncertainty level
1.0
0.8
0.6
0.4
0.2
5
25
50
75
100
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave

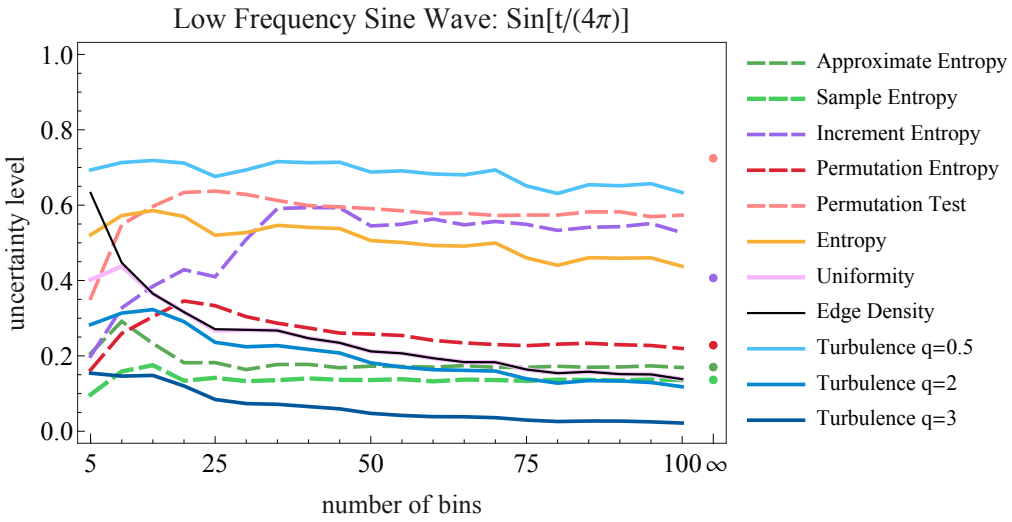

Turbulence q=0.5
uncertainty level
0.9
0.8
0.7
5
25
50
75
100
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
Turbulence q=2
uncertainty level
0.6
0.4
0.2
5
25
50
75
100
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
Turbulence q=3
uncertainty level
0.6
0.4
0.2
5
25
50
75
100
number of bins
Low Frequency Sine Wave
Basic Sine Wave
High Frequency Sine Wave
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave

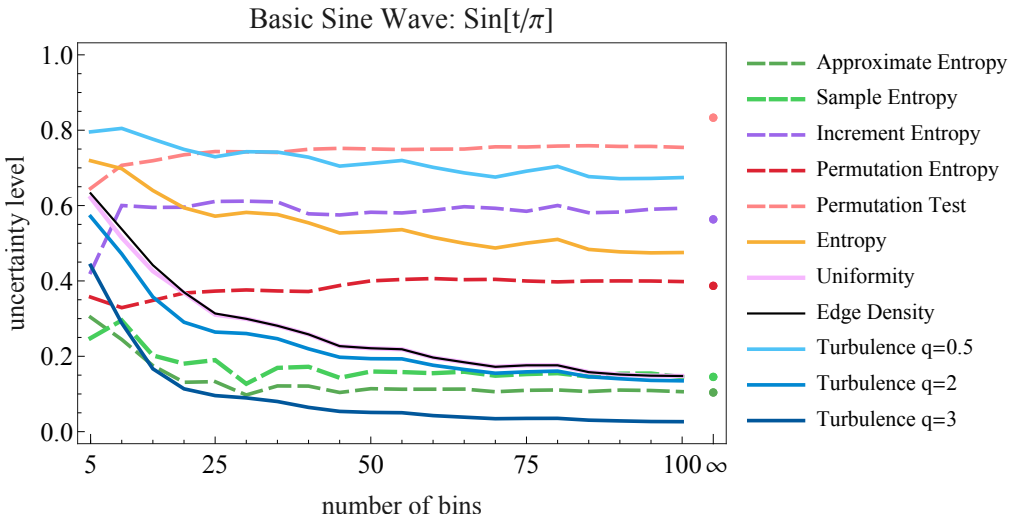
## B.4 Plots of Measure Values per Time Series across Different Numbers of Bins

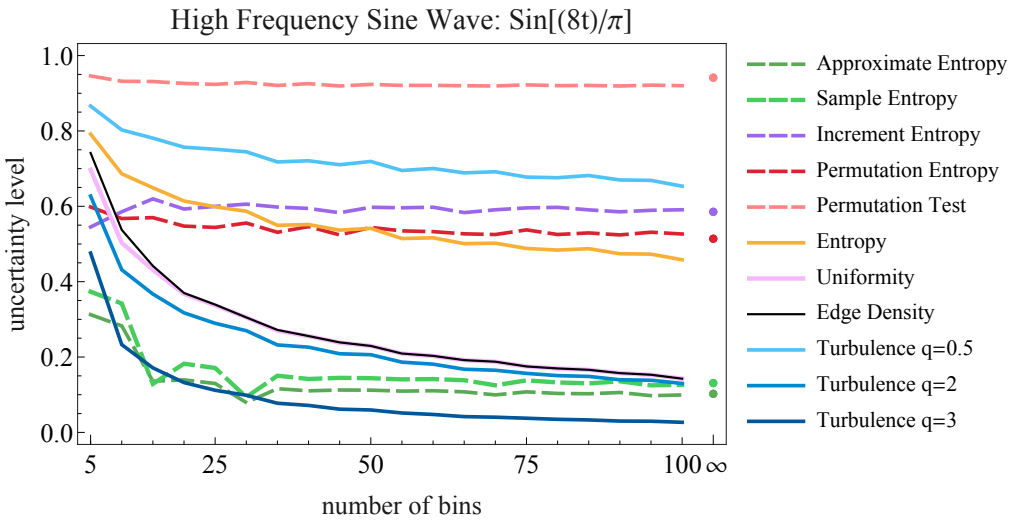
The following plots show the same results as the previous ones, but this time aggregated by dataset instead of by measure. As before, for each time series we generate the RDMM and calculate the full suite of measures using equal-width bins numbering from 5 to 100 in increments of 5. We show correlation matrices across the measures for each dataset below, but these figures are useful for comparing the level of uncertainty assessed by each measure over the range of binning. For example, that turbulence with $q = 3$ hugs the bottom of the plot while the permutation test reports near-maximal uncertainty, even though they are strongly correlated for much of the domain.

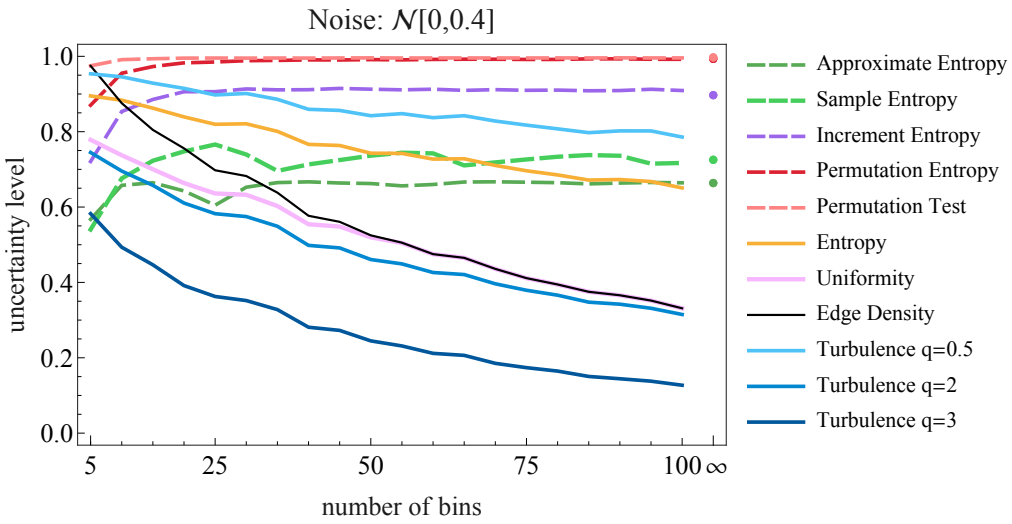
These plots indicate some obvious patterns in the comparative results. Sample entropy and approximate entropy (green lines) typically have similar patterns with respect to both the magnitude and sawtooth shape across the number of bins. Edge density (black line) is nearly monotonically decreasing for all datasets. Uniformity (pink) and turbulence with $q = 2$ and 3 (darker blues) are always less than or equal to unweighted edge density, while RDMM entropy (orange, equals turbulence with $q = 1$) and turbulence with $q = 0.5$ (light blue) follow shared, distinct trajectories. The permutation entropy and permutation test (red lines) are typically closely linked, while increment entropy (purple) follows its own course from dataset to dataset.

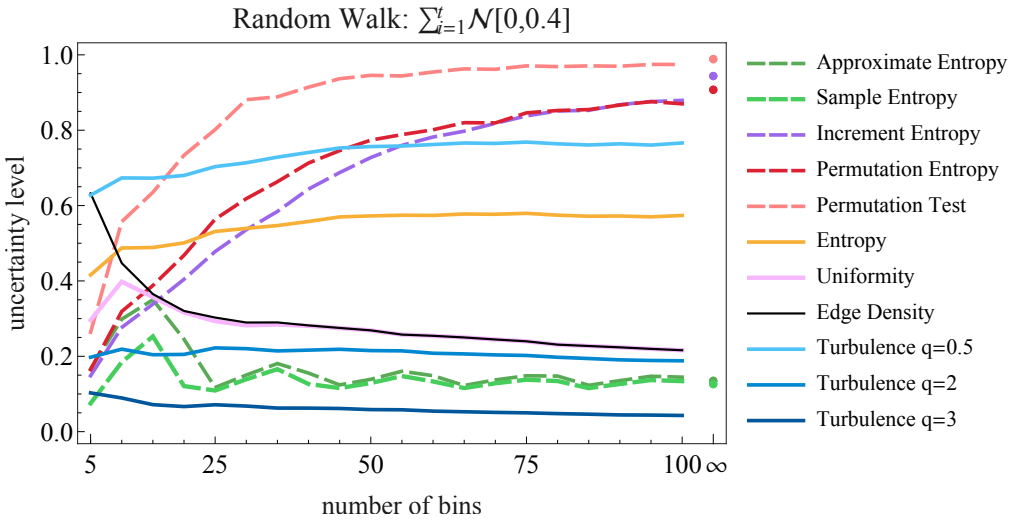
Because these plots all range from 0 to 1 on the uncertainly level, we can compare the measures to each other and to their min and max values. For example, we can see that on the random noise data, all the non-RDMM measures rate the uncertainty as greater than 0.5, but the RDMM measures all start high and drop off with increasing bins. Again, this is partly because the measures are normalized by the $n^2$ possible edges, and with only 1200 points in these series the proportion of observed edges shrinks after 35 bins. We make a similar comparison using series with 10,001 data points (below) to demonstrate that this feature is not a problem because the important thing is the relative values, and these are preserved for longer time series and different random seeds.

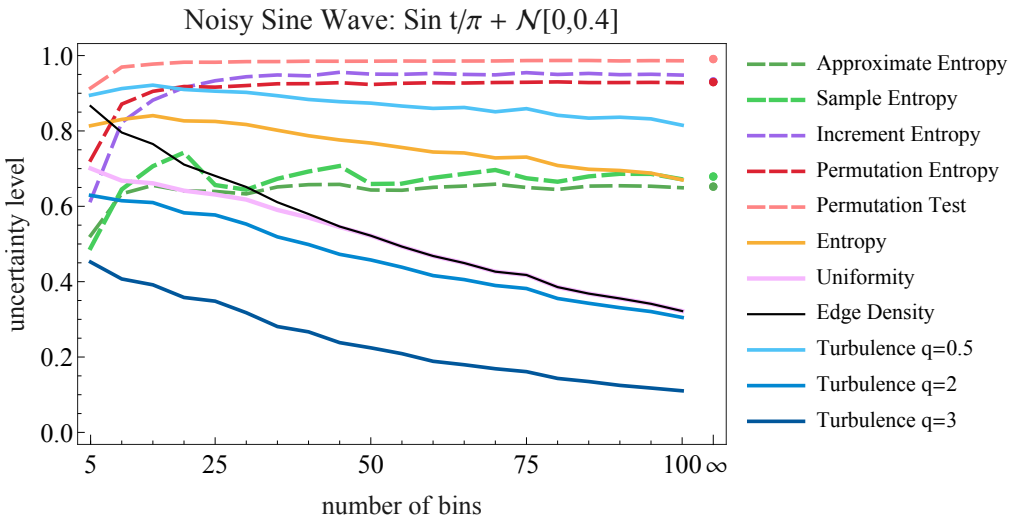


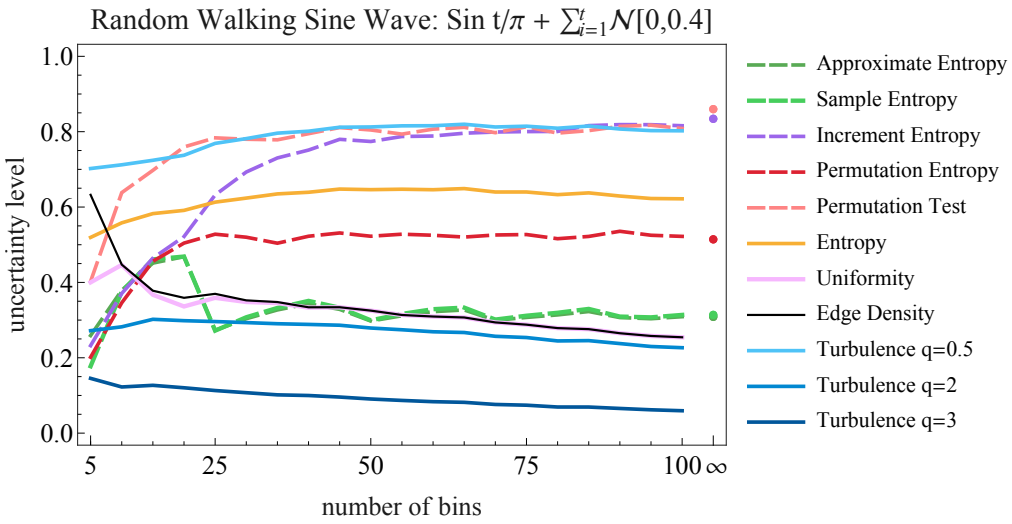

Basic Sine Wave: Sin[t/π]
uncertainty level
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3


High Frequency Sine Wave: Sin[(8t)/π]
uncertainty level
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3


Noise: 𝒩[0,0.4]
uncertainty level
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3


Random Walk: $\sum_{i=1}^{t} \mathcal{N}[0,0.4]$
uncertainty level
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3


Noisy Sine Wave: Sin t/$\pi$ + $\mathcal{N}[0,0.4]$
uncertainty level
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3


Random Walking Sine Wave: Sin t/$\pi$ + $\sum_{i=1}^{t} \mathcal{N}[0,0.4]$
uncertainty level
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
∞
number of bins
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3

## B.5 Analysis of the Consistency of Measurements across Binnings

One desiderata for any measure of uncertainty is that it should be consistent in the value it reports, and that includes robustness against different binnings of the same data. Although we employed jaggedness as a measure of uncertainty, it is actually a measure of the monotonicity, and as such it is one method to assess the consistency of the uncertainty values across different numbers of bins. One issue with this approach is that it will give higher inconsistency values to a sequence that bounces around its mean than to one that is monotonically decreasing across a large range, but we may consider the first case as more consistent. For these reason we consider multiple features when assessing a measure's consistency, and this form of consistency is only one consideration of a measures fit to our desiderata. For each measure on each dataset, we first normalize the result values onto the $[0, 1]$ range so that the relative consistency does not depend on the scale of the uncertainty assessment. Then we calculate the jaggedness of each measure's normalized uncertainty value across the binnings for each dataset.

Table B.2 shows the jaggedness across bin values for each measure for each of the generated datasets. Due to the preprocessing step, jaggedness will be zero here if and only if the value is completely constant. If it is perfectly monotonic (increasing or decreasing) the minimum value is 0.056, which appears in many places in the table. Only variance is perfectly monotonic for every dataset, but there are several measures that are highly non-monotonically varying with increasing bins. Specifically, ApEn and SampEn are highly inconsistent across binnings (we already mentioned the sawtooth pattern), and Permutation Entropy is highly inconsistent only on the high frequency sine wave.

| | Low Freq Sine | Basic Sine | High Freq Sine | Random Noise | Random Walk | Noisy Sine | Random Walking Sine |
|---|---|---|---|---|---|---|---|
| Variance | 0.056 | 0.056 | 0.056 | 0.056 | 0.056 | 0.056 | 0.056 |
| Jaggedness | 0.056 | 0.056 | 0.059 | 0.058 | 0.210 | 0.065 | 0.091 |
| Approximate Entropy | 0.113 | 0.076 | 0.077 | 0.139 | 0.164 | 0.097 | 0.165 |
| Sample Entropy | 0.122 | 0.122 | 0.102 | 0.105 | 0.174 | 0.121 | 0.140 |
| Increment Entropy | 0.080 | 0.093 | 0.148 | 0.062 | 0.056 | 0.061 | 0.057 |
| Permutation Entropy | 0.095 | 0.091 | 0.159 | 0.057 | 0.056 | 0.060 | 0.074 |
| Permutation Test | 0.073 | 0.063 | 0.135 | 0.061 | 0.056 | 0.058 | 0.070 |
| Entropy | 0.115 | 0.075 | 0.060 | 0.057 | 0.060 | 0.067 | 0.073 |
| Uniformity | 0.064 | 0.057 | 0.056 | 0.056 | 0.089 | 0.056 | 0.083 |
| Edge Density | 0.056 | 0.056 | 0.056 | 0.056 | 0.056 | 0.056 | 0.059 |
| Turbulence q=0.5 | 0.164 | 0.107 | 0.070 | 0.068 | 0.062 | 0.088 | 0.071 |
| Turbulence q=2 | 0.072 | 0.057 | 0.056 | 0.056 | 0.160 | 0.056 | 0.080 |
| Turbulence q=3 | 0.058 | 0.056 | 0.056 | 0.056 | 0.065 | 0.056 | 0.061 |

Table B.2: Non-monotonicity for each of the uncertainty measures across bin values for each of the datasets.

To assess the relative consistency of the monotonicity we calculate the mean and variance of these jaggedness values across the datasets. The comparative results are shown as shades in Figure B.8 including the rank of each measure. In terms of having consistent evaluations of uncertainty across binnings, the RDMM measures on average out-perform the previous measures, but not entirely. Turbulence with $q = 0.5$ is very erratic, and turbulence with $q = 2$ is also overly sensitive to binning on some datasets.

One worry about high consistency is a lack of sensitivity, and so the monotonicity of turbulence$_3$, variance, uniformity, and edge density here are due to their lack of sensitivity to the data rather than their consistent sensitivity to the data. Edge density very consistently decreases exactly because it is not sensitive to the edge weights, which can change considerably for different binnings. When it comes to the consistency of the measures across binnings balanced with sufficient sensitivity, RDMM entropy is a top contender, but many measures fall into the same middle-ground.

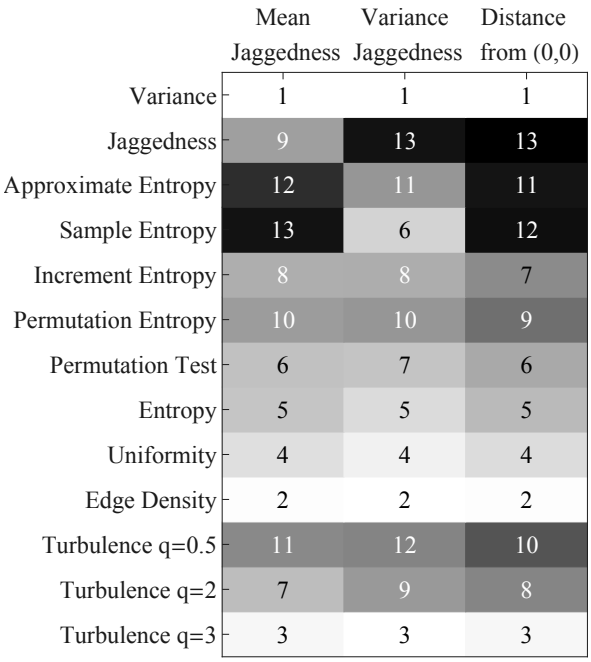



Figure B.8: Mean and variance of the jaggedness of the normalized measure values from the seven simulated datasets. Lighter shade indicate lower values and overlay text is the rank. From this we an see the relative consistency of each measure of uncertainty in terms of the consistency of increasing/decreasing values with increasing resolution.

## B.6 Effects of Reversing the Data on Measure Values

One may have the intuition that reversing the time series should not effect our assessment of its uncertainty. After all, the set and relative sequence of the numbers are the same. We contend that the order should matter, but how much it matters depends on the nature of the time series. Imagine a cat leaping onto a branch: the trajectory of the jump informs us the landing spot, but the landing spot informs us much less about the trajectory. Or you can think of crescendos in music: the rising volume and cadence lead up to an anticipated climax, but if listening to it backwards, the climax would not leave one expecting a diminuendo. Hence for many time series, running the data backwards should produce a different level of uncertainty than running it forward because the information content (and hence the uncertainty about what is going to happen next) are different.

Most of the previous measures of time series uncertainty are symmetric by design, and so they cannot reflect any predictability differences based on the order of the time series. The exception being that permutation entropy can produce different values for the reverse data when there are ties in the data. This is because ties in values are (typically) assigned ascending ranks by the order of appearance. However, this is a known flaw of that measure [30], rather than a feature. The permutation test has the same issue, and for this reason is generally recommended for real-valued data with few-to-no tied values [31]. In our case, all the non-RDMM measures yield the same values for the reversed *non*-binned data, but binning the data generates ties, which is reflected in the differences between the directions of these permutation-based measures presented in table B.3.

The construction of the Markov model, however, is inherently sensitive to the order of the data. This can be clearly seen in the following simplified example.

$$1 \to 1 \to 2 \to 2 \to 3 \to 3 \to 4 \to 5 \to 5 \to 5$$

This series produces distinct Markov models when encoded forward and backward as can be seen in Fig. B.9.

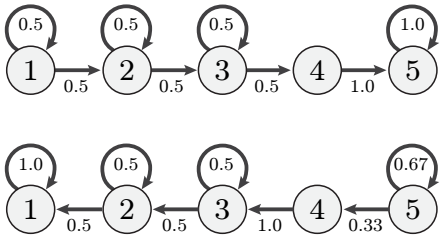



Figure B.9: Example Markov model demonstrating the difference in edge weights between running the data forward (top) and in reverse (bottom).

Ignoring the direction of the arrows, you can see the distribution of edge weights is actually quite similar. When encoding a single time series, the difference in the network-wide distribution of edge weights is tightly constrained even though individual nodes' exit transitions change dramatically (e.g., the first and last node in Fig. B.9). When encoding multiple times series into a single RDMM the reversal can have a much larger effect, but we do not cover that kind of application in this paper. The limited effect can also be seen in the small differences reported by each measure in Table B.3.

| | Low Freq Sine | Basic Sine | High Freq Sine | Random Noise | Random Walk | Noisy Sine | Random Walking Sine |
|---|---|---|---|---|---|---|---|
| Variance | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Jaggedness | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Approximate Entropy | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Sample Entropy | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Increment Entropy | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Permutation Entropy | 0.00569 | 0.000886 | 0.000894 | 0.001953 | 0.00926 | 0.004537 | 0.006696 |
| Permutation Test | 0.00892 | 0.00565 | 0.000777 | 0.00069 | 0.014063 | 0.004245 | 0.019526 |
| Runs Test | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Entropy | 0.003318 | 0.001089 | 0.000707 | 0.002521 | 0.001951 | 0.002289 | 0.001031 |
| Uniformity | 0.000067 | 0.000222 | 0.000278 | 0.003094 | 0.000243 | 0.0019 | 0.000205 |
| Edge Density | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Turbulence q=0.5 | 0.00339 | 0.00091 | 0.000841 | 0.002709 | 0.002353 | 0.001752 | 0.001197 |
| Turbulence q=2 | 0.002307 | 0.001191 | 0.000588 | 0.003084 | 0.000939 | 0.002606 | 0.001088 |
| Turbulence q=3 | 0.001084 | 0.000838 | 0.000476 | 0.004193 | 0.000391 | 0.003038 | 0.000808 |

Table B.3: Mean absolute value of the difference between running each measure forward and backward through the data across all binnings. Permutation-based measures become non-symmetric because of the ties introduced by binning.

Edge density is sensitive to the binning, but not to the edge weights, so even though it is measured on the RDMM, it is still symmetric going forward and backward. One would expect to see a bigger difference on the random walk datasets than the sine wave or noisy datasets because the walks "tell a different story" in reverse. However, at least on these samples, there is no coherent pattern in the differences.

## B.7 Correlations of Measures Values

In this section we present the pairwise correlation matrices across all measures on each of the seven generated datasets as array plots. By examining these correlations we can identify groups/classes of measures with similar scaling properties (i.e., changes in values with changes in binning). Interestingly, the sets of correlated measures are dependent on the dataset; measures that are strongly correlated on one dataset may be strongly anticorrelated on another, which undermines at least one of those measure's claim to be measuring uncertainty.

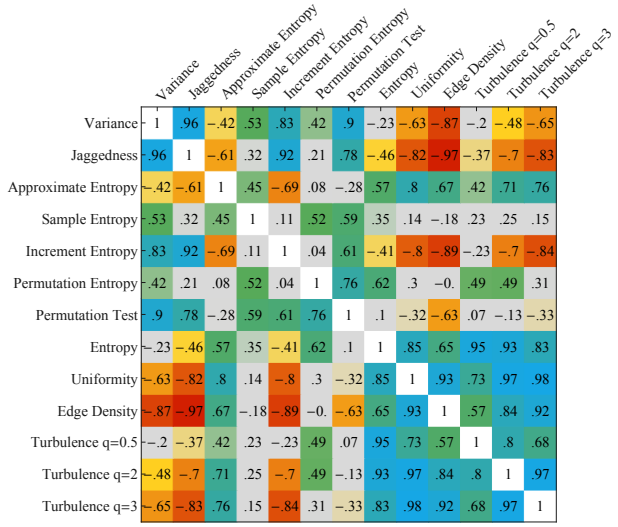
In a comparison of this sort, there is no "ground truth" to compare against: no "true" level of uncertainty that we can match each measure to. However, for the binning test we contend that the edge density measure possesses the desired scaling properties across bins: nearly monotonically decreasing with increasing numbers of bins, leveling off at high numbers of bins (saturation), and matching intuitive comparative levels among the simulated datasets. Because edge density is not sensitive to the edge weights it is not viable as a measure, and so we are not looking for a measure that is the same as edge density, but being positively correlated with edge density is a sign of capturing appropriate scaling patterns. However, a measure that is too strongly correlated with edge density reveals that it is also insufficiently sensitive to differences in the weights.

Correlations are overall lower on the low frequency sine wave than other data; however, all the RDMM measures are largely in agreement and so is ApEn. This matrix also reveals the lack of general agreement or disagreement among the non-RDMM measures on this dataset, although they are more correlated than anticorrelated (excepting ApEn and increment entropy). The basic and high frequency since waves show clear (but distinct) splits in the measures. For the basic sine wave the RDMM measures are together with ApEN and SampEn, but for for the high frequency sine wave the two permutation-based measures are also in agreement. In both cases increment entropy is less consistently anticorrelated than the others, so on deterministic data with few short-term ties all the real measures except increment entropy gets it right.

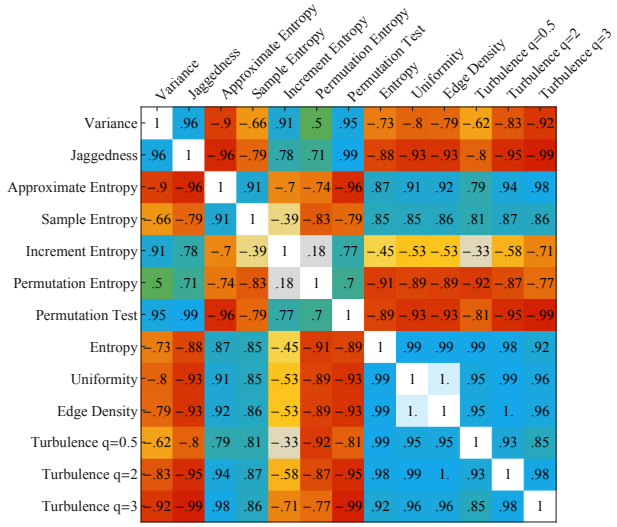
For both the random noise and noisy sine wave, all the RDMM measures are grouped together against all the non-RDMM measures. Although the RDMM measures report very different levels of uncertainty on these datasets, they all start high and decrease with increasing binning. The non-RDMM measures start high, increase up to around 25 bins, then level off. Although consecutive pairs, and even small segments, of numbers are completely random (and so we can expect high uncertainty evaluates from measures that look only at those), the overall behavior of these data streams is consistent. As we get more refined data we can say with greater confidence what distribution they are derived from. With these considerations in mind the uncertainty of the process should decrease with increasing resolution and that all the RDMM measures get this right.

The correlations among the two random walk datasets are a bit messier, but also reveal a somewhat similar patterns to each other. In both cases the edge density is correlated with uniformity and turbulence$_3$, anticorrelated with entropy and turbulence$_{0.5}$, and weakly correlated with turbulence$_2$. In fact, uniformity itself is more strongly correlated with turbulence$_2$ than edge density. On the random walk, ApEN is also correlated strongly with uniformity, but not on the random walking sine wave. On these datasets the increment and permutation-based measures are increasing with increasing resolution, and so both entropy and turbulence$_{0.5}$ follow that pattern. We have already explained above how being overly sensitive to edge weights in sparse Markov models yields this result for entropy and turbulence here. For the other measures it has more to do with how binning the data acts like a course-graining filter for the local noise; increased resolution reveals more of the tiny "vibrations" in the data to which these measures are sensitive (which also makes them correlated to jaggedness on several datasets).

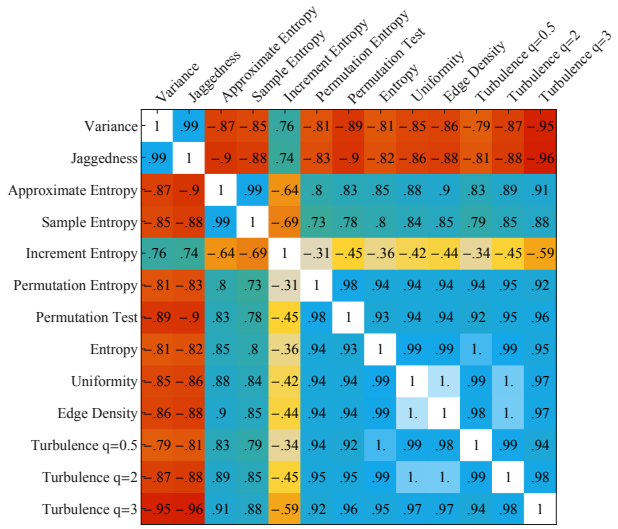
So again, on the correlation analysis, uniformity comes out ahead of all the measures (both RDMM and non-RDMM) in its consistency of correlation partners across the datasets. Specifically it is consistently correlated with edge density, but also consistently teams up with turbulence$_2$ or turbulence$_3$. Unsurprisingly, APEn and SampEn are tightly correlated, as well as the pairs of variance & jaggedness and the two permutation-based measures. Incremental entropy does its own thing, sometimes correlated variance and jaggedness, sometimes only with permutation-based measures, sometimes with all non-RDMM measures.



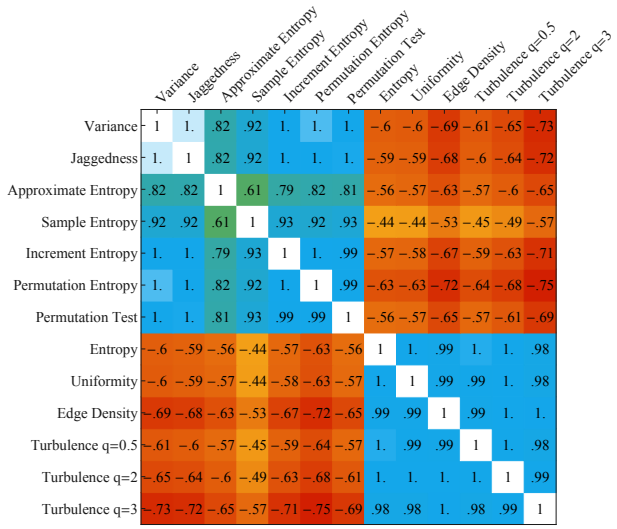

Low Frequency Sine Wave: Sin[t/(4π)]



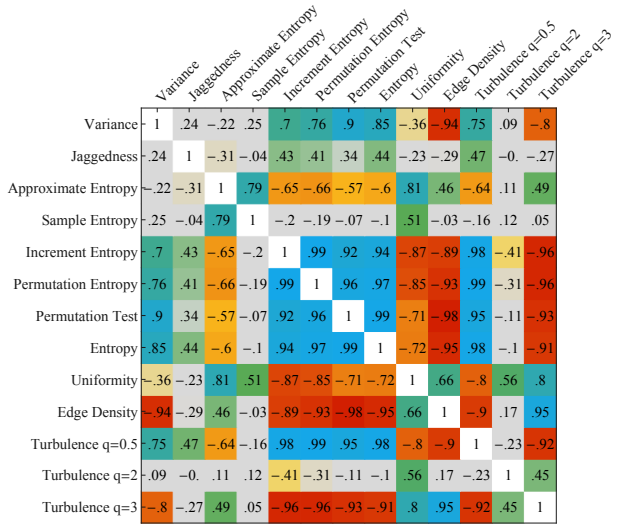

Basic Sine Wave: Sin[t/π]



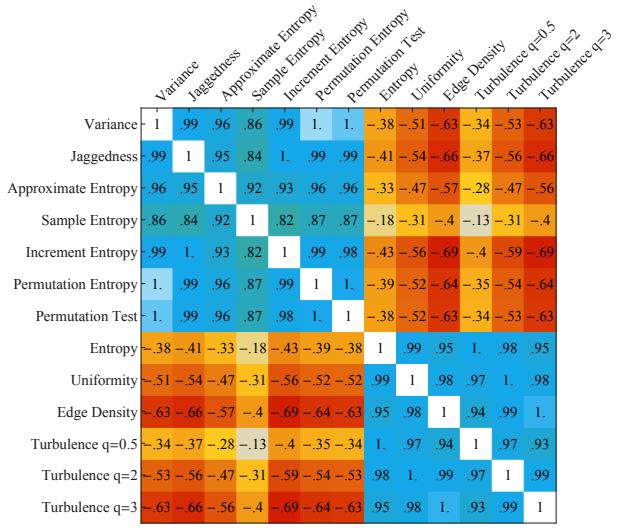

High Frequency Sine Wave: Sin[(8t)/π]



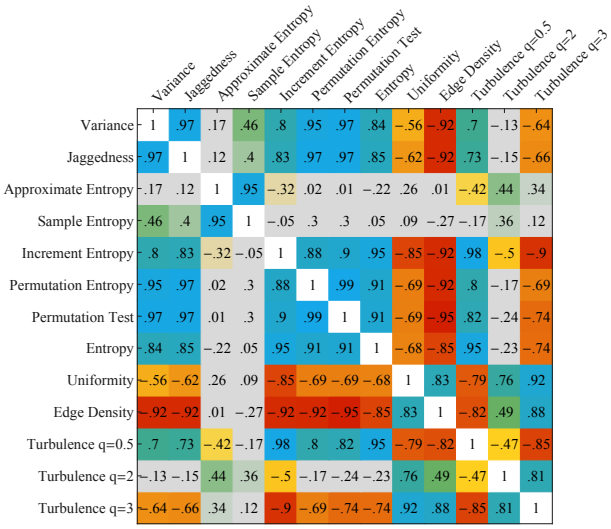

Noise: Normal[0,0.4]




Random Walk: $\sum_{i=1}^{t}$ Normal[0,0.4]




Noisy Sine Wave: Sin t/π + Normal[0,0.4]




Random Walking Sine Wave: Sin t/$\pi$ + $\sum_{i=1}^{t}$Normal[0,0.4]

## B.8 Robustness across Multiple Long Time Series

The generated time series above were limited to 1200 points because this reflects a constraint of several empirical datasets and because it suffices to demonstrate the core points. Although by no means a large number, it does exceed the recommended minimum points for the previous measures of uncertainty [32, 33, 34]. In this section we demonstrate that the features described above robustly hold for longer time series and other stochastic parameters. For all the deterministic time series, adding more points does not change the Markov model, and every "run" is identical. So here we focus on the four stochastic time series. Recall that each of the four stochastic time series utilize the same sequence of draws from a random distribution. First we describe the robustness results in terms of the same Normal distribution with mean 0 and standard deviation 0.4 used above. We also expand on this by comparing the effects on measures of uncertainty of wider standard deviations $\sigma = \{0.4, 0.8, 1.2\}$ as well as a Uniform distribution with endpoints at $\{(-0.4, 0.4), (-0.8, 0.8), (-1.2, 1.2)\}$ (see Fig. B.10).

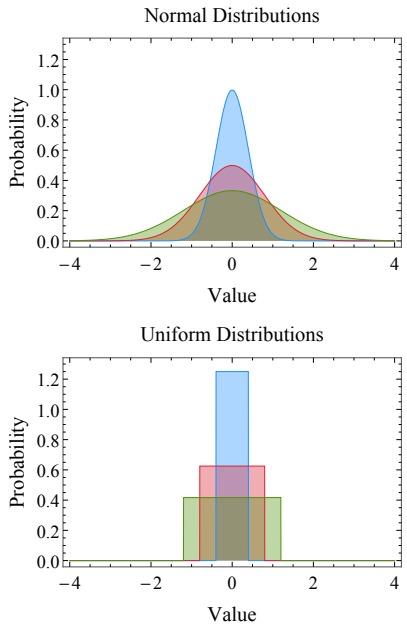



Figure B.10: Plots showing the PDFs of the six distributions used in the tests described in this section.

First we created 10,001 points for each of 100 distinct realizations of the random noise (enough to potentially achieve maximum uncertainty at 100 bins). Then, as before, we used that sequence of random numbers to build: the noise sequence (the numbers themselves), the random walk (accumulating the numbers in sequence), the noisy sine wave (adding the number to each value of the basic sine function), and the walking sine wave (accumulating the previous values at each time step). In this way we can separate both the role of the random pull among the 100 time series and the role of the sequence's pattern.

Up to this point we have already eliminated several RDMM measures as redundant or inferior, so in this analysis we focus on entropy, uniformity, and turbulence$_2$. We also include the most common and demonstrably useful non-RDMM measures: sample entropy and permutation entropy. For each stochastic time series we have a set of six plots showing the response to binning for each of the noise distributions. In addition, we show the measure values (including edge density) for each time series and distribution at 50 bins in a separate set of plots for easier comparison.

From the plots below you can see that the assessment of the uncertainty of each measure is distinct across datasets, but consistent on the same type of dynamical pattern. This is also true for sample entropy and permutation entropy. The random walkers are similar to each other, and the noisy datasets are similar to each other, but the two sets are distinguishable. For example, the walking sine wave is slightly more uncertain and less consistent in its uncertainty measurements than the random walker (although it may be difficult to tell them apart from a single noisy parameter, their patterns across noise dispersions is clearly distinguishable). The point of this section is only to show that the results above do not depend on the time series length nor on any particular draw from the random distribution. That is why we focused on relative comparisons instead of specific numbers: the relative patterns are consistent across datasets of similar dynamical types even if there is variation in the specific values.

First we examine all the plots for the random noise (in the first set below). The first and most salient observation is that all the measures are more precise on the uniform noise than the normally distributed noise. This is because a Normal distribution has the possibility to produce outliers, and these outliers greatly effect both (1) this kind of regular-interval binning and (2) the appearance of observations unique to a particular time series. The second observation is that both permutation entropy and sample entropy report nearly identical values across both the distribution type and width; i.e., neither measure is sensitive to either change (on the noise series). Contrarily, the RDMM measures are all much greater for the uniform noise than the normal noise. This is because the normal distribution produces a pattern in the exit transitions of the induced Markov model, with more edges and stronger weights in the center and fewer/weaker at the periphery, but the uniform distribution (obviously) produces a more uniform Markov model. **This is exactly the sense of uncertainty we are aiming to measure with this technique: the existence of a pattern in the time series that yields information about the generating mechanism (even when that**

**generating mechanisms is merely a stochastic process). The ability of of the RDMM measures, and the inability of the non-RDMM measures, to capture this difference is a solid win for the proposed technique.**

This difference shows up in different ways for the other generated datasets. For simple noise, the normalization of the range to the observed data diffuses the effect that the distribution's dispersion parameter could have on the RDMM measures. For example, at 50 bins edge density reaches nearly 1 for all domain widths of the uniform distribution because 10,001 points is enough to reveal nearly all possible transitions among 50 nodes regardless of the values of the PDF's domain — and even turbulence$_2$ has a value near 0.9. But on the noisy sine wave there is tradeoff between the underlying sine pattern and the noise, so a wider noise dispersion creates a more noticeable (and measurable) effect compared to the underlying sine wave. Here again, although the normal distribution has a wider domain, the uniform distribution yields greater uncertainty for the RDMM measures, but not so for sample and permutation entropy.

For the random walk, none of the measures report systemically higher or lower uncertainty values across increasing parameter values or between normal and uniform distributions (although the edge density and uniformity results are more precise for the uniform distribution). Because there is no underlying pattern in the random walk, the resulting overall pattern of accumulated noise doesn't vary (systematically) depending on the type of noise (after normalization). Random walks are random walks, and their characteristic behavior does not depend on the particular kind of (symmetric) random noise distribution.

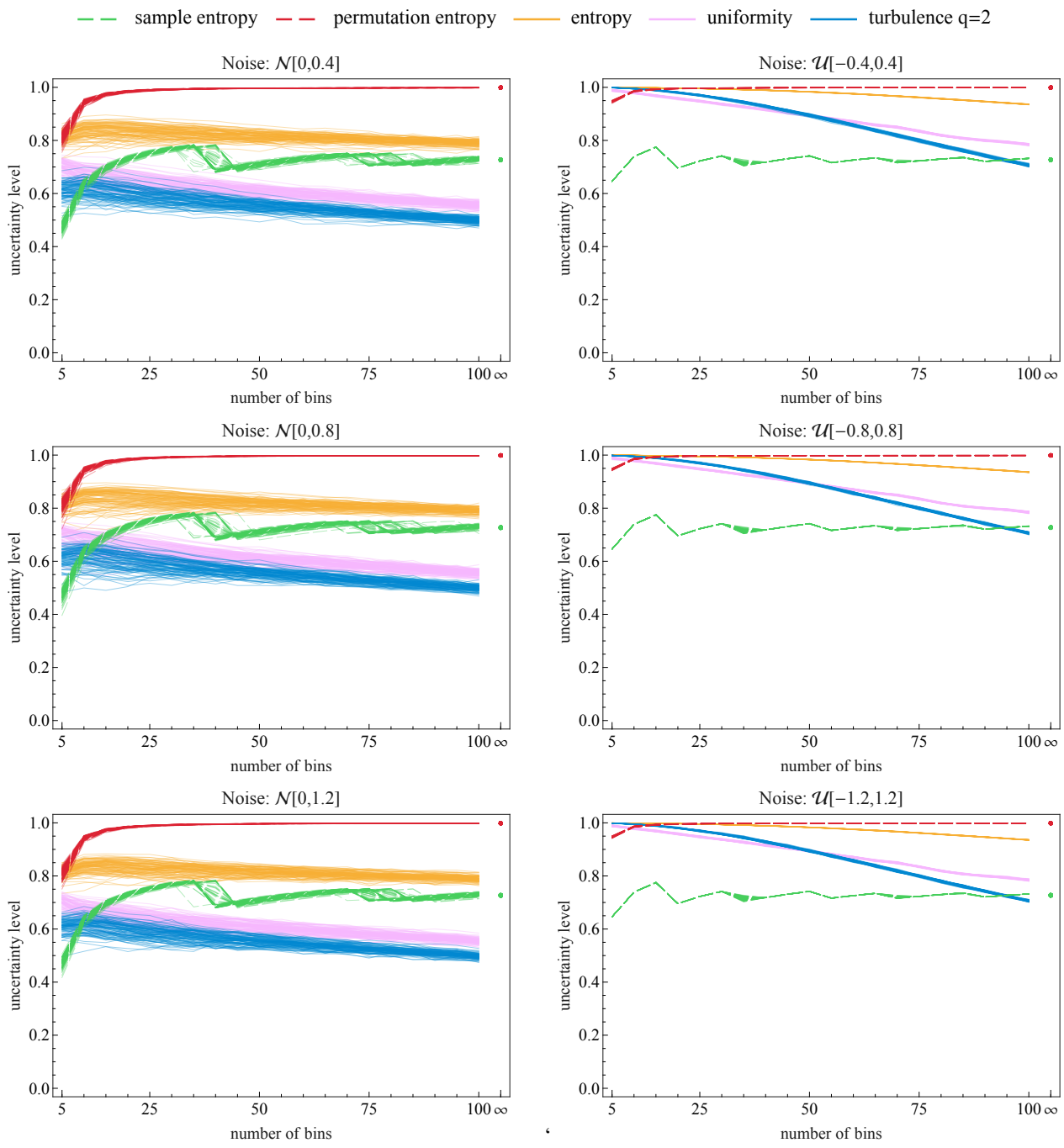
The random walking sine wave provides an interesting mix of the random walk and noisy sine wave. In this case increasing the dispersion of the noise has the effect of *reducing* the uncertainty for both normal and uniform distributions. Recall from the time series figure above (of one example of the random walking sine wave) that the effect of the normal[0,0.4] is small compared to the sine wave, so there is a visible sine wave moving up and down based on the bias provided by accumulating the noise. The larger the noise, the less it looks like a sine wave and the more it looks like a random walk, and random walks have lower uncertainty than noisy sine waves, so the net effect of increased noise is reduced uncertainty.

#### B.8.1 Value Depression through Normalization.

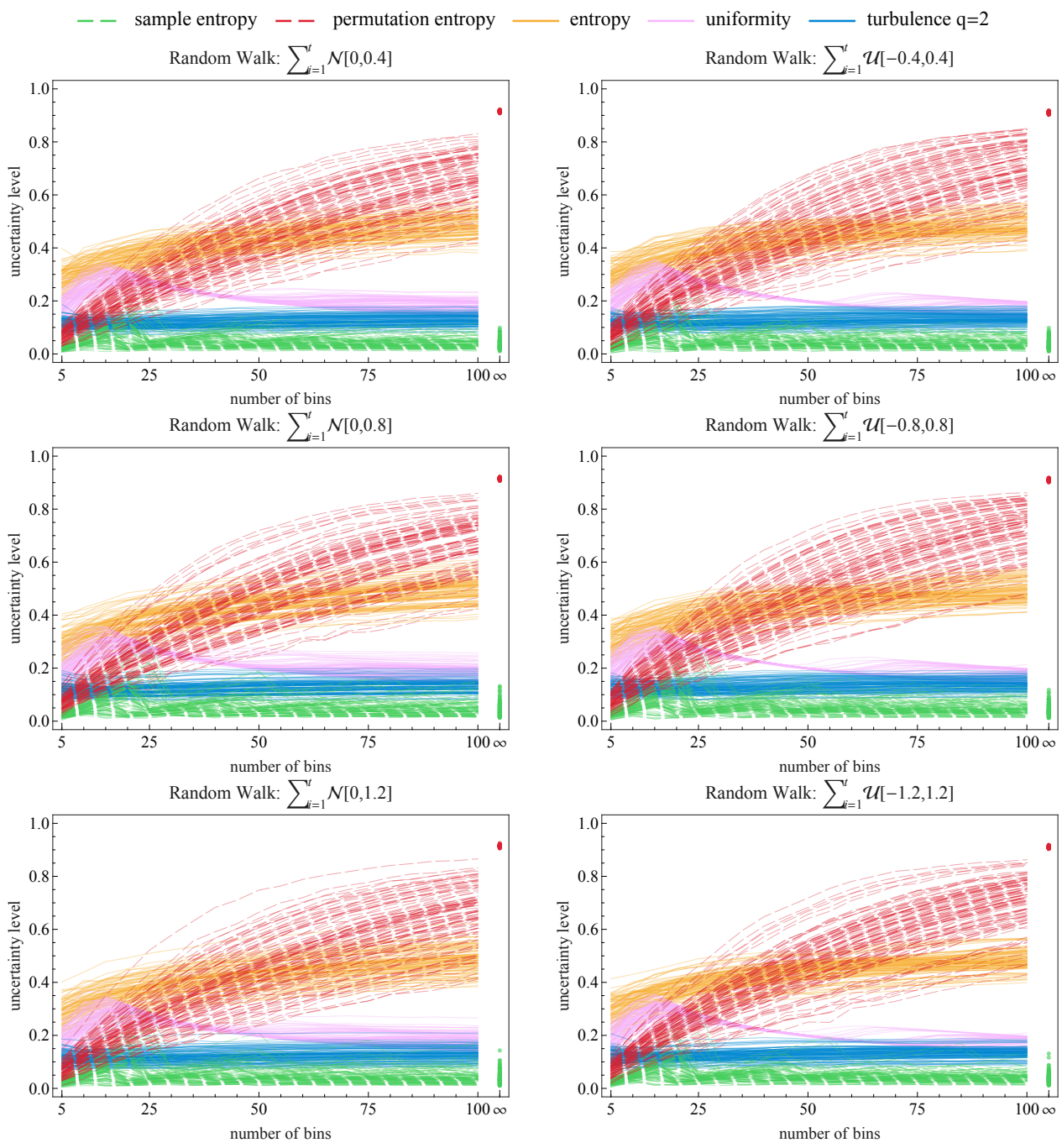
One feature worth mentioning is that, since we chose enough points to reach maximum uncertainty at 100 bins, the values here are less depressed at higher bins for this reason. Compared to the results of single runs seen above we can indeed see that the uncertainty values are greater at the higher end of the bin spectrum in most cases (i.e., less depression where more points are needed), but the effect is small enough that it doesn't affect the *ranking* of the measures.

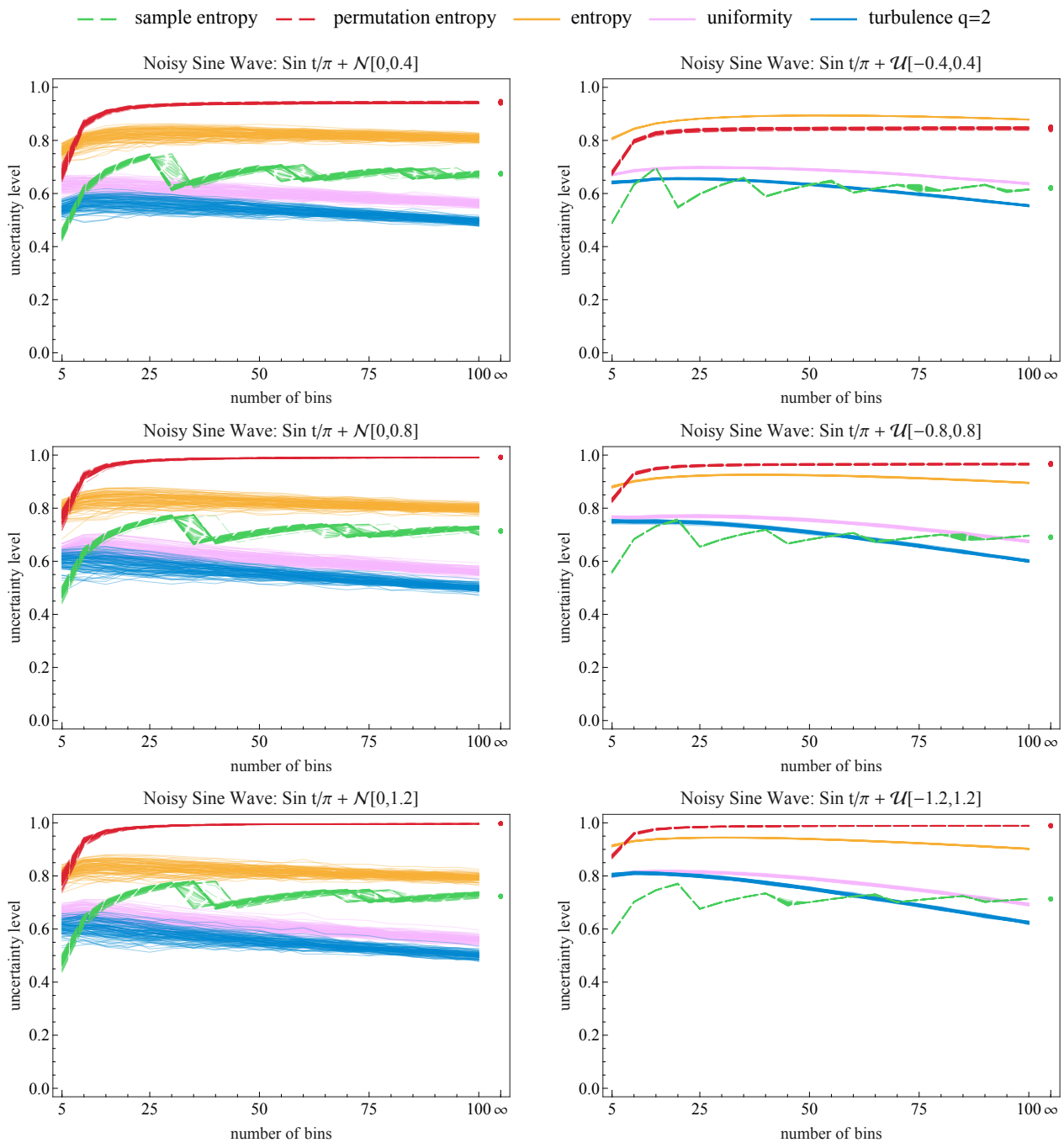
#### B.8.2 Unbinned non-RDMM measures.

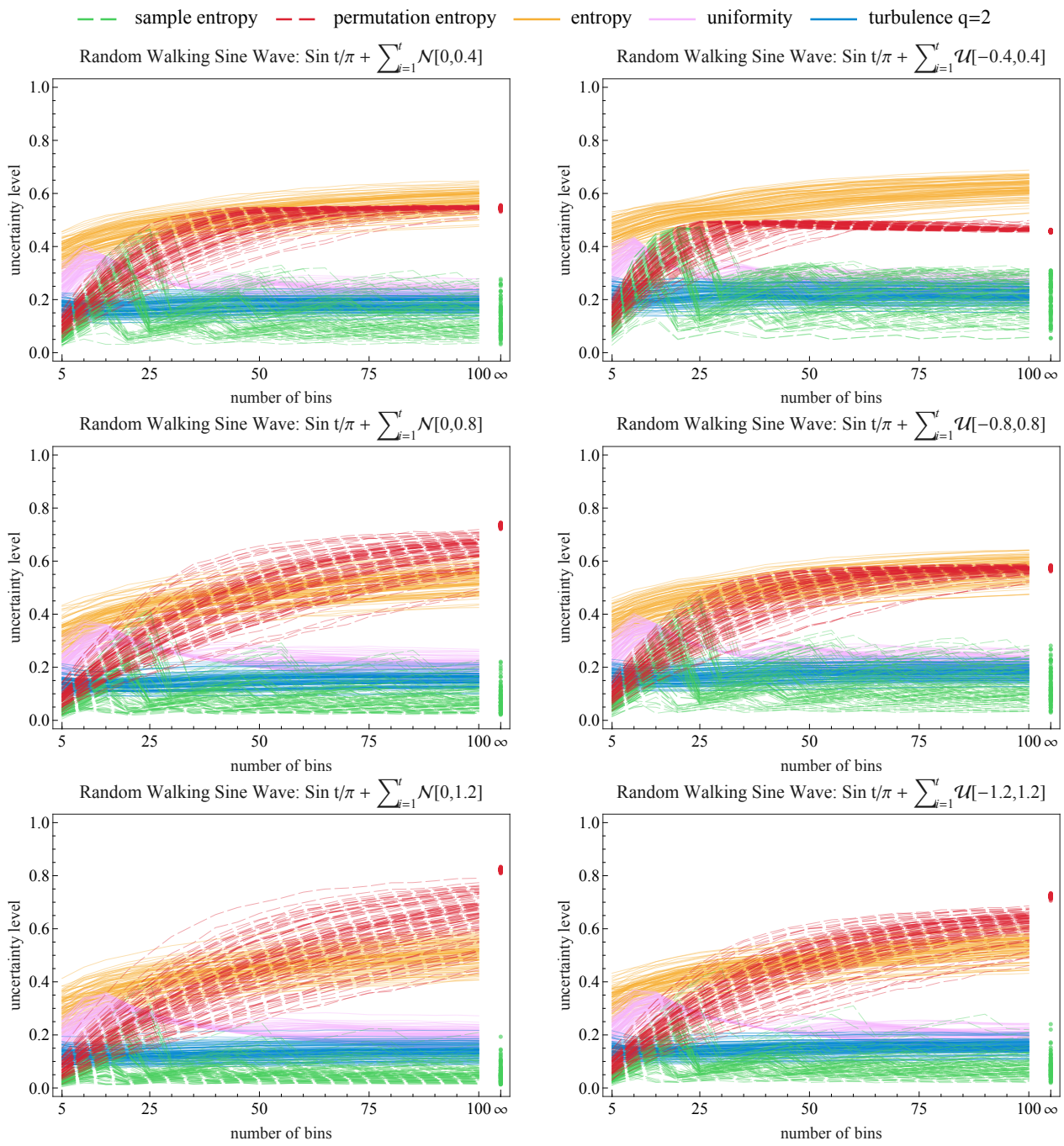
We can also do a similar comparison using the unbinned versions of the non-RDMM measures. Sample Entropy is already convergent to its unbinned value by 50 bins on these runs, so it is unaffected. Permutation quickly converges on the noise (total uncertainty) and noisy sine wave (lesser uncertainty), but not on the random walks. On the noisy sine wave and walking sine wave, the greater the width of the distribution, the greater the effect in comparison to the sine wave's underlying pattern, and thus the greater the uncertainty (of this type). On the random walk, permutation entropy is less than the random noise, but it is unaffected by the parameter of the stochastic process. As per usual, it often takes many bins for the permutation entropy to converge to its unbinned values because the presence of ties deflates the reported uncertainty, but on the noise and noisy sine wave datasets the permutation entropy plateaus around 50 bins.

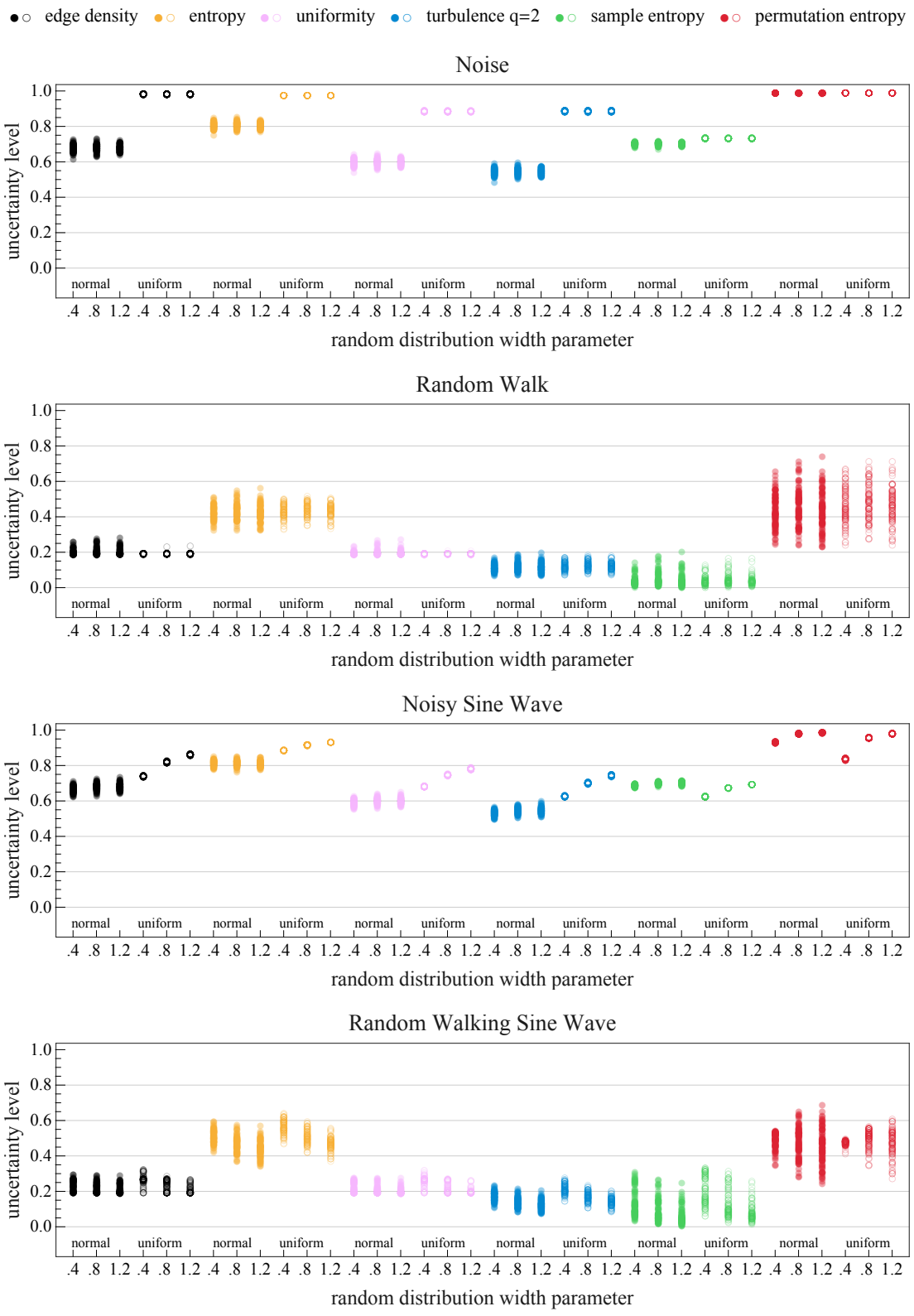

sample entropy
permutation entropy
entropy
uniformity
turbulence q=2
Noise: $\mathcal{N}[0,0.4]$
Noise: $\mathcal{U}[-0.4,0.4]$
Noise: $\mathcal{N}[0,0.8]$
Noise: $\mathcal{U}[-0.8,0.8]$
Noise: $\mathcal{N}[0,1.2]$
Noise: $\mathcal{U}[-1.2,1.2]$
uncertainty level
number of bins
0.0
0.2
0.4
0.6
0.8
1.0
5
25
50
75
100
∞

‘


sample entropy
permutation entropy
entropy
uniformity
turbulence q=2
Random Walk: $\sum_{i=1}^{t} \mathcal{N}[0,0.4]$
Random Walk: $\sum_{i=1}^{t} \mathcal{U}[-0.4,0.4]$
Random Walk: $\sum_{i=1}^{t} \mathcal{N}[0,0.8]$
Random Walk: $\sum_{i=1}^{t} \mathcal{U}[-0.8,0.8]$
Random Walk: $\sum_{i=1}^{t} \mathcal{N}[0,1.2]$
Random Walk: $\sum_{i=1}^{t} \mathcal{U}[-1.2,1.2]$
uncertainty level
number of bins
0.0
0.2
0.4
0.6
0.8
1.0
5
25
50
75
100
∞


sample entropy
permutation entropy
entropy
uniformity
turbulence q=2
Noisy Sine Wave: Sin t/π + 𝒩[0,0.4]
Noisy Sine Wave: Sin t/π + 𝒰[−0.4,0.4]
Noisy Sine Wave: Sin t/π + 𝒩[0,0.8]
Noisy Sine Wave: Sin t/π + 𝒰[−0.8,0.8]
Noisy Sine Wave: Sin t/π + 𝒩[0,1.2]
Noisy Sine Wave: Sin t/π + 𝒰[−1.2,1.2]
uncertainty level
number of bins
0.0
0.2
0.4
0.6
0.8
1.0
5
25
50
75
100
∞


sample entropy
permutation entropy
entropy
uniformity
turbulence q=2
Random Walking Sine Wave: Sin t/π + $\sum_{i=1}^{t} \mathcal{N}[0,0.4]$
Random Walking Sine Wave: Sin t/π + $\sum_{i=1}^{t} \mathcal{U}[-0.4,0.4]$
Random Walking Sine Wave: Sin t/π + $\sum_{i=1}^{t} \mathcal{N}[0,0.8]$
Random Walking Sine Wave: Sin t/π + $\sum_{i=1}^{t} \mathcal{U}[-0.8,0.8]$
Random Walking Sine Wave: Sin t/π + $\sum_{i=1}^{t} \mathcal{N}[0,1.2]$
Random Walking Sine Wave: Sin t/π + $\sum_{i=1}^{t} \mathcal{U}[-1.2,1.2]$
uncertainty level
number of bins
0.0
0.2
0.4
0.6
0.8
1.0
5
25
50
75
100
∞


edge density
entropy
uniformity
turbulence q=2
sample entropy
permutation entropy
Noise
Random Walk
Noisy Sine Wave
Random Walking Sine Wave
uncertainty level
0.0
0.2
0.4
0.6
0.8
1.0
normal
uniform
.4 .8 1.2
random distribution width parameter

## B.9 Summary of Simulated Data Analysis

Although $\text{turbulence}_2$ comes out ahead overall on our tests, it is not a Pareto superior option. The low levels of uncertainty reported for the random walks, especially at small numbers of bins, implies that this measure (as well as edge density and $\text{turbulence}_2$ or $\text{turbulence}_3$) cannot pick up on the randomness of the walk. In truth, what this reveals is that this kind of accumulative randomness produces stepwise changes that are small compared to the overall changes, and when considered as a Markov process the next value is well predicted by the previous value. Entropy and $\text{turbulence}_{0.5}$ get this wrong because increasing the number of bins increases the number of edges cutting across each bin (many of which were previous lumped together as a self-transitions), and these measures are extra sensitive to changes in the distribution of edge weights when the number of edges is small.

Another worry about the RDMM measures is that getting more data can increase the measured uncertainty levels. Consider the random noise dataset: with 1200 data point and 34 bins, the uniformity is just below 0.35. We know that the unweighted edge density will go to 1 as the length of the time series grows. The noise is normally distributed, so even at the limit we know that the distribution among bin-to-bin transitions would never be uniform, but uniformity (and turbulence) would likely increase if we kept the same bins and doubled the length of the series...though only up to the point at which the model captures the uncertainty of a normal (versus uniform) distribution.

One intuition is that, as the number of data points increases, our information about the system increases, and so all else being equal our uncertainty regarding the system should go down (not up). However, a longer random sequence contains more idiosyncratic transitions, so it would require more information to recreate it (in the sense of Kolmogorov complexity), and that is why uncertainty goes up with more data in this case. Increasing uncertainty with increasing bins is a separate issue, even though they are both "getting more information about the system", the former is getting more data points while the latter is getting more detailed information on the same data points. So uncertainty can certainly go up in the former case, but it should typically decrease in the latter case (excepting some spacial cases where specific value boundaries inordinately increase cross-boundary transitions).

Another way to think of this is that the RDMM method only captures/measures the data actually seen and does not try to generalize to the generating process. That feature is by design. For coherent patterns, getting more data will reveal rare events and strengthen (in terms of greater probabilities) the core behaviors. With more data we become more confident that the Markov model captures the behavior, and uncertainty goes down as edge weights become focused on those repeated transitions. However, in the case of random noise, getting more data simply increases our certainty that the signal is noise. With a short noisy sequence there may be an underlying pattern that cannot be discerned, but the longer the sequence gets, the more uncertain we are (and the more confident we are that we are uncertain) about the reproducibility/predictability of that sequence.

Although we tested the sensitivity of the measures between $5-100$ bins, we have also shown that the maximum number of bins that can support the maximum level of uncertainty on one-dimensional data is $B_m = \lfloor \sqrt[2]{T-1} \rfloor$. For this simulated data that number is 34, but we also tested the robustness and correlation claims on multiple, longer time series of 10,0001 points and found consistent results when $B_m = 100$. This robustness is a desirable feature as long as it doesn't reflect a lack of sensitivity. The technique being proposed never breaks with too little or too much data, but the amount of data regulates how successfully the model captures the data and how reliable the measures are. Future work will pair the uncertainty assessments with confidence levels of the uncertainty that reflect both the number and distribution of the transition observations.

# C Weather Data and Analysis Details

Weather prediction is clearly important, and hidden Markov models (HMM) have long been useful for this task [35, 36, 37, 38]. The point of the HMM approach is that there is an underlying (hidden) causal mechanism with parameters that can be reverse-engineered from the observed data and then used to make near-term predictions. Our technique is not being proposed for making predictions, it instead describes the observed dynamics in a way that facilitates new measures of system uncertainty. As mentioned above, one way to think about it is that uncertainty measures on the RDMM inform us of how complicated the HMM (or other model) would need to be to capture the dynamics in a certain weather system.

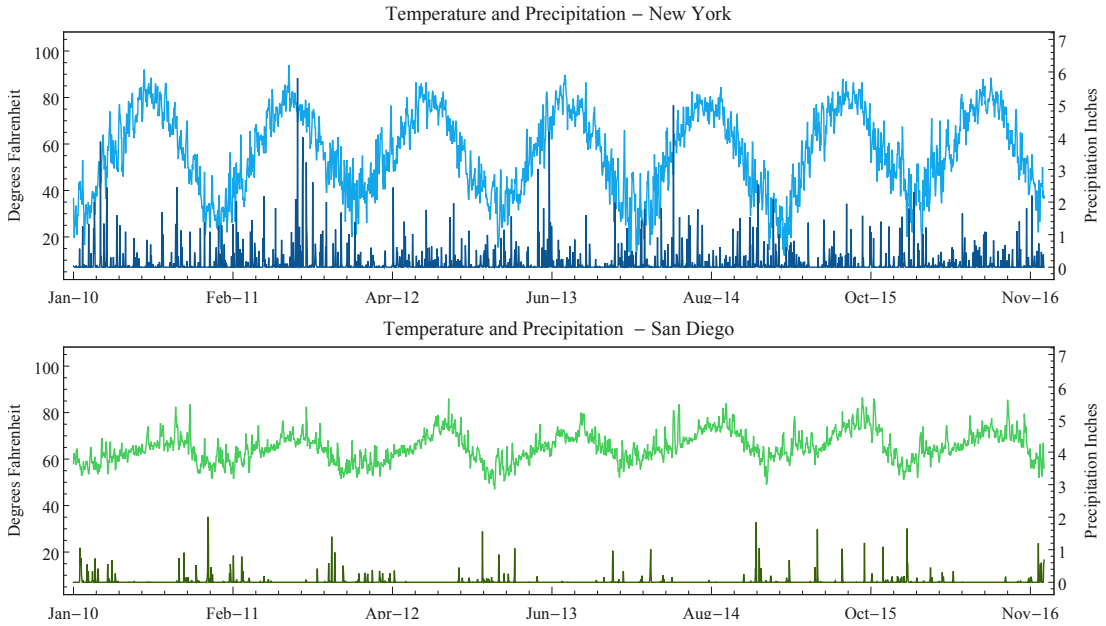
We use our RDMM technique to analyze daily precipitation and temperature data for four US cities with prototypical relative levels of uncertainty:

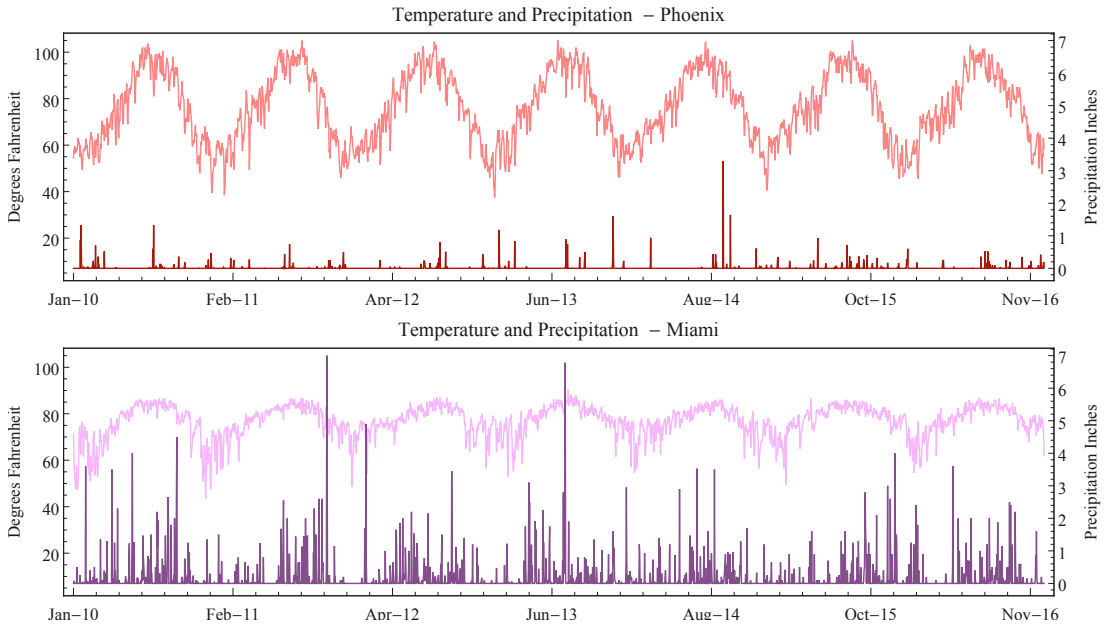
- New York City Central Park, NY
- San Diego International Airport, CA
- Phoenix Airport, AZ
- Miami Beach, FL

The data comes from the NOAA Climate Data Online: Dataset Discovery page [39]. We requested data on temperature and precipitation from Jan 1, 2010 to Dec 31, 2016 for a total length of 2,557 days. Because "average temperature" and "observation temperature" were not available for half of this date range, our temperature time series is the mean of the minimum and maximum for each day. In both time series there are a few small gaps (1-3 days of missing data) which we filled via linear interpolation.

These cities were chosen because of their distinct mixes of temperature and precipitation patterns. For those not familiar with the weather of these cities, you can see from the plots below that New York and Phoenix have similar temperature variations, but very different precipitation levels. Miami and and San Diego likewise have similar temperatures and very different rain patterns. However San Diego and Phoenix have similar rain patterns, as do New York and Miami. Thus these four cities combined occupy each square of a $2 \times 2$ grid of low and high temperature fluctuations and precipitation amounts.

Because of these relationships we can establish an intuitive ranking of the uncertainty of weather in these cities. In order from most predictable to most uncertain are (1) San Diego, (2) Phoenix/Miami, and (3) New York. Phoenix and Miami tie for second in our intuitive ranking because the two cities are trading off variation in one time series with the other; the ranking will depend on whether precipitation or temperature is more uncertain. Below we test how each measure performs in matching our expectation, as well as sensitivity to binning. Note also that volatility has been included in these analyses.






## C.1 Shared Binning

The binning here is done differently than before. Because we want to compare the uncertainty of the cities' weather on equal grounds, we use a shared binning across all four cities. So, even though the temperature in San Diego never gets above 90°F or below 40°F, it is still binned using the same range of values as New York and Phoenix. This will naturally create fewer filled bins for the San Diego dataset, but this alone does not automatically translate into less uncertainty for RDMMs. In order to make the different time series comparable with the shared binning they must be normalized by a shared value as well; we naturally choose the number of bins $B$ as the common normalization factor.

Note that this shared binning and normalization does not require that the RDMM contain nodes for the empty bins. We still gain the computational speed and memory advantages from excluding empty bins from the model and analysis. The normalization step operates on the measurement values output by applying the uncertainty measures equations on the abridged RDMM.

For RDMM entropy the shared normalization is a straightforward replacement of $n$ with $B$. The renormalization of edge density and turbulence to shared bins requires changing $1/n$ in the above formations to $n/B^2$. Thus generalized turbulence becomes

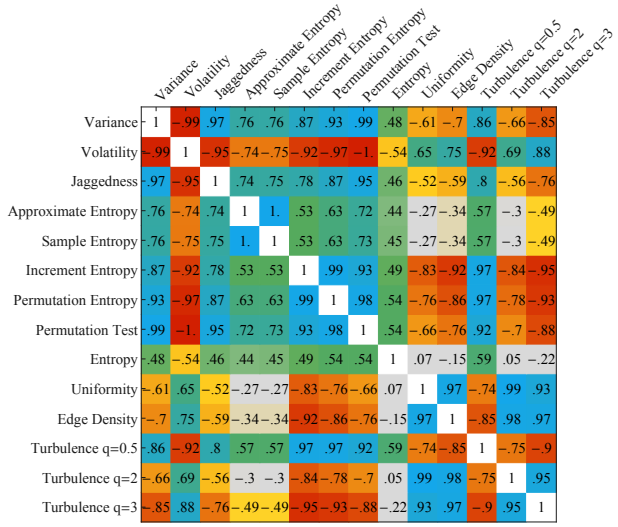
$$\frac{\frac{1}{B^2}\sum_{s_i}\kappa_i - \frac{n}{B^2}}{\left(\frac{n}{B^2}\right)^{2-q} - \frac{n}{B^2}} \tag{22}$$

to capture the fact that the minimum value is still achieved when there are $n$ edges all with weight 1. And for uniformity it is necessary to convert the $n \times n$ adjacency matrix into a $B \times B$ one by padding the difference with identity matrix entries to ensure the minimal uncertainty case is still minimally valued.

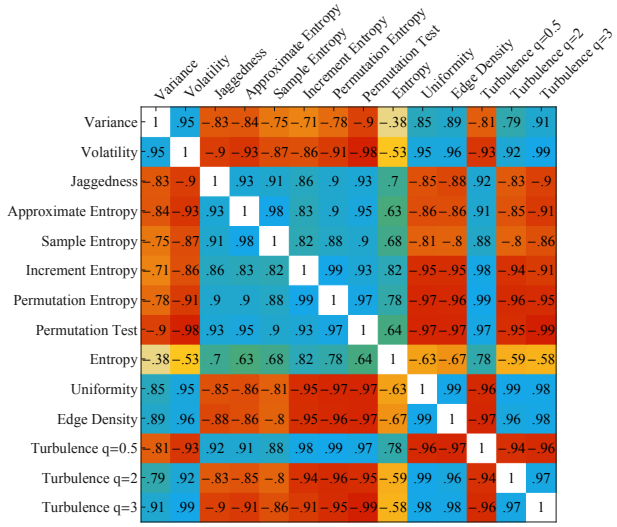
Without these shared bin adjustments the uncertainty of the temperature cities' time series may become indistinguishable because the standard RDMM measures are scale-invariant by design. But with shared bins and renormalized measures the values for each city become comparable. This requirement for shared bins applies to any situation in which the uncertainty levels being compared come from different instantiations of the same kind of system. For now, whether they should and/or can be used is evaluated on a case-by-case basis.

## C.2 Weather Data Correlations

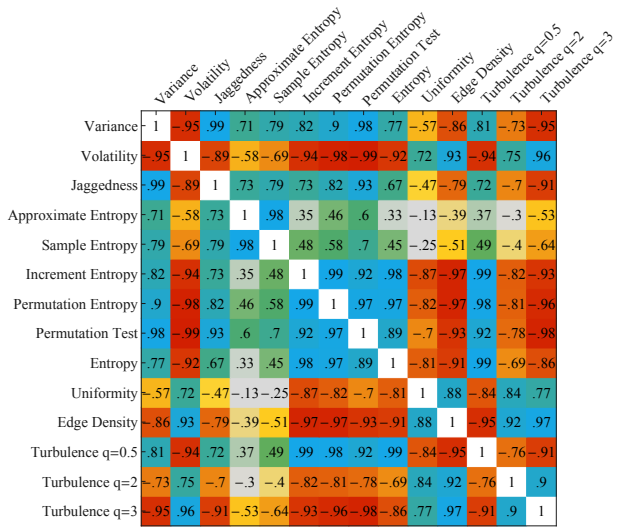
One will surely notice the similarity of each of the temperature time series to the noisy sine wave simulated data; annual cycles with daily noise. Despite this general visual similarity, the measures values are more similar to the plain random walk pattern. Recall that the noisy sine wave produced a clean split between the RDMM and non-RDMM measures. Looking at the correlation matrices below, you will see that all cities share a similar correlation pattern, and overall it is more similar to the random walkers (and closest to the pure random walk) than the noisy sine wave. For one, entropy and turbulence$_{0.5}$ join the permutation-based measures, but also the correlations among the remaining RDMM measures is weaker. Interestingly, volatility (which has not been previously included due to its mathematical restrictions) is highly correlated with the concave RDMM measures including unweighted edge density and turbulence$_2$.



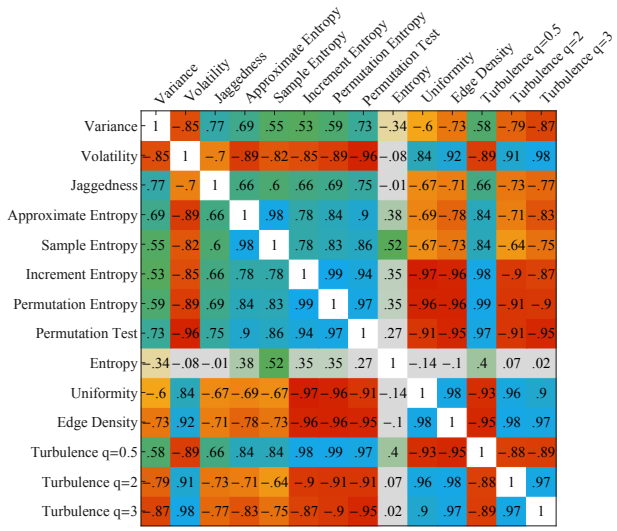

Temperature – New York




Temperature – San Diego



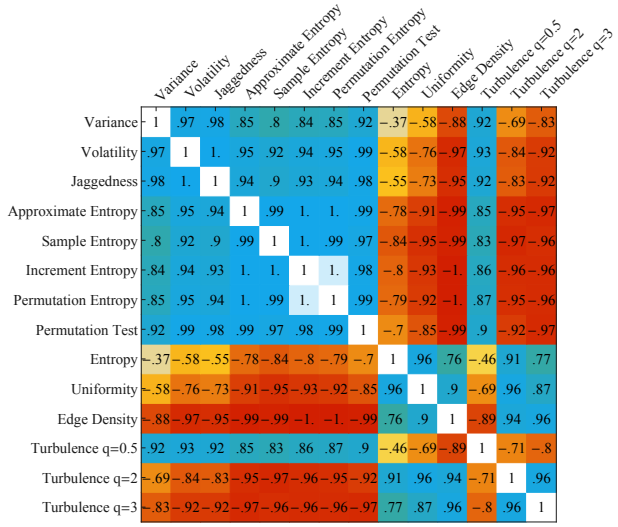

Temperature – Phoenix



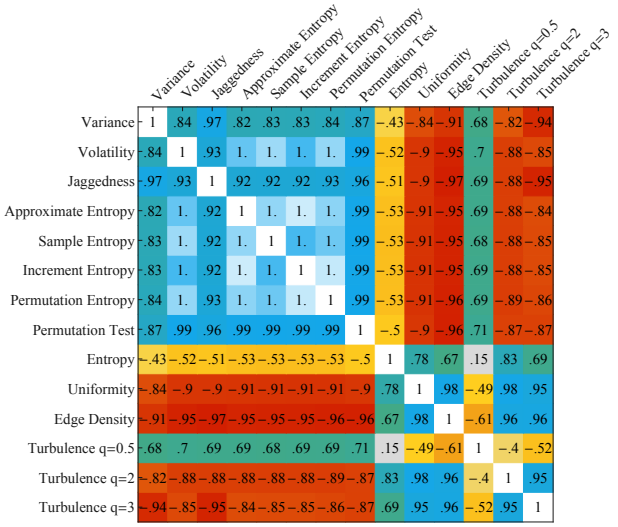

Temperature – Miami

The correlation matrices for the precipitation (mostly rain) data tell a different story. The temperature data for each city had different ranges, but they were all one-year periodic time series with significant daily noise. Rain data for these cities is less seasonally dependent and less varied from day to day. Overall these correlation patterns (a mostly clean split between RDMM and non-RDMM measures) are most similar to random noise and the noisy sine wave. One clear exception is that turbulence$_{0.5}$ is inversely correlated except for Phoenix, and edge density is instead inversely correlated for Phoenix. This is also the only case in which RDMM entropy and turbulence$_{0.5}$ are so weakly and/or negatively correlated.

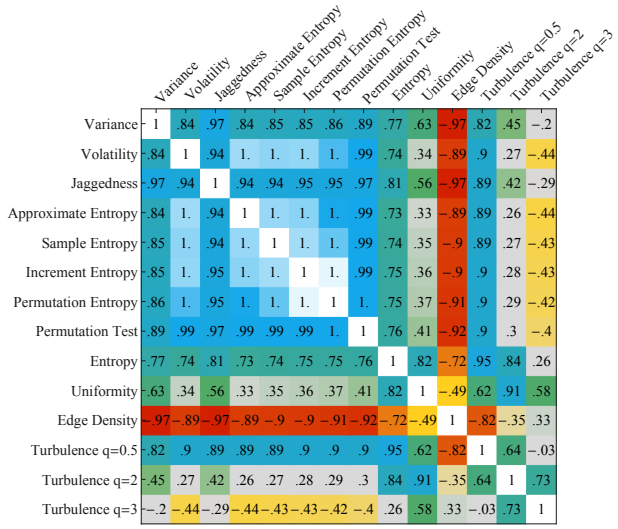
Because I did not generate data like this rain data (e.g., from a Poisson process), it is not surprising for the pattern to be distinct. More importantly for our purposes here, the fact that the RDMM measures (and turbulence$_2$ in particular) measure the uncertainty in a way that is not tracked consistently by any of the previous non-RDMM measures suffices to demonstrate *the RDMM measures comprise a distinct method for determining the uncertainty of time series*. That does not by itself mean that it is better method or that they should be preferred; it is too early to judge such a thing. However, this does establish independence and hence the possibility to convey new information about time series uncertainty.



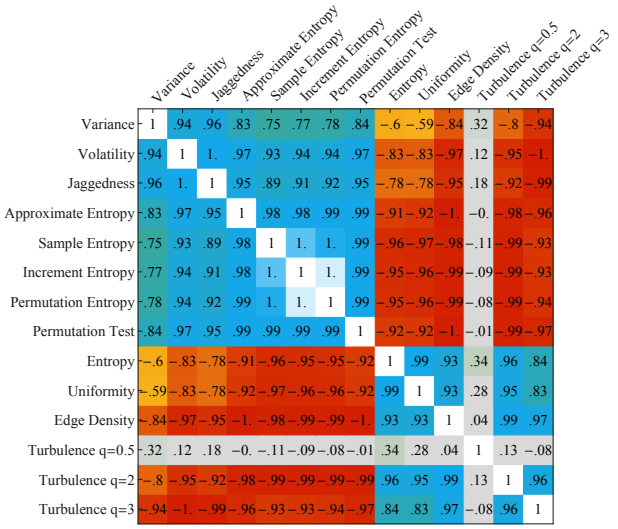

Rain – New York




Rain – San Diego




Rain – Phoenix




Rain – Miami

## C.3 Weather Data Measure Values

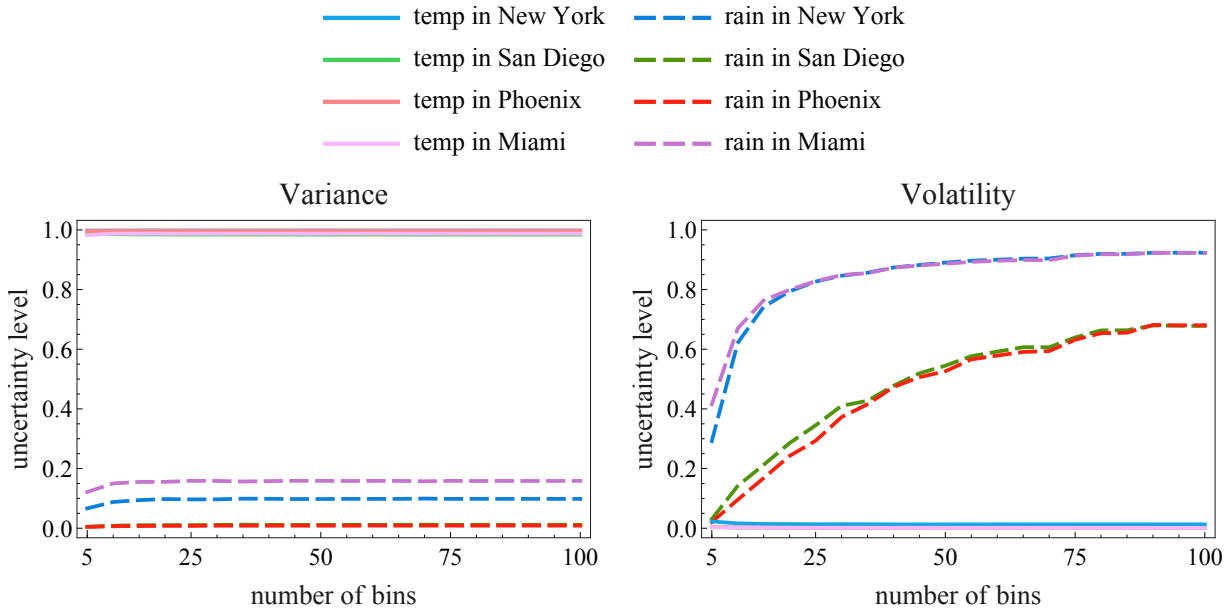
Here we have plots for each measure across numbers of bins for all eight datasets analyzed here (temp and precipitation in four cities). For reference in interpreting the following plots, with 2,557 data points the max number of bins that can generate max uncertainty $B_m$ is 50, and we provide the full sweep from 5 to 100 bins in order to show sensitivity to binning.

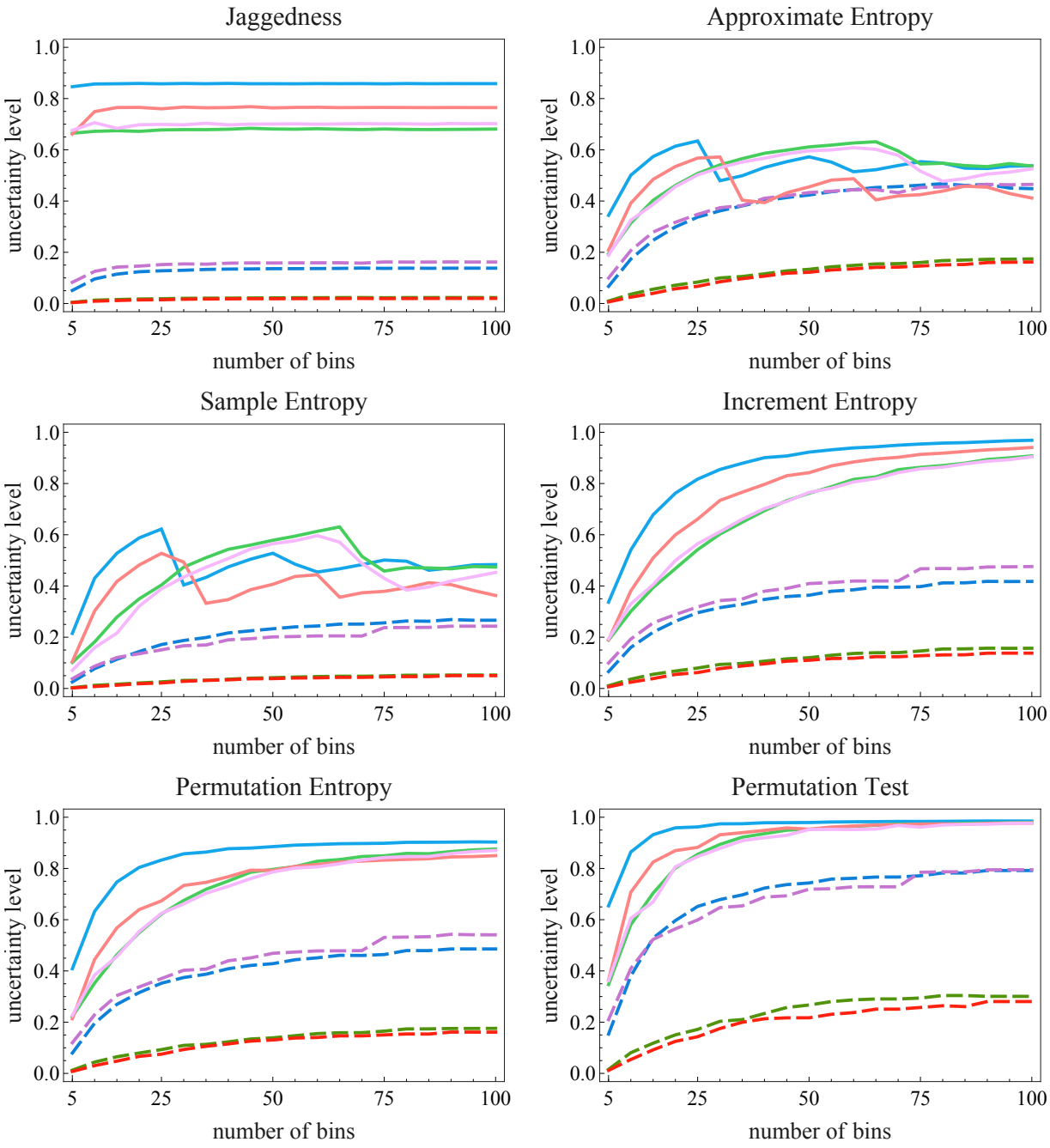
The first immediately recognizable pattern is that, except for volatility and a few crossover events at lower bin values, temperature is rated as more uncertain than the rain. It is clear from the data plots above that the temperatures have greater variance and jaggedness. Furthermore, rain tends to return to the lowest bin value (i.e., zero precipitation) in between short periods of higher values. Rain patterns may be perceived/measured as less uncertain because they more often occur in bursts of increasing then decreasing with (overall) fewer wild fluctuations on a day-by-day basis.

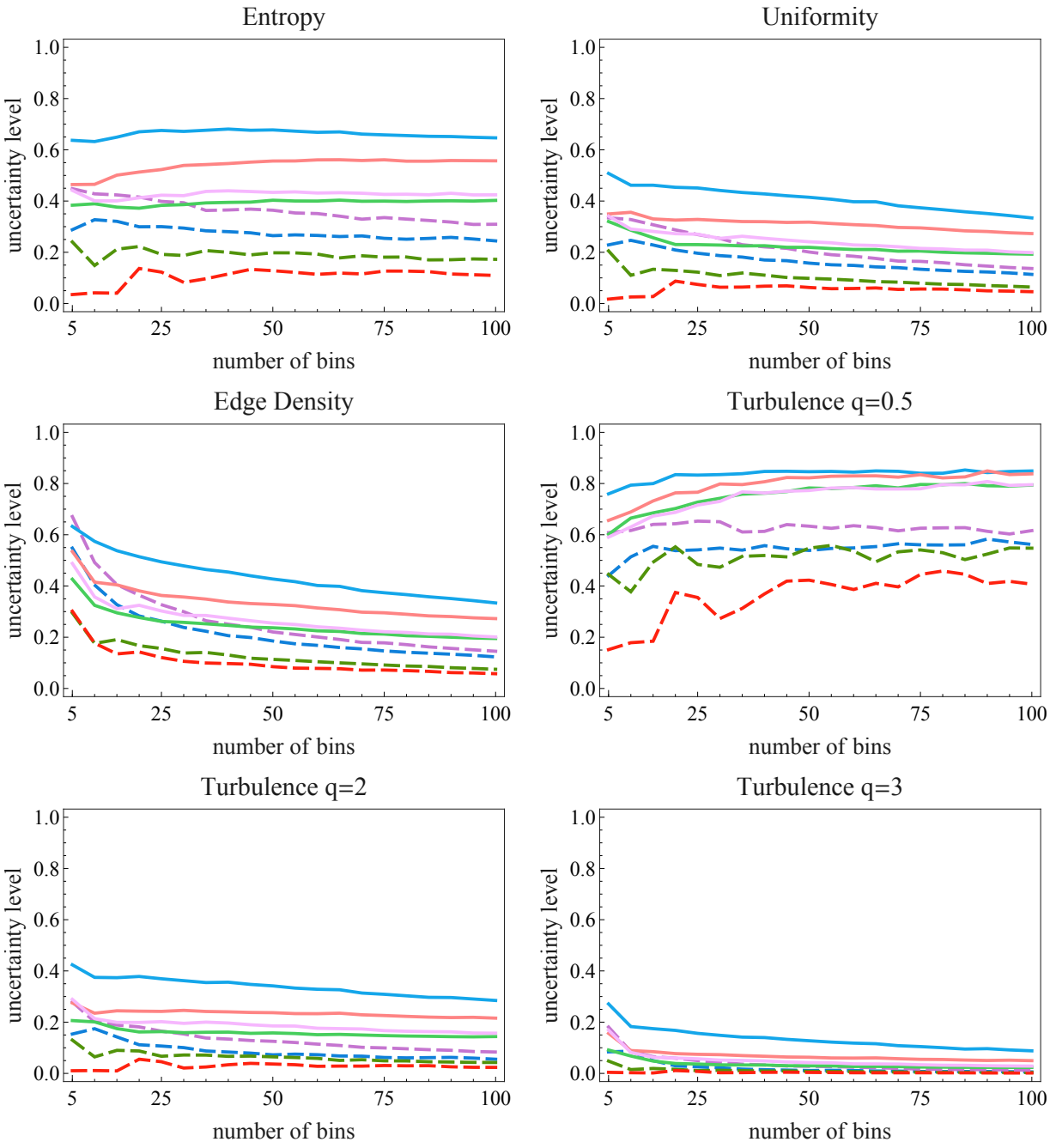
San Diego and Phoenix have less rain, and intuitively they have less uncertain rain patterns because of how often the rain level is close to zero. The non-RDMM measures consistently rate San Diego and Phoenix rain as similar, and New York and Miami rain as similar (and greater) in uncertainty. In contrast, the RDMM measures consistently rank them as Miami, New York, San Diego, Phoenix; presumably this is because there is less focus on short-term variation in the specifics of the series and more focus on the range of values which can follow from each value.

The ranking of temperates is intuitively correct for the RDMM measures, but not for most of the non-RDMM measures (c.f., Increment Entropy captures this well). One thing to note is that the temperature in New York is consistently more uncertain that Phoenix according to basically all measures; a results that coincides with it having both a wider range of values and higher daily variation. Some features can be captured by all measures, while other features are captured by a subset of the measures.

Otherwise, the patterns here are basically as we have come to expect. Variance and jaggedness are mostly insensitive to the binning, while ApEn and SampEn are overly sensitive (although only for the temperature). All the non-RDMM measures are increasing with greater resolution, which again highlights their focus on short-term variability and point-predictability rather than an assessment of larger-scale patterns in the data. Entropy and turbulence$_0$.5 are more tame on these datasets, showing roughly the same pattern as the other RDMM measures, but exacerbated upward.




Jaggedness
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
uncertainty level
number of bins
0.0
0.2
0.4
0.6
0.8
1.0
5
25
50
75
100




Recall again that the nodes in an RDMM represent only values observed in the dataset. San Diego never gets more than 2 inches of rain in a day, whereas New York gets up to 6 and everywhere in between. As a result of this difference, San Diego's rain (and temperature) RDMMs have roughly half as many nodes as the New York RDMMs when using the same bins (see Table C.1 below of percents of bins used). Recall that these are global bins; e.g., the range between the maximum and minimum values across *all* the rain data cute into $B$ bins. And because they are all normalized by the same $B$, having fewer nonempty bins does not necessarily translate into lower uncertainty. It depends on the pattern of transitions among the bins used.

**Weather Data Combined Model.** We can also look at the weather system behavior through a combined model incorporating both rain and temperature into a single RDMM. The temperature and precipitation are not clearly correlated (although rain does seem seasonal in San Diego and Miami, and therefore should be at least somewhat correlated with temperature), so each data stream provides different information about the uncertainty of the cities'

| | Number of Bins | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 | 60 | 65 | 70 | 75 | 80 | 85 | 90 | 95 | 100 |
| Temp in New York | 1 | 0.9 | 0.933 | 0.9 | 0.92 | 0.9 | 0.914 | 0.9 | 0.889 | 0.9 | 0.891 | 0.883 | 0.877 | 0.871 | 0.88 | 0.875 | 0.847 | 0.867 | 0.853 | 0.85 |
| Temp in San Diego | 0.6 | 0.5 | 0.467 | 0.45 | 0.44 | 0.433 | 0.429 | 0.425 | 0.422 | 0.42 | 0.418 | 0.417 | 0.415 | 0.414 | 0.4 | 0.4 | 0.4 | 0.411 | 0.411 | 0.41 |
| Temp in Phoenix | 0.8 | 0.7 | 0.733 | 0.7 | 0.72 | 0.7 | 0.714 | 0.7 | 0.689 | 0.7 | 0.691 | 0.7 | 0.692 | 0.7 | 0.693 | 0.7 | 0.694 | 0.667 | 0.684 | 0.68 |
| Temp in Miami | 0.8 | 0.6 | 0.533 | 0.55 | 0.52 | 0.5 | 0.486 | 0.5 | 0.489 | 0.48 | 0.473 | 0.467 | 0.477 | 0.471 | 0.467 | 0.45 | 0.447 | 0.444 | 0.453 | 0.45 |
| Rain in New York | 1 | 0.8 | 0.667 | 0.65 | 0.64 | 0.6 | 0.6 | 0.55 | 0.556 | 0.54 | 0.527 | 0.517 | 0.492 | 0.486 | 0.467 | 0.45 | 0.459 | 0.433 | 0.432 | 0.43 |
| Rain in San Diego | 0.4 | 0.3 | 0.333 | 0.3 | 0.32 | 0.3 | 0.314 | 0.3 | 0.267 | 0.26 | 0.255 | 0.267 | 0.277 | 0.257 | 0.24 | 0.25 | 0.247 | 0.233 | 0.221 | 0.22 |
| Rain in Phoenix | 0.6 | 0.4 | 0.333 | 0.3 | 0.28 | 0.3 | 0.286 | 0.25 | 0.244 | 0.22 | 0.218 | 0.233 | 0.215 | 0.214 | 0.2 | 0.188 | 0.188 | 0.2 | 0.189 | 0.19 |
| Rain in Miami | 1 | 0.9 | 0.8 | 0.75 | 0.68 | 0.667 | 0.657 | 0.65 | 0.622 | 0.62 | 0.6 | 0.583 | 0.569 | 0.557 | 0.56 | 0.538 | 0.529 | 0.533 | 0.526 | 0.51 |

Table C.1: Percent of bins used for each number of bins for each dataset.

weather. We hold off on this analysis here (mostly in consideration of space), but we plan to use the combined model as a demonstration for a follow-up analysis incorporating time windowed uncertainty assessments (to capture uncertainty variations by season). We still demonstrate the combined time series RDMM method using exchange rate data in the next section. Because we anticipate the time-windowed versions of this technique will have broad application, and comparing each of the time windows requires the same shared-binning modifications to the methods and measures, we introduce these modifications here for future elaboration.

**Summary of Weather Data Analysis.** The main point of this weather analysis is the shared binning and re-normalization required to fairly compare different datasets of the same type. This is clearly necessary to compare the temperatures in different cities because if we rescale San Diego and New York to the minimum and maximum of their respective ranges, then they will have similar *rescaled* time series and hence similar uncertainty. That is acceptable when we are merely interested in the independent measurements, but for most practical applications we expect San Diego to have less uncertain weather patterns than New York...and we do if and only if we use a shared binning. However, note that there is no benefit to using the same binning for temperature and precipitation — quite the opposite. When making comparisons of data of different kinds, using their [min, max] range is often the best, but it is important to determine the appropriate binning for each one in order to ensure a fair comparison on a case-by-case basis.

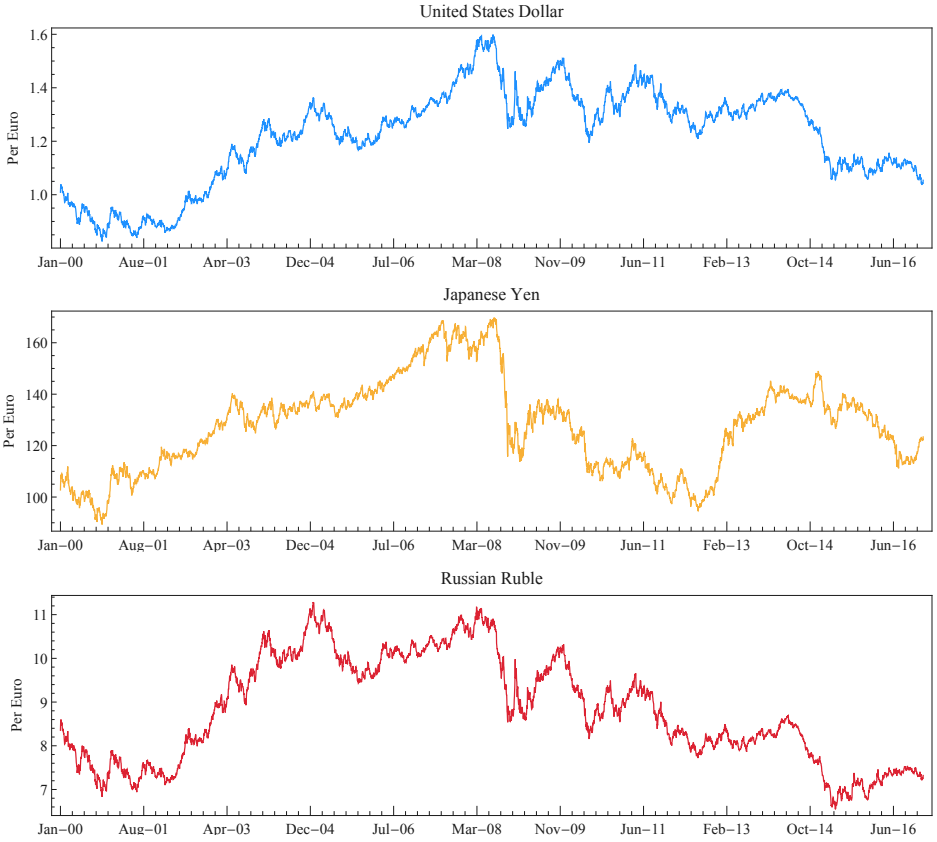
One result of this weather data analysis is to reaffirm that the RDMM technique reveals uncertainty information that is distinct from previous measures. It also confirms that the RDMM measures are not merely tracking the variance, volatility, or jaggedness of the time series. The patterns among the RDMM measures here reinforce the notion that the square root of turbulence$_2$ is the most suitable for empirical examinations. Although similar in scale and sensitivity to some other RDMM measures on these datasets, turbulence$_2$ shows the greatest consistency across binnings and correct relative values of uncertainty.

The intended application of these uncertainty measures is to multichannel physiological data: e.g., EEG, ECoG, cardiac etc. are sensor arrays with multiple signals. In order to compare the intensity/frequency of signals on the same basis we need shared bins and appropriate re-normalizations. Other augmentations are also likely necessary for specific applications (such as Markov models of changes in values and/or two-step Revealed Dynamics Markov Models), but at minimum we presented this material in order to introduce and demonstrate the required re-normalizations for various shared-bin analyses.

# D Exchange Rate Data and Analysis

We analyze the exchange of the Euro against three other currencies: United States Dollar, Japanese Yen, and Russian Ruble. These references rates were collected from 01/01/2000 to 31/12/2016 from the Banca D'Italia website [40]. Taking account of markets being closed on weekends and holidays there are 4351 data points in each time series.

There are no static, anchoring values in the currency market, only values to each other, and in this case relative to the Euro. Naturally the value (i.e., the purchasing power) of the Euro also varied during this period, as can be seen in the shared variation in the time series plots below. It is possible to standardize the time series to filter out the shared variations (e.g., by subtracting each by the mean value across the time series at each time step, or subtracting each series by its mean and dividing by its standard deviation), but for our current purposes it is acceptable to provide a Euro-centric analysis because it suffices to compare the *relative uncertainty* of the three referenced currencies to each other. That is, the relative uncertainty measurements are invariant to these kinds of manipulations, and thus the proposed technique minimizes the preprocessing required.




Although the similarity in dynamics implies that there is some shared influence (the Euro's value) driving the behavior of these time series, they are individually very similar to the random walking datasets analyzed above. We will see this similarity through many analyses below, starting with the percent of bins used. The yen always uses 100% of the bins, while the dollar and ruble come extremely close (minimum usage is 96.8%). This means that all three datasets are dense within their range, which results from accumulating small daily changes rather than large jumps (just like the random walks).

## D.1 Exchange Rate Measure Values by Measure

This discussion in this section references the plots below in which the US dollar is blue, the yen is golden, and the ruble is red (matching the data plots above).

Although the unit of measure for each time series is its value in Euros, they occupy distinct ranges of that scale because their relative values to each other are quite different. The Dollar ranges from 0.8 to 1.6 Euros, the Yen

| | Number of Bins | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 | 60 | 65 | 70 | 75 | 80 | 85 | 90 | 95 | 100 |
| United States Dollar | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.988 | 1 | 0.989 | 1 | 0.99 |
| Japanese Yen | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Russian Ruble | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.985 | 0.971 | 0.973 | 0.988 | 0.988 | 0.978 | 0.968 | 0.97 |

Table D.1: Proportion of bins used for each number of bins for each exchange rate dataset.

from 85 to 170 Euros, and the Ruble from 6.5 to 12 Euros. So in this case a comparison is fairer when distinct (rather than shared) sets of bins are used. As mentioned above, we could standardize the data before binning, and use shared binning on the standardized data, but this won't affect the relative uncertainty values because the standardization procedure puts each time series on its own basis in a way similar to using independent bins does.

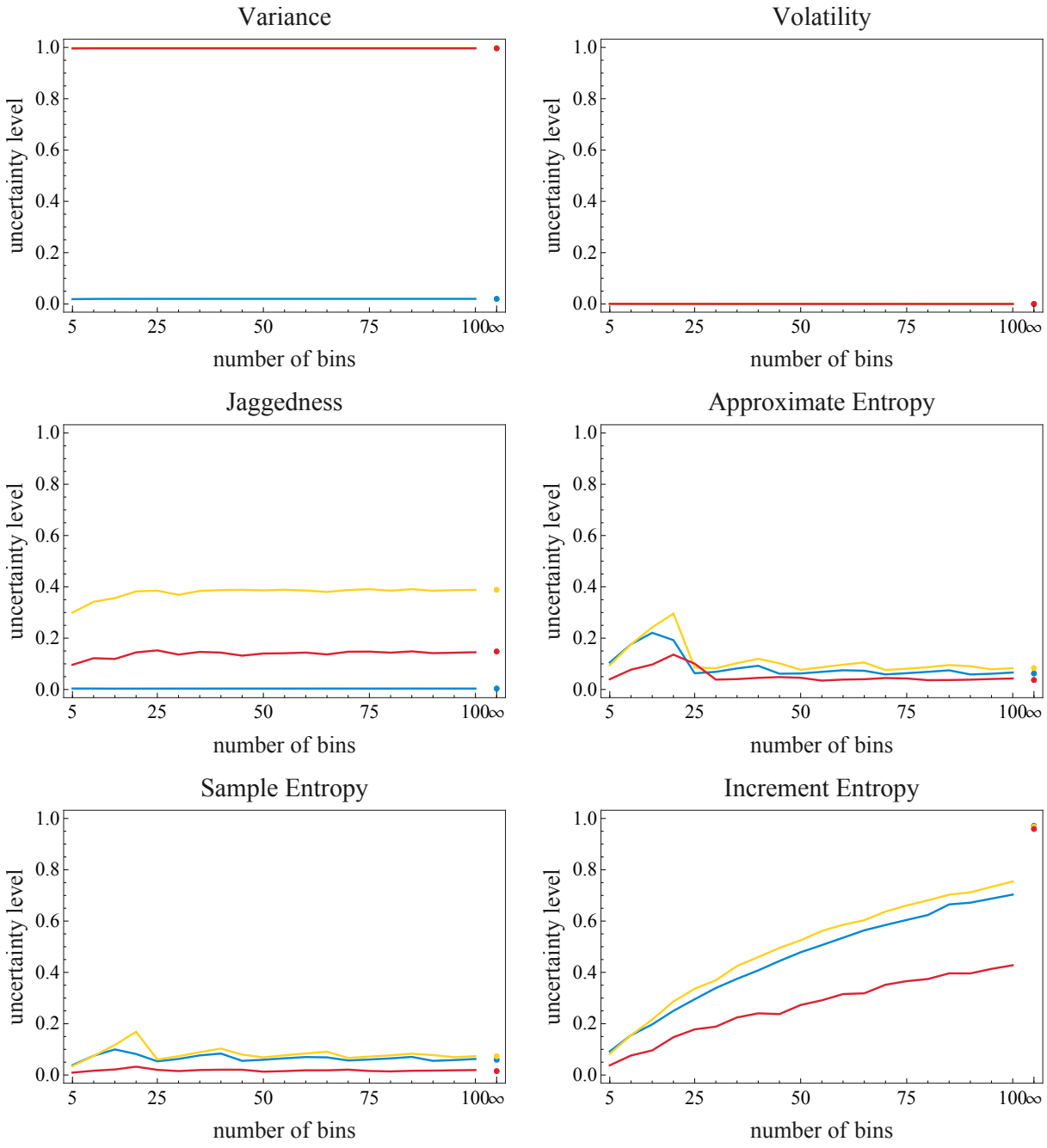
Like the weather and generated series, there is an intuitive proper ranking of the uncertainty of these three exchange rate series. Because these series values are in Euros, a natural supposition is that the Ruble depends (covaries) the most on the Euro and the Yen depends (covaries) the least. So the Ruble should be the least uncertain when captured as its value in Euros. From an ocular analysis they seem roughly similar in their short-range and long-range patterns, and so we should expect similar uncertainty profiles for all three series across all binnings. If we find deviations from the Yen ¿ Dollar ¿ Ruble ordering then that reveals information about the system behavior that is not obvious, not *a priori*, and not captured by other measures.

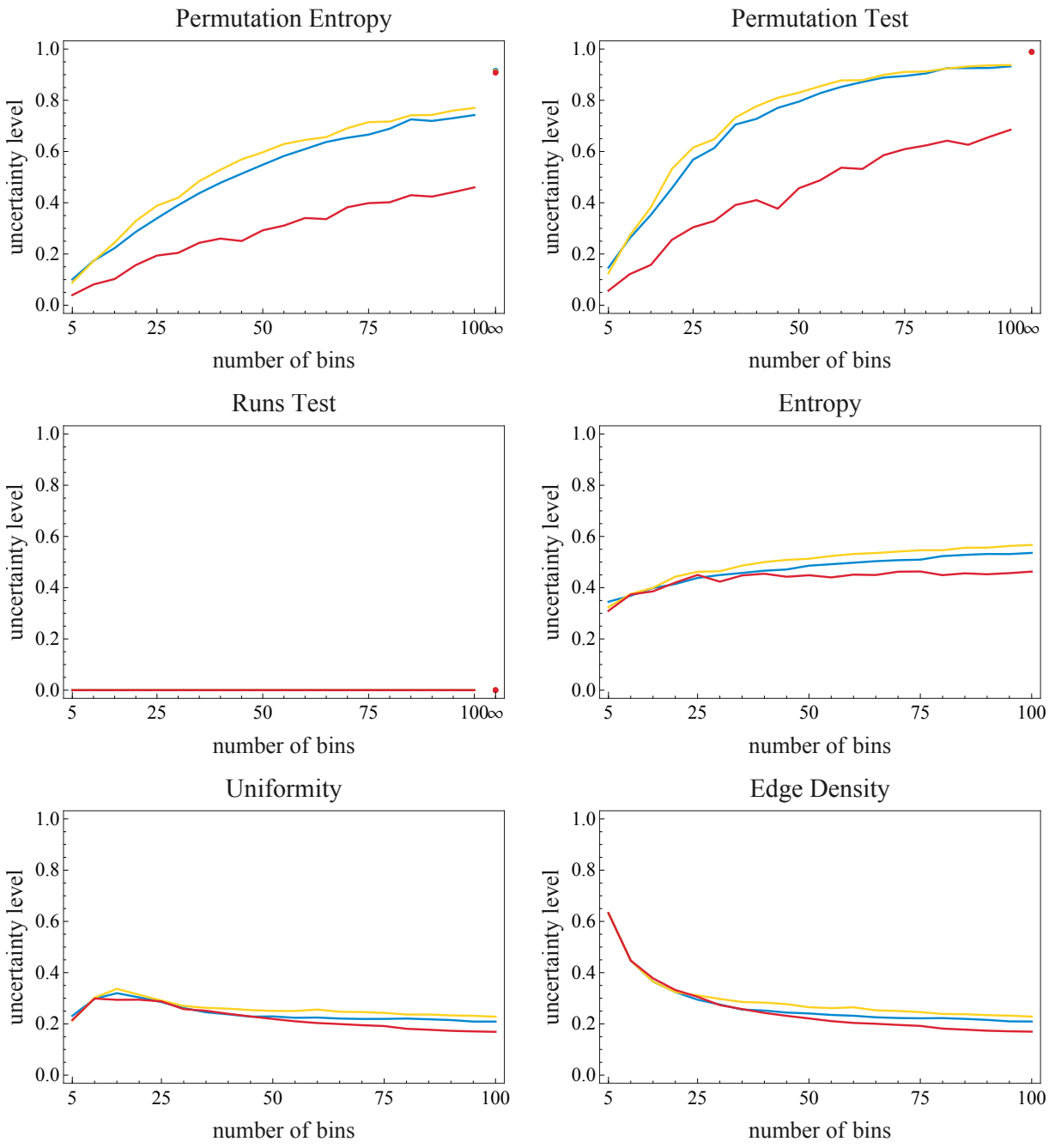
Each time series has 4,351 data points, so the maximum number of bins capable of yielding maximum uncertainty ($B_m$) here is 65, although we again sweep from 5 to 100 bins in increments of 5 to show sensitivity of the measures to the data resolution. As described in detail in the following paragraphs, the Yen follows trajectories very close to the random walks on all measures, and all three currencies are similar to each other in a fairly consistent pattern: the Yen is most uncertain, followed by the Dollar and then the Ruble. So all the (real) measures pass this face validity check of getting the ordering correct and there are no surprises in the relative dependencies on the Euro.

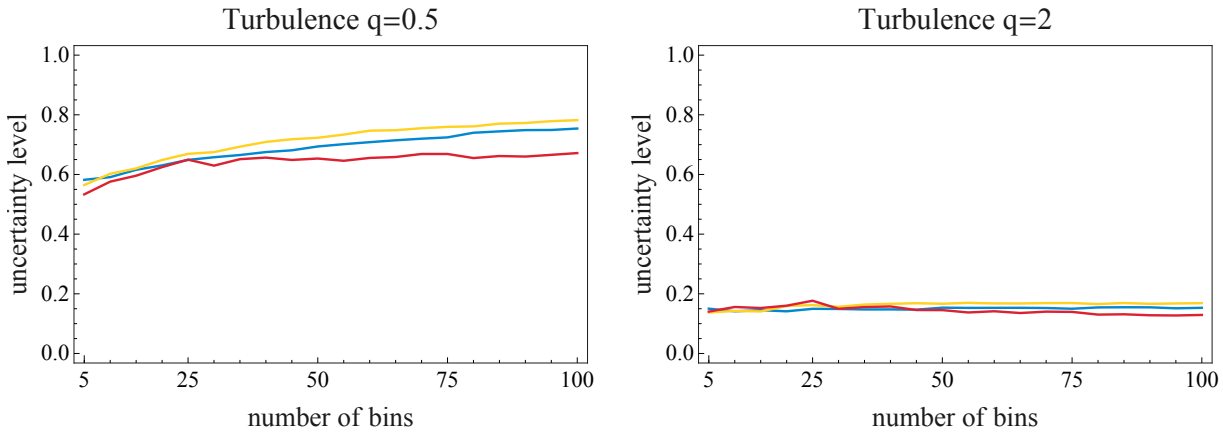
Variance and volatility are uninformative on these data, as is the runs test. The jaggedness of the yen is similar to the random walking sine, the ruble is similar to the basic sine wave, and the dollar is most similar to the low frequency sine wave. Recall that as a measure of monotonicity this implies that, whether up or down, lower jaggedness means more consecutive values in the same direction. Notice that increasing the resolution has little effect on the measurement of jaggedness because it reaches its unbinned value quickly. However, it is not so clear that Jaggedness is doing a good job of capturing how jagged the series is because, there doesn't seem to be such a big difference in monotonicity between the Yen and the Dollar, and the Dollar doesn't seem to be minimally non-monotonic (I've checked the result, it is correct, but maybe indicates a deeper issue with the measure or the normalization).

Approximate and sample entropy produce the same sawtooth pattern they usually above, and they report similar uncertainty for these exchange rates as the random walk (but much less than the walking sine wave). One of the biggest differences between this exchange rate data and the simulated data is the fact that many non-RDMM measures do not reach their unbinned value by 100 bins. For the random walkers this was only true of the increment entropy and permutation entropy, and only to a small degree. On this data we see that coarse-graining the data, even a little bit, significantly effects the reported uncertainty by these measures (as well as the permutation test). The reason is because these measures track the short-term fluctuations in the data, and so greater resolution exposes more noise (fewer ties) and the measure values continue to increase with increasing resolution because the data is "messier" than the simulated random walks (idiosyncratic changes across more time steps).

Entropy and turbulence$_{0.5}$ are steadily increasing across increasing resolution, but more slowly and smoothly than they did for the random walk and walking sine wave. Edge density is decreasing smoothly, and uniformity follows edge density closely after 20 bins. Turbulence$_2$ (and turbulence$_3$, which was left out for brevity because it added no additional value) seems insufficiently sensitive the variations in edge weights to differentiate the currencies. However, seeing the square root of the measures reveals that, although the differences are small, they are ranked appropriately. From a Markov model perspective, these kinds of random walk-like sequences are very predictable, and this fact is clearly revealed in the low and decreasing uncertainty from the core RDMM measures compared to the high and increasing non-RDMM measures.

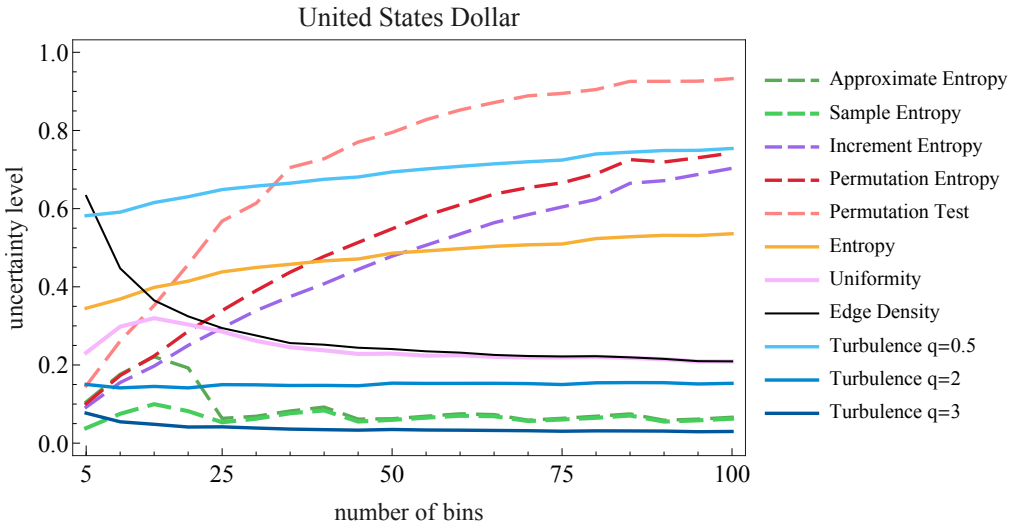

Variance
Volatility
Jaggedness
Approximate Entropy
Sample Entropy
Increment Entropy
uncertainty level
number of bins
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100∞

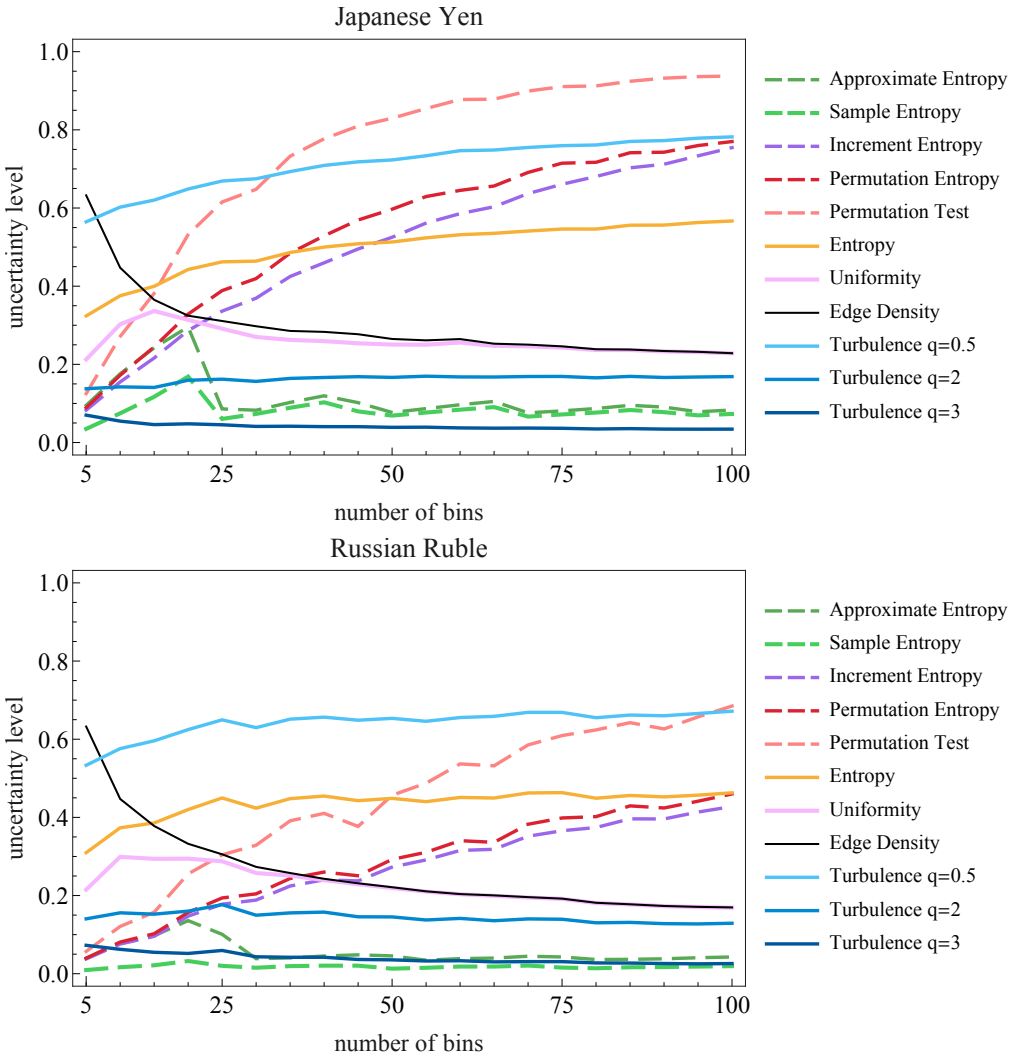

Permutation Entropy
Permutation Test
Runs Test
Entropy
Uniformity
Edge Density
uncertainty level
number of bins
0.0
0.2
0.4
0.6
0.8
1.0
5
25
50
75
100∞
100




## D.2 Exchange Rate Measure Values by Dataset

We now look at the comparison of the measures for each currency to check relative values and patterns. The first obvious pattern is that the same set of measures is increasing with increasing resolution in each case. Increment and permutation entropies follow one pattern and RDMM entropy and turbulence$_{0.5}$ follow another pattern or increasing values. The overall pattern of all the measures is extremely similar to that found on the random walk dataset for both the Dollar and Yen, but much less so for the Ruble. In fact the Ruble is less uncertain for all the measures (both RDMM and non-RDMM), although the difference is more drastic for the non-RDMM measures.

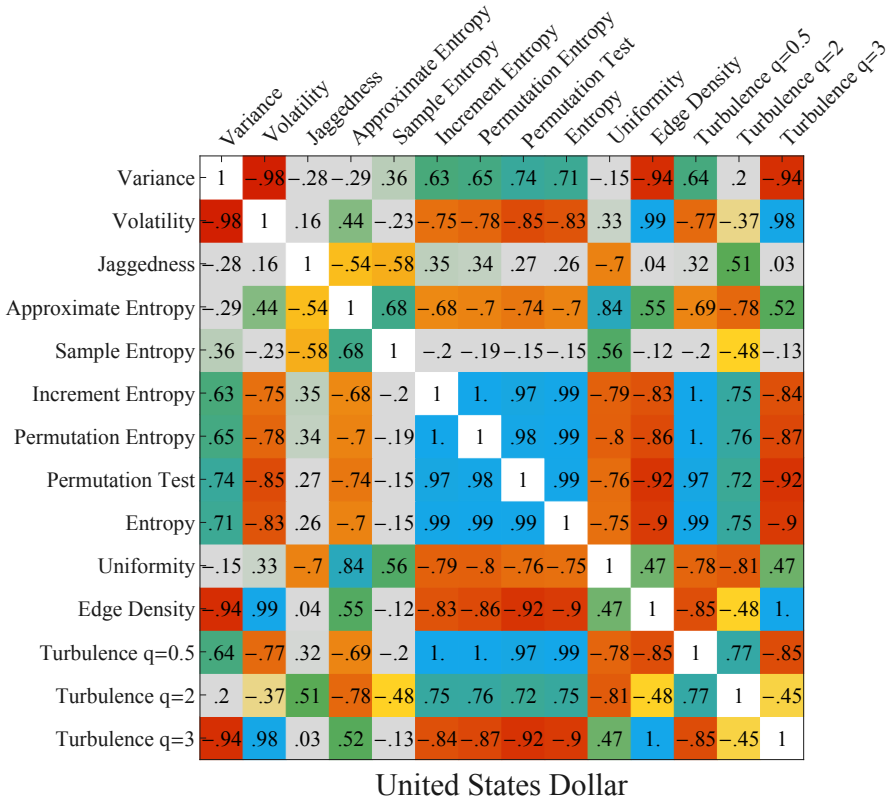



Japanese Yen
uncertainty level
number of bins
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3

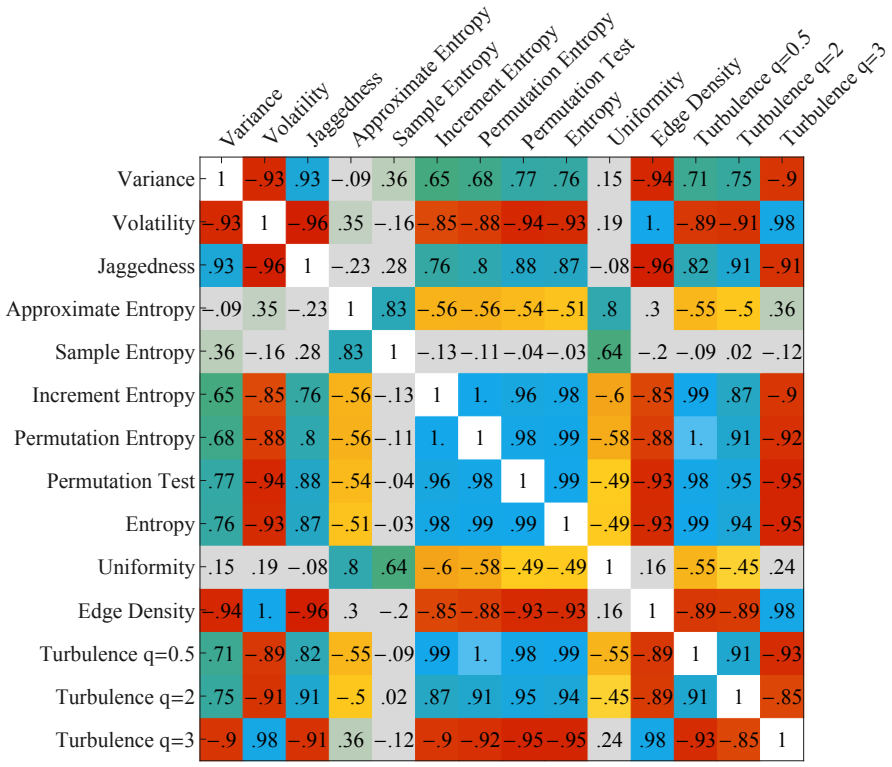

Russian Ruble
uncertainty level
number of bins
1.0
0.8
0.6
0.4
0.2
0.0
5
25
50
75
100
Approximate Entropy
Sample Entropy
Increment Entropy
Permutation Entropy
Permutation Test
Entropy
Uniformity
Edge Density
Turbulence q=0.5
Turbulence q=2
Turbulence q=3

## D.3 Exchange Rate Measure Correlations

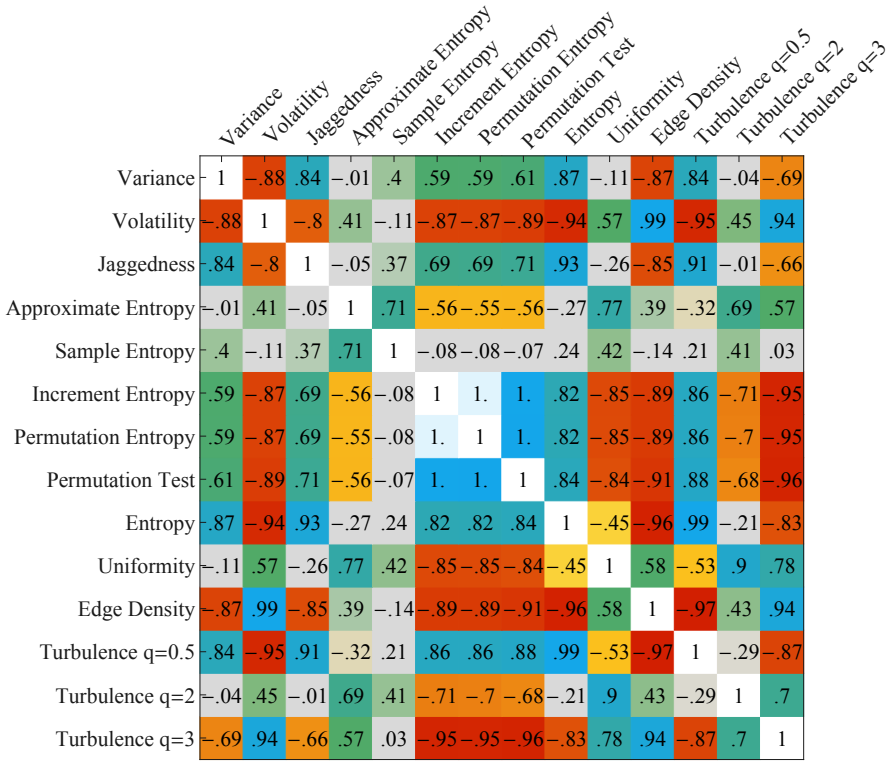
It should be no surprise by now that the three correlation matrices here also look very similar to the correlation matrix for the random walk (and random walking sine wave). The biggest differences occur for turbulence$_2$ which is now less correlated with the other RDMM measures for the Yen and Dollar. There are other particular differences, but the general pattern is the same. As we saw on the simulated data, random walk datasets are the type that produce disagreement about how the uncertainty should scale with increased resolution. If we eliminate those measures that are upward sloping, and those that are largely invariant to the edge weights, then we are left with ApEN, SampEn, and turbulence$_2$. It's not clear who the "winner" is here. On the Ruble they are correlated, but on the Yen and Dollar turbulence$_2$ is coupled to the increasing measures while ApEN and SampEn are coupled to the insensitive ones.



United States Dollar




Japanese Yen




Russian Ruble

## D.4 Combined Exchange Rate Analysis

Here we analyze the uncertainty of all three exchange rate datasets together instead of separately. One of the advantages of the RDMM encoding is that it can capture multiple dimensions simultaneously by making each node a combination of values across the dimensions. The result is still a Markov model of the phase space, and the RDMM will only include the parts of the discretized space that are actually observed (non-empty voxel bins). Because we only need a node for those combinations of values that actually occur in the combined time series, it suffers less from the exponential expansion of the phase space volume with each added dimension. There are never more nodes in the RDMM than points in the data, and with binning there are (if the model is going to be useful) many fewer nodes than datapoints regardless of the dimensionality. As a demonstration, we build the RDMM from the three exchange rate times series together and compare the uncertainty results here to the average of the results from the (binned and unbinned) non-RDMM measures.

One thing we already noted above is that, because all these are exchange rates against the Euro, and the Euro itself fluctuated in its own value during this period, the general dynamics of each time series are similar. That is to say, they are correlated. And because they are correlated, most of the $B \times B \times B$ possible combinations of the variables are never seen, and so they do not get represented as a node of the RDMM. Compared to the one-dimensional times series above in which nearly 100% of the bins were used, we can see in Table D.2 that raising the dimension still greatly increases the number of nodes, but also greatly *reduces the percent of bins* that are occupied (when the data are correlated).

There are 4351 data points in each time series and obviously the same number in the combined time series. When we plug this into our general equations for determining the maximum number of bins that can support a full uncertainty level with 3 dimensions, the result is $B_m = 4$. Four bins *per variable*, for three variables, produces 64 voxels in the 3D space, which is the same number of bins supported by that number of data points in the 1D case above. We again analyze from 5 to 100 bins per variable, and what this means is that we can expect to see large suppression of the RDMM measures from the decreased density rather than true measures of the model structure. It also means that the number of observations per combination of value drops dramatically after the first few binning values, and with it our confidence that we have captured a real and coherent behavior in the data. We leave this issue for future work to focus on the differences among measures on the combined model versus combining the measures, but address them in the discussion section below.

Non-RDMM measures have no way to measure the aggregate uncertainty of multiple time series besides aggregating the measures derived from each one separately. None of those methods support multi-dimensional time series, nor are there clear generalizations of the calculations, although they can (and have been) applied via aggregation and/or dimension reduction to some of the key applications that are multidimensional systems (brain data, heart data, economic data). The RDMM approach can seamlessly calculate the uncertainty of these combined time series. As just noted, this does require a large volume of data to support such an analysis or else lose either (1) the needed resolution or (2) sufficient confidence in the results. Some modern datasets have the needed length, but the lack of any methods to handle them directly means that fewer than expected were collected at sufficient scales (chicken and egg problem).

The next point is about how the measures of uncertainty differ with a multidimensional analysis. One would guess that the non-RDMM measures would be unaffected by the dimensionality because they are being calculated from the individual series values (using the mean value). Well, that is true for non-binned data, but when we lower the resolution of the three-dimensional data, the selection of which points are in which bins changes with different bin boundaries. When the selection of points in a bin changes, that also changes the centroid of the points within that bin, and it is those centroid points that are used as the value for that bin to calculate the uncertainty measures. Thus combining the datasets into a three-dimensional one does change the binned non-RDMM measures even though they are calculated as the mean value across each dataset determined independently (from binning based on the values in three-dimensions).

The RDMM measures also change with binning, of course, because a Markov model build from all the dynamics of the three points considered together (i.e., the dynamics in the three-dimensional phase space) will yield a different distribution of exit transitions across all the nodes. Compared to the above plots of the currencies considered separately, you can see in the plot below that ApEn and SampEn become smoother and slightly lower. Increment entropy and the permutation-based measures are elevated both in the sense of rising earlier and reaching higher levels. Entropy and turbulence$_{0.5}$ lose their upward slopes and become "well-behaved" measures with larger values, although likely due to normalization depression. The other RDMM measures behave basically as expected: they have the same initial drop and a smooth curve towards very low values (even after applying the square root). And they are further depressed by the normalization. What these results tell us is that, given a sufficient resolution, the time series of combined exchange rates produces a Markov model that is nearly deterministic.

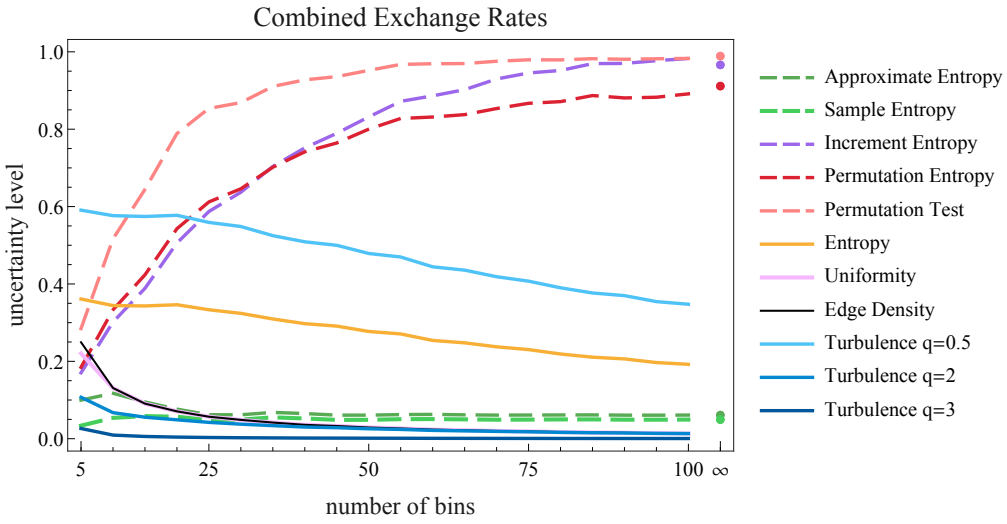
To properly interpret this, we have to consider the se-

ries' Markov model representation. Each bin 3D phase space only has a few observations, and hence only a few exit transitions. Many will have only one transition with weight 1. That extremely low value is an accurate assessment of the uncertainty of the generated Markov model, but we have good reason to question whether this Markov model is an accurate representation of the data. This leads us to the deeper discussion in the next section of the pros and cons of this RDMM methodology for measuring uncertainty.

However, what this section does demonstrate is that building and analyzing the RDMM from multidimensional data is just as easy as the single dimension case, and (due to correlation, anti-correlation, and other relationships among the data series) may reveal very different results than one could get from the aggregation of the results obtained separately. That is a clear and distinct advantage of the RDMM approach for analyzing multi-channel datasets. Although more advanced binning techniques can partially alleviate the increase in nodes while increasing the resolution, it is still the case that high-dimension applications need more data. The curse of dimensionality isn't lifted, but its symptoms are less severe.

| | Number of Bins per Variable | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 | 60 | 65 | 70 | 75 | 80 | 85 | 90 | 95 | 100 |
| Number of Nodes | 42 | 132 | 249 | 401 | 560 | 709 | 883 | 1066 | 1202 | 1392 | 1558 | 1722 | 1858 | 2023 | 2149 | 2287 | 2440 | 2505 | 2640 | 2709 |
| Proportion of Bins Used | 0.336 | 0.132 | 0.074 | 0.05 | 0.036 | 0.026 | 0.021 | 0.017 | 0.013 | 0.011 | 0.009 | 0.008 | 0.007 | 0.006 | 0.005 | 0.004 | 0.004 | 0.003 | 0.003 | 0.003 |

Table D.2: Number of nodes generated (i.e., observed bin values) and the resulting proportion of bins used for each number of bins for the combined analysis.



# E Expanded Discussion

The main paper's discussion and conclusions sections include material relating this research back to previous work, material on the importance of our main results, and directions for future work. Here we provide some expanded thoughts on each of these points as well as additional material that is slightly outside our primary focus.

**Comparison to Hidden Markov Models and Markov Chains.** There is nothing new about capturing time series as Markov models and looking for patterns in the structure of the Markov model to elucidate patterns in the system behavior. Capturing time series as Markov model facilitates measuring the difference/distance between two series [], which suffices to use Markov models for unsupervised classification (clustering) of time series data []. Typically Markov models are used for categorical data, and we simply converted our real-valued data into categorical data by binning the values; another simple mathematical feat with prior precedent but little widespread attention. In fact their dependence on categorical data (separately or in conjunction with value-ranged data) is both a limitation and benefit. For more sophisticated empirical systems, much of the heavy lifting of applying RDMMs is in the binning step, and a great deal of sophisticated AI can be (and has been) brought to bear on the binning.

We do not endeavor to cover anything like a history of Markov models, but they have been widely known and used in a variety of fields for a variety of purposes. Historically their primary use is a generative tool for predictive modeling, and in this case Hidden Markov Models (HMMs) are far more common. Their use for prediction has been largely overshadowed by more advanced techniques that can more easily handle longer memories and more sophisticated patterns, however most of these advanced predictive tools lack any descriptive or explanatory power. Capturing time series behavior as a Markov model can reveal features of its dynamics even in cases where the Markov assumption is inappropriate, and the interpretation is intuitive even if occasionally misleading. The RDMM approach here utilizes the descriptive power of Markov models to facilitate measure of the uncertainty. Here we present turbulence and the other RDMM uncertainty measures using the sequence of value directly, but the same measures can be applied to two-stage Markov models, ones generated from the sequence of slopes, or other encodings of the data into any kind of Markov model.

**Comparison to Neural Nets.** Markov models have some superficial similarities to artificial neural networks (ANN), and deriving the edge weights of RDMMs has a close analogy. Feeding in more data to either system will strengthen the edges that are repeated and weaken those that are rare. Although RDMMs can be used for within-sample prediction based on the Markovian assumption, that is not their purpose. They are meant to be descriptive only, like capturing a dataset as a histogram, but with respect to the dynamics. However, it is possible to build ("train") multiple RDMMs from distinct datasets, and then match a new dataset's RDMM to the learned ones based on their network properties (work in progress). In addition to the identification/classification task, because the RDMM encodes the dynamics in a way that is meaningful rather than a black box, they can be dissected and explored for further insights. Specific micro-behaviors can then be identified and exploited as motifs of the Markov model's network. RDMMs can also be used for unsupervised learning of behavior categories by clustering on measures of network properties beyond the uncertainty measures presented here. We are currently engaged in all these applications and others that leverage the unique properties of this methodology.

**Main Conclusions.** The main conclusion of this analysis is that the various measures of time series uncertainty/complexity track different features of time series, and thus capture different senses of uncertainty. Increment entropy and the permutation-based measures are focused on short-term local variation and do not account for global patterns, so these measures increase with increasing resolution, especially on noisy data (although increment entropy does have a distinct pattern in a few cases). This certainly is one aspect of time series uncertainty, and capturing it (with permutation entropy as recommended by our results) is useful for capturing this aspect.

ApEn and SampEn look for globally shared local pattern; so they also assign high uncertainty to time series that are noisy overall, but do a poor job at accounting for many obvious underlying patterns. Because SampEn was designed to overcome a specific known bias of ApEn, it is the recommended measure from this class. We will not make the case either way about whether these measures capture an essential feature of time series uncertainty, but considering their popularity for this use we acknowledge that they comprise another independent class of uncertainty measures.

The RDMM measures developed here offer a different global view of local variation. It assesses the predictability based on the full distribution of state transitions; i.e., how deterministic the next value is from the given value, across all values. This is implicitly claiming that being predictable is being Markovian, and here it is limited to one-step Markov models. As we have frequently repeated, we only intend the Markov encoding to be descriptive of the dynamics seen (not the generating

mechanism). Markov models built directly from the phase space capture the patterns observed, and could be used to create sequences with similar uncertainty profiles (which we confirmed), but they are unlikely to create sequences similar to the original data.

Among the RDMM measures presented here, turbulence with $q = 2$ has the best overall features as a measure of uncertainty: (1) it is not especially sensitive when the model is sparse or dense, (2) it correctly ranked each of our test cases, and (3) strikes an adequate balance between sensitivity to the data and robustness to the binning. Of particular interest is the fact that Shannon entropy of the RDMM, the standard go-to measuring for this kind of system complexity, does such a poor job. Entropy reveals itself to be overly sensitive to the bin boundaries and mischaracterizes random walks. The turbulence measures (based on the effective degree of the nodes) are tunable based on the shape parameter $q$, and are worth further investigation for further refinements and perhaps a data-driven specification of the $q$ parameter (e.g., one that scales with the density of the Markov model's adjacency matrix). Uniformity is a novel specification of a measure similar to both entropy and effective degree, but it lacks sensitivity to the specific distribution of edge weights and as a result fails to properly capture uncertainty of these systems.

RDMM's dependence on coarse-graining forces a parameter for the number of bins. We have often discussed the $B_m$ value that can be set according to a heuristic based on the number of data points and the number needed to reach maximum uncertainty. However this recommended value may recommend too few bins than what is useful for other purposes unless the dataset is very large (i.e., very long time series). Because many useful applications involve short time series, an alternative approach (pursued in future work) is a normalization of the uncertainty measures by the maximum level attainable given (1) the number of data points and (2) the number of nodes generated by a desired binning. However, we will generally be interested in relative values, so for applications on data using the same binning and time series length, the lack of a proper normalization is unproblematic.

The use of RDMMs for measuring the uncertainty of time series should be useful in many contexts, many of which are currently being explored (such as brain and cardiac signal data). There will likely be some contexts in which it is inappropriate, and an existing measure like ApEn will be preferable; the main point here is that they measure different features of the time series, and so one would benefit from an ensemble approach that combines the insights from multiple measures. The fact that Markov encodings of phase spaces have so many other applications (such as measuring robustness, capturing coherent behavior regimes, pattern recognition, and unsupervised learning of coherent behavior motifs), their use for measuring uncertainty on general time series represents only part of a suite of tools based on RDMMs.

One advantage of the RDMM approach is the ability to directly apply to high-dimensional time series, a capability of increasing demand. Higher dimensionality requires either a lot of data or a lower resolution, but we have seen that many of the previous measures of uncertainty perform unexpectedly and counterintuitively with changes in the binning. Because the only way to aggregate these measures is something like averaging their dimension-wise values, the combined uncertainty may not accurately reflect the patterns in the high-dimensional sequence. We demonstrated this worry by showing that the average of the RDMM measures is distinct from the RDMM measures of the combined time series, and the differences depend on the interactions/correlations of the individual series. Aside from even stricter requirements on the size of datasets, high-dimensional applications is where we expect to see the greatest comparative advantage for the RDMM approach.

**Future Work.** We have several ongoing projects based on variations of the RDMM methodology. For example, by analyzing data in moving time-windows we can map changes in the uncertainty of the time series in order to categorize behavioral regimes [41] and detect precursors to events [42]. Questions of data requirements become highlighted in this case because if we want, for example, windows of length 100, the resolution should be low, but low resolutions produce inaccurate measures of uncertainty because the contours of the data are sometimes smoothed away.

When using the RDMM technique, noise in the data is handled natively because (1) the data are binned to aggregate small variations and (2) large variations will typically show up as low probability transitions (or small deviations in the existing transitions) with only small effects on the measures of uncertainty. Furthermore, it is possible to use the edge observation frequencies to assign a (non-parametric multinomial) confidence measure to the transition probabilities. Using the confidence levels of the edges we can prune the RDMM of edges below an established threshold confidence level. However, if the idea is to remove sensor noise so it is not confused with signal variability, then this is probably better handled by data cleaning techniques specialized to the sensor/data being analyzed because "fuzzy" sequences with a strong structure may look similar to noisy with respect to the model-wide set of exit distributions.

Aside from handling higher dimensions, another advantage of the RDMM approach is the ability to capture multiple time series in a single model as independent trials. That is, rather than considering them to be different dimensions of the phase space, the separate time series can be included as separate runs of the process. This is,

of course, especially useful in applications to stochastic simulations, but can also be useful for clinical data. In this way, rather than having especially long time series it is possible to combine multiple short ones by simply increasing the transition frequency counts across all the time series before normalization and achieve higher resolutions and/or higher confidence. This assumes that all trials are generated by the same process (so the RDMM is a model of the process rather than a particular run through it) and that they are independent (in the sense that the dynamics of one do not influence the others). So it would be bad to combine behaviors of multiple people who cooperated in the same experiment, but it would be fine to combine the behaviors of multiple people went through the same experiment separately. Establishing criteria for when to consider time series as dimensions or trials is the focus of another extension of this methodology.

Measures of uncertainty are often deployed to identify changes in the characteristic behavior of time series such as cardiac arrhythmias [43], epileptic EEGs [44, 45], financial dynamics, [46], and many others. To apply these kinds of measures it is necessary to cut up the series into smaller time windows and calculate the uncertainty measures for each one. Having established the robustness and compared the evaluation of the RDMM measures here, we will next turn to their performance on tests such as identifying the precursors to seizure onset and abnormal heart rate variability using this time-windowed approach.

# References


[1] M. Vidyasagar, *Hidden Markov processes: theory and applications to biology*. Princeton University Press, 2014. 1

[2] N. Ye *et al.*, "A markov chain model of temporal behavior for anomaly detection," in *Proceedings of the 2000 IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, vol. 166, p. 169, West Point, NY, 2000. 1

[3] E. K. Rains and H. C. Andersen, "A bayesian method for construction of markov models to describe dynamics on various time-scales," *The Journal of chemical physics*, vol. 133, no. 14, p. 144113, 2010. 1

[4] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989. 1

[5] S. Fine, Y. Singer, and N. Tishby, "The hierarchical hidden markov model: Analysis and applications," *Machine learning*, vol. 32, no. 1, pp. 41–62, 1998. 1

[6] C. R. Shalizi and K. L. Shalizi, "Blind construction of optimal nonlinear recursive predictors for discrete sequences," June 2004. 1

[7] W. Zucchini, I. L. MacDonald, and R. Langrock, *Hidden Markov models for time series: an introduction using R*, vol. 150. CRC press, 2016. 1

[8] G. Tauchen, "Finite state markov-chain approximations to univariate and vector autoregressions," *Economics letters*, vol. 20, no. 2, pp. 177–181, 1986. 1

[9] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web.," Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120. 1

[10] N. Singhal and V. S. Pande, "Error analysis and efficient sampling in markovian state models for molecular dynamics," *The Journal of chemical physics*, vol. 123, no. 20, p. 204909, 2005. 1

[11] R. V. Donner, Y. Zou, J. F. Donges, N. Marwan, and J. Kurths, "Ambiguities in recurrence-based complex network representations of time series," *Physical Review E*, vol. 81, p. 015101(R), 2010. 1

[12] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Physics Reports*, vol. 424, no. 4-5, pp. 175–308, 2006. 1

[13] J. Zhang and M. Small, "Complex network from pseudoperiodic time series: Topology versus dynamics," *Physical Review Letters*, vol. 96, p. 238701, 2006. 1

[14] R. V. Donner, Y. Zou, J. F. Donges, N. Marwan, and J. Kurths, "Recurrence networks - a novel paradigm for nonlinear time series analysis," Aug 2009. 1

[15] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication*. University of Illinois Press, 1971. 3

[16] M. Laakso and R. Taagepera, "Effective number of parties: A measure with application to west europe," *Comparative Political Studies*, vol. 12, pp. 3–27, 1979. 4

[17] A. Barrat, M. Barthelemy, R. Pastor-Satorras, and A. Vespignani, "The architecture of complex weighted networks.," *Proceedings of the National Academy of Sciences*, vol. 101 (11), pp. 3747–3752, 2004. 4

[18] T. Opsahl, F. Agneessens, and J. Skvoretz, "Node centrality in weighted networks: Generalizing degree and shortest paths.," *Social Networks*, vol. 32, pp. 245–251, 2010. 4

[19] E. H. Simpson, "Measurement of diversity," *Nature*, vol. 163, p. 688, 1949. 4

[20] S. M. Pincus, "Approximate entropy as a measure of system complexity.," *Proceedings of the National Academy of Sciences*, vol. 88, no. 6, pp. 2297–2301, 1991. 6

[21] J. Bruhn, H. Röpcke, and A. Hoeft, "Approximate entropy as an electroencephalographic measure of anesthetic drug effect during desflurane anesthesia," *The Journal of the American Society of Anesthesiologists*, vol. 92, no. 3, pp. 715–726, 2000. 6

[22] J. M. Yentes, N. Hunt, K. K. Schmid, J. P. Kaipust, D. McGrath, and N. Stergiou, "The appropriate use of approximate entropy and sample entropy with short data sets," *Annals of biomedical engineering*, vol. 41, no. 2, pp. 349–365, 2013. 6

[23] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 278, no. 6, pp. H2039–H2049, 2000. 6

[24] C. Bandt and B. Pompe, "Permutation entropy: a natural complexity measure for time series," *Physical review letters*, vol. 88, no. 17, p. 174102, 2002. 7

[25] B. Fadlallah, B. Chen, A. Keil, and J. Principe, "Weighted-permutation entropy: A complexity measure for time series incorporating amplitude information," *Physical Review E*, vol. 87, no. 2, p. 022911, 2013. 7

[26] X. Liu, A. Jiang, N. Xu, and J. Xue, "Increment entropy as a measure of complexity for time series," *Entropy*, vol. 18, no. 1, p. 22, 2016. 7

[27] D. E. Knuth, *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1997. 7

[28] J. V. J. V. Bradley, "Distribution-free statistical tests," tech. rep., 1968. 7

[29] NIST, "Nist/sematech e-handbook of statistical methods," 4 2012. 7

[30] L. Zunino, F. Olivares, F. Scholkmann, and O. A. Rosso, "Permutation entropy based time series analysis: Equalities in the input signal can lead to false conclusions," *Physics Letters A*, vol. 381, no. 22, pp. 1883–1892, 2017. 30

[31] T. L. Eudey, J. D. Kerr, and B. E. Trumbo, "Using r to simulate permutation distributions for some elementary experimental designs," *Journal of Statistics Education*, vol. 18, no. 1, 2010. 30

[32] S. M. Pincus and A. L. Goldberger, "Physiological time-series analysis: what does regularity quantify?," *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 266, no. 4, pp. H1643–H1656, 1994. 34

[33] S. Pincus, "Approximate entropy (apen) as a complexity measure," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 5, no. 1, pp. 110–117, 1995. 34

[34] M. Costa, A. L. Goldberger, and C.-K. Peng, "Multiscale entropy analysis of biological signals," *Physical review E*, vol. 71, no. 2, p. 021906, 2005. 34

[35] K. Gabriel and J. Neumann, "A markov chain model for daily rainfall occurrence at tel aviv," *Quarterly Journal of the Royal Meteorological Society*, vol. 88, no. 375, pp. 90–95, 1962. 42

[36] J. E. Caskey Jr, "A markov chain model for the probability of precipitation occurrence in intervals of various length," *Monthly Weather Review*, vol. 91, no. 6, pp. 298–301, 1963. 42

[37] C. Haan, D. Allen, and J. Street, "A markov chain model of daily rainfall," *Water Resources Research*, vol. 12, no. 3, pp. 443–449, 1976. 42

[38] C. Alasseur, L. Husson, and F. Pérez-Fontán, "Simulation of rain events time series with markov model," in *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium*, vol. 4, pp. 2801 – 2805, 2004. 42

[39] NOAA, "Daily summries (temperature and precipitation), january 1, 2010 – december 31, 2016," August 2017. retrieved: August 07, 2017. 42

[40] B. D'ITALIA, "Reference exchange rates against euro from 01/01/2000 to 31/12/2016," 2017. 50

[41] D. Abásolo, R. Hornero, P. Espino, J. Poza, C. I. Sánchez, and R. de la Rosa, "Analysis of regularity in the eeg background activity of alzheimer's disease patients with approximate entropy," *Clinical Neurophysiology*, vol. 116, no. 8, pp. 1826–1834, 2005. 61

[42] H. Ocak, "Automatic detection of epileptic seizures in eeg using discrete wavelet transform and approximate entropy," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2027–2036, 2009. 61

[43] D. E. Lake and J. R. Moorman, "Accurate estimation of entropy in very short physiological time series: the problem of atrial fibrillation detection in implanted ventricular devices," *American Journal of Physiology-Heart and Circulatory Physiology*, vol. 300, no. 1, pp. H319–H325, 2011. 62

[44] Y. Cao, W.-w. Tung, J. Gao, V. A. Protopopescu, and L. M. Hively, "Detecting dynamical changes in time series using the permutation entropy," *Physical Review E*, vol. 70, no. 4, p. 046217, 2004. 62

[45] N. Kannathal, M. L. Choo, U. R. Acharya, and P. Sadasivan, "Entropies for detection of epilepsy in eeg," *Computer methods and programs in biomedicine*, vol. 80, no. 3, pp. 187–194, 2005. 62

[46] L. Molgedey and W. Ebeling, "Local order, entropy and predictability of financial time series," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 15, no. 4, pp. 733–737, 2000. 62