Archeologia e Calcolatori
35.1, 2024, 215-232
doi 10.19282/ac.35.1.2024.15

CQARCHAEO: A PYTHON PACKAGE FOR COSINE QUANTOGRAM ANALYSIS AND MONTE CARLO SIMULATIONS

1. Introduction

In 1974, D.G. Kendall published a seminal article addressing the archaeological challenge of identifying the presence of a 'quantum' in a numerical sample when its size is not known in advance. Using the case study of megalithic sites and testing the hypothesis that many dimensions were expressed in terms of a 5.44 ft (1.66 m) quantum, Kendall designed a statistical technique – Cosine Quantogram Analysis (CQA) – to assess the potential presence of one or more basic units in a sample.

The CQA, often integrated with Monte Carlo simulation, has been extensively employed in archaeology for studying prehistoric weight systems (e.g. Petruso 1992; Pare 1999; Rahmstorf 2010; Pakkanen 2011; Hafford 2012; Ialongo *et al.* 2021) and weight-regulated money (Ialongo *et al.* 2018; Ialongo, Lago 2021; Montalvo-Puente *et al.* 2023), becoming the standard methodology for metrological studies. Before drawing any meaningful interpretation on CQA results, it is necessary to assess their significance in statistical terms. This evaluation is customarily conducted by running a Monte Carlo simulation, that is, by performing CQA on a certain number – conventionally not less than 100 – of randomized datasets (Kendall 1974).

The Monte Carlo simulation (see § 2.5), when performed without the assistance of programming languages, would require the manual repetition of mechanical operations. In addition to the evident inefficiency of the process – e.g., user-driven repetition may more easily lead to errors – the action may also involve a considerable time investment. Possible automation of the process could be achieved using macros integrated into spreadsheets. However, the optimal solution in terms of computational power and speed is only attainable through programming. Furthermore, since both Cosine Quantogram Analysis and Monte Carlo simulations are performed on the same sample and are integral parts of the same method, it is desirable to conduct the analyses within the same environment. While there are freely downloadable spreadsheets and an R package (https://github.com/maciejkasinski/quantatools/) for performing CQA (Ialongo 2019; Poigt 2022, 88), there is currently no equally straightforward way to perform the Monte Carlo simulation within the same analysis environment.

To fill this gap, we have developed a freely downloadable Python package (CQArchaeo) designed for researchers with some programming experience. We have created a guided version of the script to run with Google Colab, suitable for researchers with minimal programming knowledge. More experienced programmers have access to the source code and can modify it according to their research needs (https://zenodo.org/records/10658914). This package enables the setting and execution of CQA and Monte Carlo simulations in a few seconds, allowing users to export results to a spreadsheet or print the 'Quantogram' along with the corresponding significance level. This package belongs to the category of so-called Little Minions, small collections of scripts that simplify specific tasks, often created by the research team itself (Thiery *et al.* 2021), but made available and downloadable via software sharing platforms such as GitHub (https://github.com/).

In this article, we outline the rationale of CQA and Monte Carlo simulations applied to archaeological data and the interpretation of the results. We illustrate the functionality of the Python package and offer detailed instructions for its proper use. We show various examples of application on archaeological data, and finally, we discuss the applicability and limitations of the analyses.

2. Methods: Kendall's formula and Monte Carlo simulation

2.1 *Cosine Quantogram Analysis*

CQA is used to determine whether the presence of one or more basic units can be hypothesized within a numeric sample. It is based on Kendall's formula (1974):

$$\phi(q) = \sqrt{2/N} \sum_{i=1}^{n} cos\,(\frac{2\pi\varepsilon_i}{q})$$

where N is the sample size, q is the targeted quantum, and ε is an error component. Each element i is divided for the tested quantum q, and ε is the remainder of this operation. The last part of the formula tests the error component ε for each single measurement in the sample of for each quantum in the analysis-range, is tested in a way that gives a score ranging between 1 (perfect fit, obtained when ε is =0 or very close to q), and -1 (no fit, obtained when ε is exactly in the middle between 0 and q) (Ialongo, Lago 2021, 5), and the individual scores of each error component are summed. Different quanta in an arbitrarily defined range are tested simultaneously, and the results are plotted in a graph that commonly goes by the name of 'Quantogram,' with the y-axis representing $\phi(q)$ (the test statistic), and the x-axis representing the quantum scale. Higher peaks of $\phi(q)$ indicate possible basic units in a numerical sample.

The CQA requires the preliminary setting of three parameters: 1) the range of values of the archaeological sample to analyze (henceforth 'analysis-range'), 2) the range of quanta to test ('quantum-range'), and 3) the interval between each tested quantum ('quantum-step'). All parameters are entirely arbitrary, but they partly depend on the actual extension of the sample to be analysed, and foremost on the initial guess of the approximate value of the relevant quantum one expects to find (i.e., the unit of measurement), as well as on the standard error of the measurement system one is trying to determine. In general, the choice of the analysis-range, quantum-range, and quantum-step depends on the hypothesis being tested through the analysis. Here we clarify how these settings work, by using past research on Bronze age weight systems in Western Eurasia as an example.

2.2 *Setting the analysis-range*

The analysis-range is an arbitrary sub-sample drawn from the complete sample of measurements. Extracting a sub-sample is a necessary step to exclude measurements that are either too much smaller or too much bigger than the expected best-fitting quantum. Simply 'feeding' the complete sample to the formula would determine the risk of producing false positives or false negatives. Before preparing the sub-sample, it is good practice to have an initial guess about the approximate value of the potentially relevant quantum one expects to identify. For example, when it comes Bronze age Western Eurasia, substantial research of the past 50 years or so has confidently established an overall interval for weight units ranging approximately between 8 g and 15 g (Ialongo *et al.* 2021). Once the initial guess *q* is defined, it is important that the analysis-range complies with two rules-of-thumb: 1) the minimum value of the analysis-range is only slightly smaller than the initial guess, and 2) the standard error of the maximum value of the analysis-range is not bigger than the initial quantum guess.

Setting the minimum value properly (i_{min}) is crucial as it significantly limits the chance of obtaining false negatives (Ialongo, Lago 2021). By testing measurements that are too much smaller than the guessed unit, in fact, one would inevitably obtain negative $\phi(q)$ scores. This is to say that, if we were to test a potentially relevant quantum of c. 10 g, we would inevitably obtain negative scores for every measurement in the analysis-range that is substantially smaller than 10 g. In a way, it would be like calculating the average height of the population of a country without excluding pre-adult individuals from the calculation. As a rule-of-thumb, i_{min} should be smaller than the initially-guessed quantum *q* but bigger than *q*/2.

The maximum value of the analysis-range (i_{max}) has a much smaller impact on the results of the analysis than i_{min} does. However, the reason why i_{max} should be set properly is that a potentially relevant small quantum (relatively to the analysis-range) is likely not significant for measurements that are much bigger than the relevant quantum itself. The reason for this is that equal-arm balances (i.e., the only type of scales that existed in the Bronze age) always produce a measurement error that is relative to the mass of the object being weighed. This can lead to a situation in which a given quantum that is a good fit for small measurements can eventually become smaller than the standard error of bigger measurements, hence invalidating its significance for larger numbers. Let us consider an example accounting for the standard error of Bronze age weight systems, calculated based on textual, archaeological and experimental evidence as a Coefficient of Variation of c. 5% (Ialongo *et al.* 2021). A weight unit of 10 g ± 5% (i.e., 9.5 – 10.5 g) will be a very good fit for a measurement of 20 g ± 5% (19 – 21 g). However, it would be a much less accurate descriptor for a measurement of 200 g ± 5% (190 – 210 g), as it would be approximately equal to its error, which, in practical terms, means that we will never know if that measurement was intended to be 19, 20, or 21 times the unit.

A good rule-of-thumb for choosing the value of i_{max} is taking into account the standard error of the system we are attempting to analyse, and make sure that the error of i_{max} is at most roughly equal to the initial guess for the relevant quantum:

$$i_{max} \approx \frac{q}{k}$$

where q is the initial guess for the relevant quantum, and k is the known standard error of the system of measurement under analysis. For the known range of Bronze age weight units (c. 8 g – 15 g), an analysis-range of i_{min}=7 and i_{max}=200 provides consistent results throughout Western Eurasia.

2.3 *Setting the quantum-range*

The quantum-range defines the range of tested quanta that is eventually showed on the graph. As far as the aim of the analysis is simply to display potentially relevant peaks – without testing for statistical significance – there is no particular recommendation. In general, it would make sense to centre the quantum range around the value of the guessed relevant quantum, as this would provide greater visual clarity. In addition, the range should not be too stretched forwards or backwards, as – for the reasons discussed above – any potential peak that is too much bigger or too much smaller than the guessed quantum is at risk of being a false positive.

When the aim of the analysis is to obtain a test of statistical significance, however, it would be good practice to limit the quantum-range to a relatively small interval centred symmetrically around the peak that one wants to test. This is recommended, as the largest the interval, the likelier is for the Monte

Carlo simulation (see below) to obtain extreme values that may not be related to the value one wants to test. It follows that the final quantum-range for a Monte Carlo simulation should be set only after it is observed that the CQA actually produces an 'interesting peak' that is deemed worth testing. A good rule-of-thumb is to set the peak that one wants to test at the centre of the graph's x-axis, and set the quantum-range between c. 0.5 – 1.5 times the peak value. For example, if one wants to test a peak value of c. 10 g, the quantum-range should be set between c. 5-15 g.

2.4 *Setting the quantum-step*

Finally, the script also requires to define an incremental step for the range of quanta. This setting is entirely arbitrary, while keeping in mind that the smaller the step, the more detailed the analysis will be, but the more quanta are tested, the heavier will be the computational load and hence, the longer the time for the script to be executed. It is important to choose a step that is short enough to test a sufficient number of possible basic units, while not being excessively detailed, considering the characteristics of the values in the sample (integers, numbers with decimals, etc.).

2.5 *Monte Carlo simulation*

The Monte Carlo simulation involves generating a conventional number of datasets by randomizing the values of the original sample, and testing each of them individually through CQA. The script records the highest $\phi(q)$ value obtained in each iteration of the analysis and, at the end of the simulation, it counts how many simulations yielded a $\phi(q)$ value greater than that recorded for the original sample. The null-hypothesis is that the observed CQA results are simply generated by chance. i.e., they are not significant. The goal of the Monte Carlo simulation is to assess the likelihood that a slightly different dataset can give equal or better results than the real sample. The rationale is that, if a random dataset can give higher values of $\phi(q)$, than one cannot exclude that the peaks observed in the archaeological sample are simply due to chance. The script uses the two conventional significance (α) thresholds of 1% and 5%, and displays them as horizontal lines directly on the graph, which allows one to visually assess the results: if the peak given by the CQA is higher than the 1% and 5% thresholds, it means that there is a probability of, respectively, less than 1% and 5% that the observed peak is due to chance. Therefore, the null-hypothesis is rejected if the archaeological sample exceeds the established α level.

3. Methods: from the Kendall's formula to programming

We have developed a Python script that takes as input a spreadsheet (CSV or XLSX) containing the data to be analysed. The input dataset should be a spreadsheet with a single column and a header row. It can include either integer numbers or float numbers with decimals preceded by a point (i.e., not a comma). The user needs to determine the data range to include in the analysis by setting the parameters of min_value (i_{min}) and max_value (i_{max}); the range of quanta to test by setting the parameters of min_quantum, max_quantum, and the step; whether to perform the Monte Carlo simulation (Montecarlo_sim = True) or not perform it (Montecarlo_sim = False). The Monte Carlo simulation has user-adjustable parameters, including the percentage to use for data randomization (mc_randomization), set by default to 15% (0.15), and the number of simulated datasets to create (mc_iterations).

After the parameters are set, the sample size *N* of the dataset is calculated based on the desired range. This allows for the computation of a coefficient used in the analysis, expressed in the first part of the Kendall formula. To solve the second part of the formula:

$$\sum_{i=1}^{n} \cos\left(\frac{2\pi\varepsilon_i}{q}\right)$$

a pandas DataFrame is populated with all the results obtained by testing each element for every quantum. The results obtained for each quantum are summed and populate the column of a DataFrame named 'Phi_q_values'. On the row with the same index, in the 'Quanta' column, there is the corresponding quantum for each sum.

In the final graph, there will be a plot of the $\phi(q)$ values for each quantum and a vertical line corresponding to the best quantum. The user can set the α level at 1% or 5% and plot it on the output graph as a horizontal line corresponding to the threshold value of $\phi(q)$.

4. CQArchaeo: instruction for use

The package is implemented using the Python programming language (https://www.python.org/) and is supported by a number of dependencies, including NumPy (https://numpy.org/) for scientific computing and data management, Pandas (https://pandas.pydata.org/) for tabular data management, and Matplotlib (https://matplotlib.org/stable/) and Seaborn (https://seaborn.pydata.org/) for plotting and visualisation. Although the package does not have a graphical interface, a number of precautions have been taken to make it as user-friendly as possible. Installation of the package is facilitated by the presence of the library in the PyPI online repository (https://pypi.org/). To install the package, simply use the following command in a terminal.

```
pip install cqarchaeo
```

the command will install the package and its dependencies. For convenience, we recommend using the Anaconda distribution (https://www.anaconda.com/) and creating a specific environment. To use the functions and perform the analysis, we recommend the use of Jupyter notebooks (https://jupyter.org/). These files allow Python code to be placed in cells that can be executed individually, providing a simpler, more intuitive interface that can be easily shared and can contain text, images, links and so on, in addition to code. Each code snippet provided from here on represents a single cell in the Jupyter notebook. For ease of use, a pre-compiled Jupyter notebook is available on the project home page, which can be easily modified to suit the user's needs thanks to its pre-defined structure. Returning to the code and structure of the package, functions need to be imported and used:

```
from cqarchaeo import CQAnalysis, compare_quantograms
```

Once the functions have been imported, we can create an instance of the CQAnalysis class named cqa with user-defined parameters.

```
cqa = CQAnalysis(data = 'measures.xlsx', min_value = 7,
max_value = 200, min_quantum = 4, max_quantum = 24, step
= 0.02, Montecarlo_sim = True, mc_parameter = 0.15, mc_it-
erations = 100)
```

Inside the parentheses are all the arguments that can be customised by the user: the date parameter indicates the path to the XLSX or CSV file containing the measurements (see § 3), while the other elements represent parameters specific to the analysis, such as the measurement range to be taken into account (min_value and max_value) and the limits of the quanta to be tested (min_quantum and max_quantum), together with the step defining the progressive number of quanta to be tested (step). These parameters have default values defined when the package was created but can be defined at will and in relation to the data to be analysed. The next parameters relate to the Monte Carlo simulation: it is possible to decide whether it should be carried out (Montecarlo_sim = True) and to set a number of parameters, including the number of iterations (mc_iterations = 100). The cqa variable created is an instance, which in a sense contains all the data and other functions that can be realised from that data. For example, it is possible to save the table of quanta and the corresponding ϕ value using the CQAnalysis.save_quanta() method, which allows the table of values to be saved in XLSX or CSV format.

In the case of our example:

```
cqa.save_quanta()
```

It is also possible to plot the quantomgram via the CQAnalysis.plot_quantogram() method:

```
cqa.plot_quantogram(figsize=(10,6), plot_best_quantum=True,
save=False, dpi=300, plot_alpha_5=True, plot_alpha_1=False)
```

This method has a number of customizable parameters, including the possibility to specify the plot width and height (in inches), save the image in PNG format, and display the confidence intervals of the Monte Carlo simulation (plot_alpha_1, plot_alpha_5). Finally, a function was developed to compare different quantograms. This is particularly interesting as it allows a number of distributions to be plotted on a single graph to compare them and check for any overlapping peaks. This function is useful when comparing 2 or more quantograms, which can be done as follows:

```
cqa1 = CQAnalysis(data = 'data1.xlsx', args, kwargs)
cqa2 = CQAnalysis(data = 'data2.xlsx', args, kwargs)
```

where data1.xlsx and data2.xlsx are the data to be analysed and args, kwargs are the customizable parameters. It should be emphasized that to get the best results from this type of analysis, the parameters must be the same for both instances. We can now use the function:

```
compare_quantograms(quantogram_list = [cqa1, cqa_2],
figsize=(10, 6), color_list=["black", "green"], al-
pha_list=[0.2, 1],label_list=None, plot_montecarlo_
bound=[True, True])
```

Note that most parameters take as input a list (in this case [cqa1, cqa2]) containing the two CQAnalysis instances. The different parameters (colour_list, alpha_list, plot_montecarlo_bound) must be lists of the same length as quantogram_list, where the first element corresponds to the first element of quantogram_list, the second element to the second element of quantogram_list, and so on. The function allows to save the image in PNG format with the save parameter. The full documentation and examples can be found on the project page (https://github.com/lrncrd/CQArchaeo), along with a link for the example notebook to Google Colab (https://colab.research.google.com), a platform for running Python code online, which can be used to test the package or perform analysis without having to download any additional programs to your own computer.

5. CQA in archaeological research

In this section, we present some archaeological case studies, highlighting situations where the CQA yields positive results; negative results; false negative

Fig. 1 – Fragmented, semi-finished and complete bronze objects from Late Bronze age battlefield of Tollense Valley (photo V. Minkus; courtesy of T. Terberger) (from Uhlig *et al*. 2019).

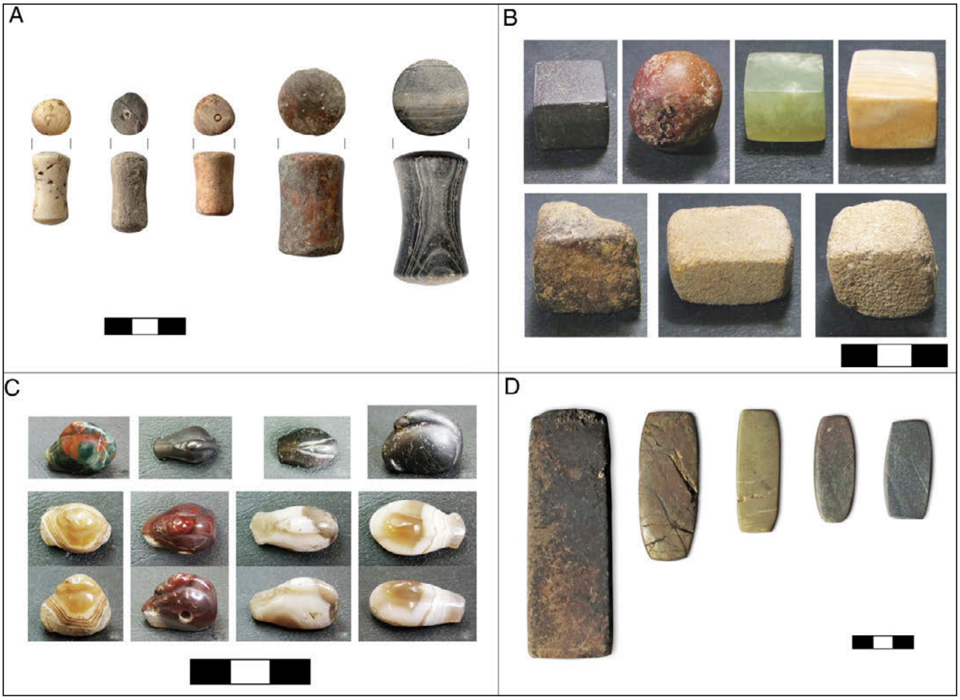

Fig. 2 – Examples of Western Eurasian balance weights of the Bronze age: A) spool-shaped weights from Tiryns, Greece (photo L. Rahmstorf); B) cubic weights from Dholavira, India (photo E. Ascalone); C) duck-shaped weights from Susa, Iran (photo E. Ascalone); D) parallelepiped weights from Lipari, Italy (photo N. Ialongo) (from Ialongo *et al*. 2021).

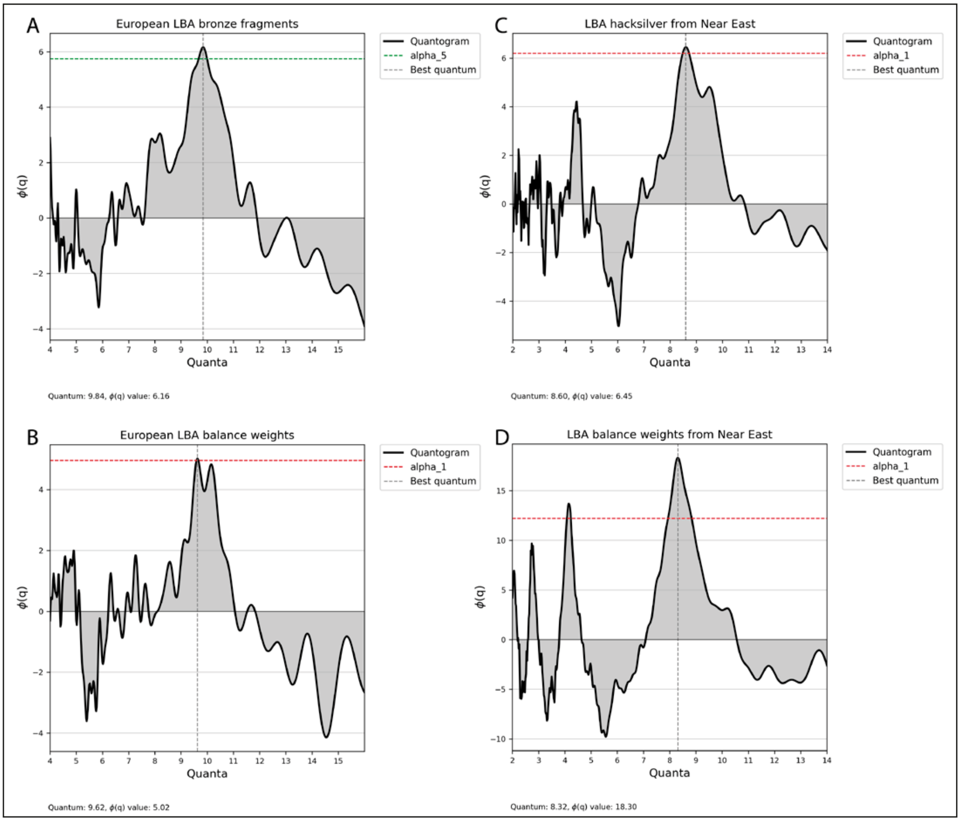

Fig. 3 – CQA and Monte Carlo simulations of weight (in grams): A) European Late Bronze age bronze fragments; B) European Bronze age balance weights; C) Late Bronze age hacksilver from Near East; D) Late Bronze age balance weights from Near East. Range (7-200 gr). Range of quanta (A-B: 4-16; C-D: 2-14). Step: 0.02.

results; false positive results. We use databases containing archaeological data that have already been published and are freely downloadable from the supplementary materials of the referenced articles.

5.1 *Positive results*

The most significant results of CQA in archaeology have been achieved in the study of weight systems and weight-regulated money. For example, the Eurasian Bronze age balance weights follow fairly clear sequences of multiples, consistent with various basic units in different regions (the complete data for the Indus valley, Mesopotamia, Aegean-Anatolia, and Europe were recently collected by Ialongo *et al.* 2021) (Fig. 2). In some of these regions,

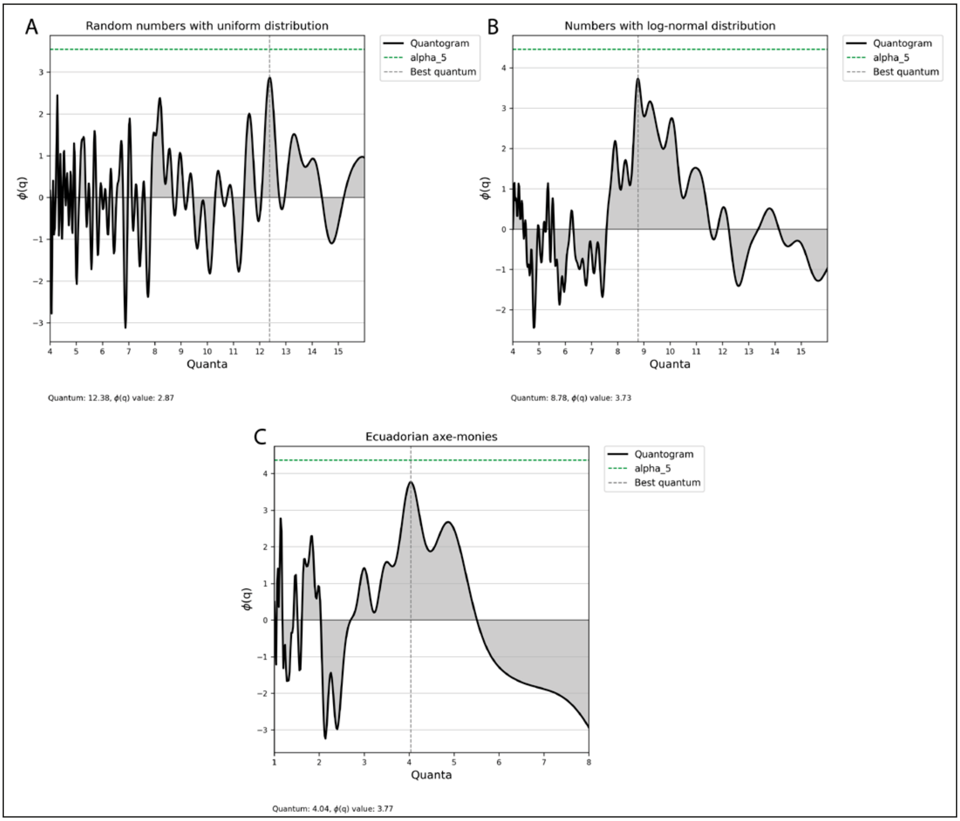

Fig. 4 – CQA and Monte Carlo simulations of weight (in grams): A) random numbers with uniform distribution; B) numbers with log-normal distribution; C) Ecuadorian axe-monies. Range (A-B: 7-200; C: 3-30). Range of quanta (A-B: 4-16; C: 1-8). Step: 0.02.

forms of money based on the weight-based fragmentation of metal have been recognized, following exactly the same system of multiples of balance weights (Europe: Ialongo, Lago 2021; Mesopotamia: Ialongo *et al.* 2018) (Fig. 1). The CQA confirms the existence of a possible basic unit, which is further validated by the Monte Carlo Simulation (1,000 iterations, α=0.01; α=0.05) (Fig. 3).

5.2 *Negative results*

For its purposes, the CQA works well for quantal distributions but yields negative results for other types of distributions. The null hypothesis for each CQA analysis is that the sample is randomly generated, and therefore

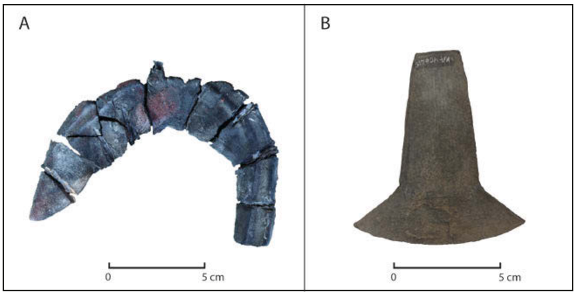

Fig. 5 – A) Replica of a bronze sickle broken during an experiment of weight-regulated fragmentation (from Lago *et al.* 2023); B) specimen of Ecuadorian 'axe-money' in arsenical copper alloy (AD 600-1532) from the Pre-Columbian Art Museum House, Quito (from Montalvo-Puente *et al.* 2023).

the observed peak is due to chance. In Python, we create two DataFrames: one containing 1,000 numbers generated randomly to follow a uniform distribution using the function provided within the numpy library's 'random' module, and the other containing numbers generated to follow a log-normal distribution using the 'lognormal' function within the same module. The results clearly show that the best quanta are, in fact, peaks of random values, well below the α level calculated with the Monte Carlo simulation (Fig. 4).

In the case of archaeological data, especially for pre-literate societies, we often do not know the processes that led to the formation of the data. However, we can hypothesize and test our assumptions. In the field of pre-contact archaeology on the South American Pacific coast, it has long been theorized those objects shaped like arsenical copper axes actually served as currency (Fig. 5, B). These objects, known as 'axe-monies', were presumably produced by carefully calculating their dimensions to result in multiples of 5 grams (Holm 1966-1967, 138).

Recently, a sample of over 700 axes was measured and weighed to test the hypothesis of a quantal distribution of weight or dimensions. The results of CQA and Monte Carlo simulation (1,000 iterations, α=0.05) have demonstrated that this hypothesis is not tenable, and the best quantum – which is around 4 grams and not 5 grams as previously assumed – is well below the established α level. It is not possible, therefore, to exclude that the observed peak is simply due to chance (Montalvo-Puente *et al.* 2023) (Fig. 4, C).

5.3 *False negative results*

The methodology is based on the idea that the Monte Carlo simulation helps establish the probability that the analysis result is random. However,

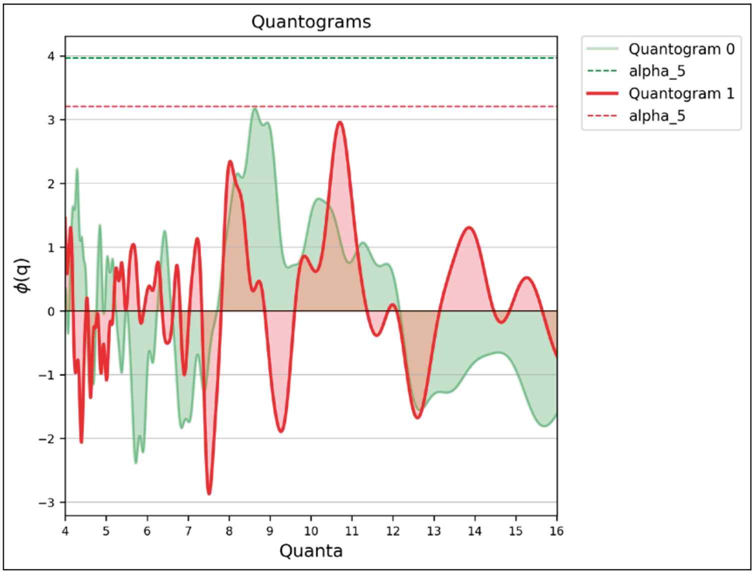

Fig. 6 – CQA and Monte Carlo simulations of weight (in grams): Quantogram 0. Bronze fragments from experimental weight-regulated fragmentation; Quantogram 1. Subsampling of archaeological fragmented sickles from European Bronze age hoards (n=117). Range (7-200). Range of quanta (4-16). Step: 0.02.

there are some circumstances that can lead to a result that is not significant enough (low $\phi(q)$) even in the presence of a dataset with a quantal distribution. A determining factor is, for example, the amount of data, especially in the presence of datasets characterized by statistical noise. A fundamental prerequisite for obtaining reliable results should be to have a large amount of data to analyze. Determining the minimum amount of data required is a problem that is challenging to solve through statistics, as it depends on the characteristics of the sample. For instance, bronze fragments of the European Bronze age are significant beyond the 0.05 α level threshold when analysing the entire published sample (n=1,397) (Ialongo, Lago 2021). By randomly subsampling the same dataset with a decreasing number of measurements and subjecting it to CQA and Monte Carlo simulation, it can be observed that the significance of the results diminishes each time. With smaller sample sizes, the best quantum is far from the α level. In this scenario, the limited amount of data may yield a false negative (Fig. 8).

Another example of how sample size influences analyses is provided by an archaeological experiment on weight-regulated fragmentation of some copper alloy sickles (Lago *et al.* 2023) (Fig. 5, A). From the breakage of 20 sickles replica, 138 fragments were obtained, of which 117 fell within the

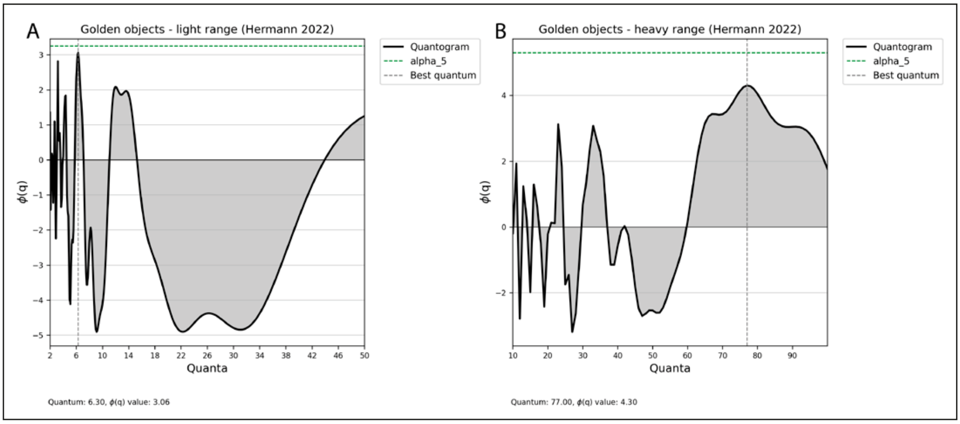

Fig. 7 – CQA and Monte Carlo simulations of weight (in grams): A) British gold objects from the Bronze age – light range; B) British gold objects from the Bronze age – Heavy range. Range (A: 1-60; B: 60-410). Range of quanta (A: 2-50; B: 10-100). Step (A: 0.1; B: 1).

value range of 7 to 200 grams. The goal of the experiment was to obtain fragments whose weight was consistent with certain multiples of 10 grams, attempting to simulate the fragmentation phenomenon observed in the European Late Bronze age (Ialongo, Lago 2021). Although concentrations of data corresponding to multiple of 10 grams were observable by the anlyses, the CQA returned a peak at approximately 8.63 grams, whose $\phi(q)$ is well below the significance threshold set at $\alpha=0.05$.

We compared the results of CQA and Monte Carlo simulations derived from the experimental fragmentation with a sub-sampling of the same size (n=117) of a dataset with 1,533 fragmented bronze sickles from archaeological contexts (Lago *et al.* 2023) (Fig. 6). The comparison was performed using the compare_quantograms function integrated into our package (see § 4). It results that with the same amount of data, even the dataset of archaeological fragmented sickles – which, when analyzed in its entirety, exhibits a quantal configuration having 9.9 as best quantum and surpassing the significance threshold set at $\alpha=0.05$ – does not appear to be quantally distributed. The higher peaks, corresponding to 8- and 11-grams ca., are similar but not the same as the established basic unit around 10 grams. Therefore, we can assert that an insufficient amount of data can yield incorrect results as well as false negatives.

5.4 *False positive results*

However, subsampling can also lead to false positive results. A few years ago, a sample of about 150 gold objects from the Bronze age in the British Isles and France had led to the hypothesis that these objects were used as

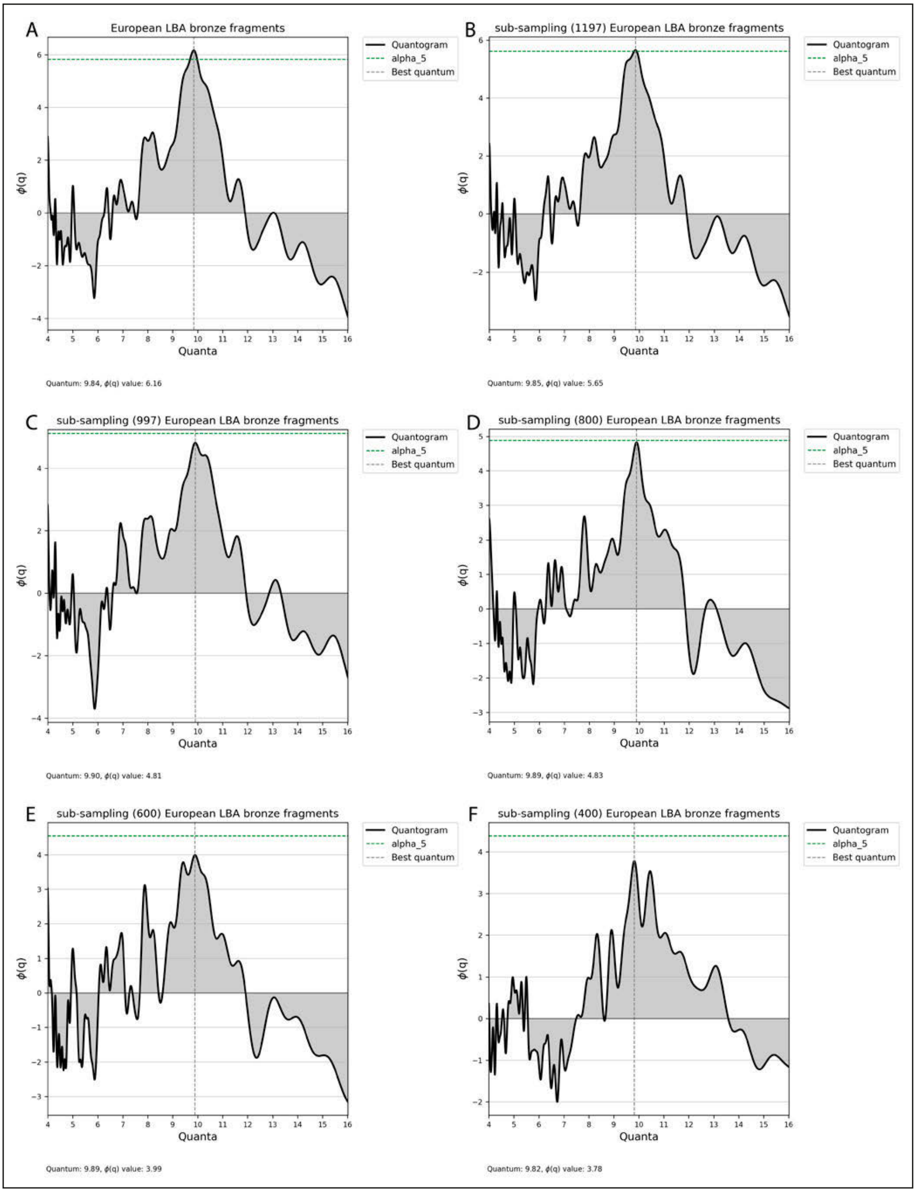

Fig. 8 – CQA and Monte Carlo simulations of weight (in grams): A) European Late Bronze age bronze fragments; B) sub-sampling (n=1,197) of European Late Bronze age bronze fragments; C) sub-sampling (n=997) of European Late Bronze age bronze fragments; D) sub-sampling (n=800) of European Late Bronze age bronze fragments; E) sub-sampling (n=600) of European Late Bronze age bronze fragments; F) sub-sampling (n=400) of European Late Bronze age bronze fragments. Range (7-200). Range of quanta (4-16). Step: 0.02.

'hackgold' weighing in exchange economies (Rahmstorf 2019). By subjecting the weight of some of these objects (bar torcs, penannular bracelets, and dress fasteners) to CQA, some peaks in value were identified that exceeded the 5% α level set. More recently, the sample of gold objects weighed from the British Isles has greatly increased (n=863), and new analyses have been conducted. By increasing the sample size, the results of the CQA do not allow to exclude that the observed peaks are due to chance (Hermann 2022) (Fig. 7).

6. Discussion and final remarks

The CQA is the standard procedure to verify the existence of quantal patterns that may be hidden behind archaeological datasets, while the Monte Carlo simulation is a highly effective tool for assessing the significance of the results. Even though the rationale of the analysis has been explained in dozens of archaeological articles, yet its applicability remains somewhat underestimated. In fact, although the CQA is suitable for the analysis of any measurable phenomenon (e.g. area, volume, distance, etc.) in the archaeological field, it has had widespread application only in metrological studies on weight. In this particular context, it has provided the best possible tool for validating/refuting various long-standing hypotheses (e.g. Petruso 1992; Pare 1999; Ialongo *et al.* 2021; Ialongo, Lago 2021; Hermann 2022; Montalvo *et al.* 2023). The analysis needs to be based on solid assumptions: precise measurements and large datasets; just as important are the criteria adopted in selecting the range to analyze and the quanta to test. If these premises are respected, the CQArchaeo package provides a precious free-to-use support for researchers who do not use programming languages.

Giancarlo Lago
Dipartimento di Beni Culturali
Università del Salento
Dipartimento di Scienze Economiche (DSE)
Università di Bologna
giancarlo.lago@unibo.it

Lorenzo Cardarelli
Dipartimento di Scienze dell'Antichità
Sapienza Università di Roma
Istituto di Scienze del Patrimonio Culturale - CNR
lorenzo.cardarelli@uniroma1.it

Nicola Ialongo
Seminar für Ur- und Frühgeschichte
Georg-August-Universität Göttingen
nicola.ialongo@uni-goettingen.de

REFERENCES

Hafford W.B. 2012, *Weighting in Mesopotamia. The balance pan weights from Ur*, «Akkadica», 133, 21-65.

Hermann R. 2022, *Weight regulation in British and Irish Bronze age gold objects: A reanalysis and reinterpretation*, «Antiquity», 96, 336-353 (https://doi.org/10.15184/aqy.2021.54).

Holm O. 1966-1967, *Money axes from Ecuador*, «Folk», 8-9, 136-143.

Ialongo N. 2019, *The earliest balance weights in the West: Towards an independent metrology for Bronze age Europe*, «Cambridge Archaeological Journal», 29, 1, 103-124 (https://doi.org/10.1017/S0959774318000392).

Ialongo N., Hermann R., Rahmstorf L. 2021, *Bronze age weight systems as a measure of market integration in Western Eurasia*, «Proceedings of the National Academy of Sciences», 118, 27, e2105873118 (https://doi.org/10.1073/pnas.2105873118).

Ialongo N., Lago G. 2021, *A small change revolution. Weight systems and the emergence of the first Pan-European money*, «Journal of Archaeological Science», 129, 105379 (https://doi.org/10.1016/j.jas.2021.105379).

Ialongo N., Vacca A., Peyronel L. 2018, *Breaking down the bullion. The compliance of bullion-currencies with official weight-systems in a case-study from the ancient Near East*, «Journal of Archaeological Science», 91, 20-32 (https://doi.org/10.1016/j.jas.2018.01.002).

Kendall D.G. 1974, *Hunting quanta*, «Philosophical Transactions of the Royal Society of London. Mathematical and Physical Sciences», 276, 231-266 (https://doi.org/10.1098/rsta.1974.0020).

Lago G., Cianfoni M., Scacchetti F., Pellegrini L., La Torre A. 2023, *Breaking sickles for shaping money. Testing the accuracy of weight-based fragmentation*, «Journal of Archaeological Science: Reports», 49, 103968 (https://doi.org/10.1016/j.jasrep.2023.103968).

Montalvo-Puente C.E., Lago G., Cardarelli L., Pérez-Molina J.C. 2023, *Money or ingots? Metrological research on pre-contact Ecuadorian "axe-monies"*, «Journal of Archaeological Science: Reports», 49, 103976 (https://doi.org/10.1016/j.jasrep.2023.103976).

Pakkanen J. 2011, *Aegean Bronze age weights, chaînes opératoires and the detecting of patterns through statistical analyses*, in A. Brysbaert (ed.), *Tracing Prehistoric Social Networks through Technology: Adiachronic Perspective on the Aegean*, London, Routledge, 143-66.

Pare C.F.E. 1999, *Weights and weighing in Bronze age Central Europe*, in *Eliten in der Bronzezeit: Ergebnisse zweier Kolloquien in Mainz und Athen*, Mainz-Bonn, Verlag des Römisch-Germanischen Zentralmuseums, 43, 421-514.

Petruso K.M. 1992, *Ayia Irini. The Balance Weights: An Analysis of Weight Measurement in Prehistoric Crete and the Cycladic Islands*, Keos 8, Mainz, Philipp von Zabern.

Poigt T. 2022, *De poids et de mesure. Les instruments de pesée en Europe occidentale durant les âges des Métaux (XIV*e*-III*e *s. a.C.). Conception, usages et utilisateurs*, Bordeaux, Ausonius éditions Université Bordeaux Montaigne (https://doi.org/10.46608/dana8.9782356134165).

Rahmstorf L. 2010, *The concept of weighing during the Bronze age in the Aegean, the Near East and Europe*, in I. Morley, C. Renfrew (eds.), *The Archaeology of Measurement. Comprehending Heaven, Earth, Time in Ancient Societies*, Cambridge, Cambridge University Press, 88-105 (https://doi.org/10.1017/CBO9780511760822.012).

Rahmstorf L. 2019, *Scales, weights and weight-regulated artefacts in Middle and Late Bronze age Britain*, «Antiquity», 93, 1197-1210 (https://doi.org/10.15184/aqy.2018.257).

Thiery F., Visser R., Mennenga M. 2021, *Little minions in archaeology. An open space for RSE software and small scripts in digital archaeology*, Conference paper, «International Series of Online Research Software Events (SORSE)» (https://sorse.github.io/downloads/event-043.pdf).

Uhlig T., Krueger J., Lidke G., Jantzen D., Lorenz S., Ialongo N., Terberger T. 2019, *Lost in combat? A scrap metal find from the Bronze age battlefield site at Tollense*, «Antiquity», 93, 1211-1230 (https://doi.org/10.15184/aqy.2019.137).

ABSTRACT

Cosine Quantogram Analysis (CQA) is a statistical analysis employed in archaeology for the study of numerical datasets with hypothesized quantal distribution. To verify thesignificance of the results, the analysis is often combined with the execution of Monte Carlo simulations. In this article, we present a freely downloadable Python package (CQArchaeo) that integrates CQA and Monte Carlo simulations in the same environment, making the analysis customizable in the main parameters. We provide a guide that enables the use of this tool even for researchers with limited experience in Python programming and demonstrate the applicability, functioning, and main limitations of the analysis on some archaeological datasets.