Possibilistic Bounds for Granular Counting

Corrado Mencar^(✉)

Dipartimento di Informatica, Università degli Studi di Bari “Aldo Moro”,
70125 Bari, Italy
`corrado.mencar@uniba.it`

Abstract. Uncertain data are observations that cannot be uniquely mapped to a referent. In the case of uncertainty due to incompleteness, possibility theory can be used as an appropriate model for processing such data. In particular, granular counting is a way to count data in presence of uncertainty represented by possibility distributions. Two algorithms were proposed in literature to compute granular counting: exact granular counting, with quadratic time complexity, and approximate granular counting, with linear time complexity. This paper extends approximate granular counting by computing bounds for exact granular count. In this way, the efficiency of approximate granular count is combined with certified bounds whose width can be adjusted in accordance to user needs.

Keywords: Granular counting · Possibility theory · Uncertain data

1 Introduction

Data uncertainty may arise as a consequence of several conditions and require proper management [1]. The simplest approach is to ignore uncertainty by estimating a precise value for each observation, but this simplistic approach, though of very simple application, can lead to a distortion in the subsequent processing stages that is difficult to detect. A more comprehensive approach should take into account data uncertainty and propagate its footprint throughout the entire data processing flow. In this way, the results of data processing reveal their uncertainty, which can be evaluated to assess their ultimate usefulness.

Several theories can be applied to represent and process uncertainty, such as Probability Theory [2], which is however a particular case falling in the Granular Computing paradigm. Granular Computing also includes classical Set Theory [3], Rough Sets Theory [4], Evidence Theory [5] and Possibility Theory [6]. The choice of a particular theory depends on the nature of uncertainty; in particular, possibility theory deals with uncertainty due to incomplete information, e.g. when the value of an observation cannot be precisely determined: we will use the term *uncertain data* to denote data characterized by this specific type of uncertainty, therefore we adopt the possibilistic framework in this paper.

A common process on data is counting, which searches for the number of data samples with a specific value. Data counting is often a preliminary step for

different types of analysis, such as descriptive statistics, comparative analysis, etc. It is a fairly simple operation when data are accurate, but it becomes non-trivial when data are uncertain. In fact, the uncertainty in the data should propagate in the count, so that the results are granular rather than precise.

Recently, a definition of granular count through Possibility Theory was proposed [7]. It was shown that the resulting counts are fuzzy intervals in the domain of natural numbers. Based on this result, two algorithms for granular counting were defined: an exact granular counting algorithm with quadratic-time complexity and an approximate counting algorithm with linear-time complexity. Approximate granular counting is appealing in applications dealing with large amounts of data due to its low complexity, but a compromise must be accepted in terms of accuracy of the resulting fuzzy interval. In particular, it is not immediate to know how far is the result of the approximate count from the fuzzy interval resulting from the exact granular count.

In this paper an algorithm is proposed for bounded granular counting, which computes an interval-valued fuzzy set representing the boundaries in which the exact granular count is located. In this way, the efficiency of approximate granular count is combined with certified bounds whose width can be adjusted in accordance to user needs.

The concept of granular count and related algorithms are briefly described in Sect. 2, while the proposal of bounded granular count is introduced in Sect. 3. Section 4 reports some numerical experiments to assess the efficiency of the proposed algorithm, as well as an outline of an application in Bioinformatics.

2 Granular Count

A brief summary of Granular Counting is reported in this Section. Further details can be found in the original papers [7, 8].

We assume that data are manifested through *observations*, which refer to some objects or *referents*. The relation between observations and referents—which is called *reference*—may be uncertain in the sense that an unequivocal reference of the observation to one of the referents is not possible. We model such uncertainty with Possibility Theory [6] as we assume that uncertainty is due to the imprecision of the observation, i.e. the observation is not complete enough to make reference unequivocal.

Given a set R of referents and an observation $o \in O$, a possibility distribution is a mapping

$$\pi_o : R \mapsto [0, 1]$$

such that $\exists r \in R : \pi_o(r) = 1$. The value $\pi_o(r) = 0$ means that it is impossible that the referent r is referred by the observation, while $\pi_o(r) = 1$ means that the referent r is absolutely possible (though not certain). Intermediate values of $\pi_o(r)$ stand for gradual values of possibility, which quantify the completeness of information resulting from an observation. (More specifically, the lower the possibility degree, the more information we have to *exclude* a referent.) The possibility distributions of all observations can be arranged in a *possibilistic assignment table*, as exemplified in Table 1.

Table 1. Example of possibilistic assignment table. Each row is a possibility distribution π_{o_j} .

	r_1	r_2	r_3
o_1	1	0.3	0.54
o_2	0.8	1	0.6
o_3	1	0	0
o_4	0.864	0.91	1
o_5	1	0	0
o_6	0.5	1	0.64
o_7	1	0.8	1
o_8	0.2	0.5	1
o_9	1	0	0
o_{10}	0.6	1	0.78

2.1 Definition of Granular Count

By using the operators of Possibility Theory, as well as the assumption that observations are non-interactive (i.e. they do not influence each other), the possibility degree, that a subset $O_x \subseteq O$ of $x \in \mathbb{N}$ observations is *exactly*¹ the set of observations referring to a reference $r_i \in R$, is defined as:

$$\pi_{O_x}(r_i) = \min \left\{ \min_{o \in O_x} \pi_o(r_i), \min_{o \notin O_x} \max_{r \neq r_i} \pi_o(r) \right\} \quad (1)$$

with the convention that $\min \emptyset = 1$. Informally speaking, Eq. (1) defines the possibility degree that O_x is the subset of all and only the observations of r_i by computing the least possibility degree of two simultaneous events: (i) all observations of O_x refer to r_i , and (ii) all the other observations refer to a different referent.

In order to compute the possibility degree that the number of observations referring to a referent r_i is N_i , we are not interested in a specific set O_x , but in *any set* of x elements. We can therefore define the possibility value that the number of observations for a referent r_i is x as:

$$\pi_{N_i}(x) = \max_{O_x \subseteq O} \pi_{O_x}(r_i) \quad (2)$$

for $x \leq m$ and $\pi_{N_i}(x) = 0$ for $x > m$. Equation (2) provides a granular definition of count. Counting is imprecise because observations are uncertain.

It is possible to prove that a granular count as in Eq. (2) is a fuzzy interval in the domain of natural numbers. A fuzzy interval is a convex and normal fuzzy set on a numerical domain (in our case, it is $\mathbb{N}$). Convexity of a fuzzy set can be established by proving that all α -cuts are intervals, while normality

¹ In the sense that any observation non belonging to O_x does not refer to r_i .

Fig. 1. Exact granular count of referent r_1 as in Table 1

of the granular count is guaranteed because of the normality of the possibility distributions π_o for all $o \in O$. Figure 1 depicts an example of granular count.

2.2 Algorithms for Granular Counting

The direct application of Eq. (2) leads to an intractable counting procedure as all possible subsets of O must be considered. On the other hand, a polynomial-time algorithm can be devised by taking profit of the representation of a granular count as a fuzzy interval. In particular, the granular counting algorithm builds the fuzzy interval by considering the α -cut representation of fuzzy sets. On such basis, two variants of granular counting algorithms can be devised:

- *Exact* granular counting uses all the values of α that correspond to some possibility degree in the possibilistic assignment table;
- *Approximate* granular counting uses the values of α taken from a finite set of evenly spaced numbers over $[0, 1]$. The number of such values depend on a user-defined parameter n_α .

The approximate granular counting is more efficient than the exact version because it does not require to scan the possibilistic assignment table, though at the price of a new required parameter.

Exact granular counting (Algorithm 1) and approximate granular counting (Algorithm 2) share the same core algorithm (Algorithm 3) and only differ by how the set of α -values are computed. In essence, the core algorithm computes the granular count in an incremental way, by reckoning the α -cuts of the fuzzy interval for each α value provided in input.

In brief, the core algorithm works as follows. Given the possibilistic assignment table $\mathbf{R}$, the index i of the referent and the set A of α -cuts, the array $\mathbf{r}$ represents the possibility degrees that an observation refers to r_i , i.e. $\mathbf{r}_j = \pi_{o_j}(r_i)$

Algorithm 1: EXACTGRANULARCOUNT

Data: $\mathbf{R}, i$
 /* $\mathbf{R}$: possibil. assignment table */
 /* i : index of referent to count */
Result: $N \in [0, 1]^m$
 1 $A \leftarrow \{\alpha \in \mathbf{R} : \alpha \neq 0\};$
 2 **return** GRANULARCOUNT($\mathbf{R}, i, A$);

Algorithm 2: APPROXIMATEGRANULARCOUNTING

Data: $\mathbf{R}, i, n_\alpha$
 /* $\mathbf{R}$: possibil. assignment table */
 /* i : index of referent to count */
 /* n_α : number of α -levels */
Result: $N \in [0, 1]^m$
 1 $\varepsilon \leftarrow 10^{-12};$
 2 $A \leftarrow \{\varepsilon + k \cdot \frac{1-\varepsilon}{n_\alpha-1} : k = 0, 1, \dots, n_\alpha - 1\};$
 3 **return** GRANULARCOUNT($\mathbf{R}, i, A$);

(line 1), while $\bar{\mathbf{r}}$ represents the possibility degrees that an observation refers to any other referent different from r_i (line 2). N is the array representing the granular count (line 3). The main cycle (lines 4–17) loops over each $\alpha \in A$ and computes the bounds $x_{\min}$ and $x_{\max}$ of the corresponding α -cut (line 5). These bounds are calculated by looping over all observations (lines 6–13), so that $x_{\max}$ is incremented if the possibility degree that the current observation refers to r_i is greater than or equal to α (lines 7–8), while $x_{\min}$ further requires that the possibility degree that the observation refers to any other referent is less than α (lines 9–10). When both $x_{\min}$ and $x_{\max}$ are computed, the degrees of membership of the granular count are updated accordingly (lines 14–16).

For a fixed referent, the time-complexity of exact granular count is $\mathcal{O}(nm^2)$ (being n the number of referents and m the number of observations), while the time-complexity of approximate granular count drops to $\mathcal{O}(m(n + n_\alpha))$. In consideration that, in typical scenarios, the number of observations is very large (i.e., $m \gg n$), especially in comparison with the number of referents, it is deduced that approximate granular counting is the preferred choice in the case of very large amounts of uncertain data.

3 Bounding the Exact Granular Count

The time-complexity of approximate granular count linearly depends on the number of α values which, in turn, depend on the value of the parameter n_α . On one hand, low values of n_α lead to fast computation of granular counts; on the other hand, low values of n_α may lead to a rough estimate of the possibility degrees of the exact granular count.

Algorithm 3: GRANULARCOUNT

```

Data:  $\mathbf{R}, i, A$ 
Result:  $N \in [0, 1]^m$ 
/*  $m$  is the number of observations */
1  $\mathbf{r} \leftarrow [\mathbf{R}_{ji}]$  for  $j = 1, 2, \dots, m$ ;
2  $\bar{\mathbf{r}} \leftarrow [\max_{k \neq i} \mathbf{R}_{jk}]$  for  $j = 1, 2, \dots, m$ ;
3  $N \leftarrow [0, 0, \dots, 0]$  ( $m + 1$  times);
4 for  $\alpha \in A$  do
5    $x_{\min} \leftarrow 0$ ;  $x_{\max} \leftarrow 0$ ;
   /* Compute  $\alpha$ -cut */
6   for  $k = 1, 2, \dots, m$  do
7     if  $\mathbf{R}_{ki} \geq \alpha$  then
8        $x_{\max} \leftarrow x_{\max} + 1$ ;
9       if  $\bar{\mathbf{r}}_k < \alpha$  then
10         $x_{\min} \leftarrow x_{\min} + 1$ ;
11      end
12    end
13  end
  /* Update granular count */
14  for  $x \in x_{\min}, \dots, x_{\max}$  do
15     $N[x] \leftarrow \max\{N[x], \alpha\}$ ;
16  end
17 end
18 return  $N$ 

```

In Fig. 2 the Jaccard similarity² measure between approximate granular count and exact granular count is reported for n_α between 2 and 100: even though similarity values close to 1 are reached for $n_\alpha \gtrsim 20$, for smaller values a significant dissimilarity can be observed. In order to assess whether the discrepancy between approximate and exact granular counts is acceptable for a problem at hand, it is important to identify some bounds for the exact granular count when only an approximate count is available.

3.1 α -Cut Computation

In order to identify such bounds, a closer look at Algorithm 3 is necessary. The algorithm computes the granular count for the i -th referent given a possibilistic assignment table $\mathbf{R}$ and a set A of α -values. The main cycle within the algorithm computes the α -cut of the granular count, which is represented by the array

² Given the exact granular count π_N and an approximate count $\tilde{\pi}_N$, the Jaccard similarity index is computed as

$$S = \frac{\sum_{x=0}^m \min\{\pi_N(x), \tilde{\pi}_N(x)\}}{\sum_{x=0}^m \max\{\pi_N(x), \tilde{\pi}_N(x)\}}$$

Fig. 2. Similarity of approximate count to exact granular count for referent r_1 in Table 1

N and corresponds to the possibility distribution π_N . For a given value of α , the variable $x_{\max}$ counts the number of observations that refer to r_i with a possibility degree $\geq \alpha$; on the other hand, the variable $x_{\min}$ counts the number of observations that refer to r_i with a possibility degree $\geq \alpha$ and refer to any other referent with possibility degree $< \alpha$. As a consequence, $x_{\min} \leq x_{\max}$. Since in our analysis we will consider different values of α , we shall denote the two variables as $x_{\min}^{(\alpha)}$ and $x_{\max}^{(\alpha)}$ respectively.

By construction, the value $x_{\max}^{(\alpha)}$ corresponds to the cardinality of the set

$$O_{\max}^{(\alpha)} = \{o \in O \mid \pi_o(r_i) \geq \alpha\} \quad (3)$$

while the value $x_{\min}^{(\alpha)}$ is the cardinality of the set

$$O_{\min}^{(\alpha)} = \left\{ o \in O \mid \pi_o(r_i) \geq \alpha \wedge \max_{r \neq r_i} \pi_o(r) < \alpha \right\} \quad (4)$$

with the obvious relation that $O_{\min}^{(\alpha)} \subseteq O_{\max}^{(\alpha)}$. On this basis, it is possible to prove the following lemmas:

Lemma 1. *If $x_{\min}^{(\alpha)} > 0$, then for all $x < x_{\min}^{(\alpha)}$: $\pi_N(x) < \alpha$.*

Proof. By Definition (1), we can write

$$\pi_{O_x}(r_i) = \min \{P, Q\} \quad (5)$$

where $P = \min_{o \in O_x} \pi_o(r_i)$ and $Q = \min_{o \notin O_x} \max_{r \neq r_i} \pi_o(r)$. We focus on Q .

Let $O_x \subset O$ be a subset of x observations. Since $x < x_{\min}^{(\alpha)}$, there exists at least one observation o' belonging to $O_{\min}^{(\alpha)}$ but not to O_x , i.e. $o' \in O_{\min}^{(\alpha)} \setminus O_x$. Since $o' \in O_{\min}^{(\alpha)}$, by definition $\max_{r \neq r_i} \pi_{o'}(r) < \alpha$, therefore $Q < \alpha$ because $o' \notin O_x$. As a consequence, $\pi_{O_x}(r_i) < \alpha$. This is true for all subsets of cardinality $x < x_{\min}^{(\alpha)}$, therefore:

$$\pi_N(x) = \max_{O_x} \pi_{O_x}(r_i) < \alpha$$

Lemma 2. For all $x > x_{\max}^{(\alpha)}$: $\pi_N(x) < \alpha$.

Proof. Let $O_x \subseteq O$ be a subset of x observations. If $x_{\max}^{(\alpha)} = m$ then $O_x = \emptyset$ because there is not a number of observations greater than m ; in such a case, $\pi_N(x) = 0 < \alpha$.

Similarly to the proof of the previous lemma, we split Definition (1) as in Eq. (5) but now we focus on P . Since $x > x_{\max}^{(\alpha)}$ there exists an observation $o' \in O_x$ that does not belong to $O_{\max}^{(\alpha)}$, therefore $\pi_{o'}(r_i) < \alpha$. As a consequence, $P < \alpha$, thus $\pi_{O_x}(r_i) < \alpha$. This is true for all subsets of cardinality $x > x_{\max}^{(\alpha)}$, thus proving the thesis.

Lemma 3. For all $x_{\min}^{(\alpha)} \leq x \leq x_{\max}^{(\alpha)}$, $\pi_N(x) \geq \alpha$

Proof. Obvious from Definitions (3) and (4).

The previous lemmas show that, for a given value of α , the exact granular count must satisfy the following relations:

$$\begin{aligned} \pi_N(x) &\in [0, \alpha[\text{ if } x < x_{\min}^{(\alpha)} \vee x > x_{\max}^{(\alpha)} \\ \pi_N(x) &\in [\alpha, 1] \text{ if } x_{\min}^{(\alpha)} \leq x \leq x_{\max}^{(\alpha)} \end{aligned}$$

that is:

Theorem 1. The interval $[x_{\min}^{(\alpha)}, x_{\max}^{(\alpha)}]$ is the α -cut of π_N .

In Fig. 3 the 0.3- and 0.7- cuts are used to depict the regions that bound the values of π_N . Notice that such regions have been computed without knowing the actual values of the exact granular count.

3.2 Bounds for Exact Granular Count

Thanks to the properties of α -cuts, it is possible to identify tight bounds for an exact granular count by using the results of an approximate granular count. In fact, given two values $\alpha' < \alpha''$, the α'' -cut is included in the α' -cut, therefore

$$\left[x_{\min}^{(\alpha'')}, x_{\max}^{(\alpha'')} \right] \subseteq \left[x_{\min}^{(\alpha')}, x_{\max}^{(\alpha')} \right]$$

Fig. 3. α values bound the exact granular count π_N

and it is easy to verify the following properties:

$$\begin{aligned}
 x < x_{\min}^{(\alpha')} &\Rightarrow \pi_N(x) < \alpha' \\
 x_{\min}^{(\alpha')} \leq x < x_{\min}^{(\alpha'')} &\Rightarrow \alpha' \leq \pi_N(x) < \alpha'' \\
 x_{\min}^{(\alpha'')} \leq x \leq x_{\max}^{(\alpha'')} &\Rightarrow \pi_N(x) \geq \alpha'' \\
 x_{\max}^{(\alpha'')} < x \leq x_{\max}^{(\alpha')} &\Rightarrow \alpha' \leq \pi_N(x) < \alpha'' \\
 x > x_{\max}^{(\alpha')} &\Rightarrow \pi_N(x) < \alpha'
 \end{aligned}$$

The previous relations suggest a strategy for computing the bounds of an exact granular count: supposing that, for some x , it is known that $\alpha_1 \leq \pi_N(x) \leq \alpha_2$ and a new value α is considered: if $x_{\min}^{(\alpha)} \leq x \leq x_{\max}^{(\alpha)}$ then it is possible to assert that $\pi_N(x) \geq \alpha$, therefore $\max\{\alpha, \alpha_1\} \leq \pi_N(x) \leq \alpha_2$; if x is outside this interval, then $\pi_N(x) < \alpha$ therefore $\alpha_1 \leq \pi_N(x) \leq \min\{\alpha, \alpha_2\}$.

On the basis of such strategy, it is possible to define a bounded granular counting algorithm to compute bounds of the exact granular count when a set A of α -cuts is given, which is reported in Algorithm 4. In this algorithm the bounds are represented by the arrays N_l and N_u (lines 3–4) and they are updated for each $\alpha \in A$ so as to satisfy the above-mentioned relations (lines 15–22).

The resulting bounded granular count is an Interval-Valued Fuzzy Set (IVFS) [9] which assigns, to each $x \in \mathbb{N}$, an interval $[\pi_{N_L}(x), \pi_{N_U}(x)]$ representing the possibility distribution of $\pi_N(x)$, which may be unknown. In Fig. 4 it is shown an example of bounded granular count by generating the set of α -values as in approximate granular count with $n_\alpha = 5$. It is possible to observe that the core of the granular count (i.e. the set of x values such that $\pi_N(x) = 1$) is precisely identified because the approximate granular counting algorithm includes $\alpha = 1$ in the set of α -values to be considered. Also, since a value of α very close to 0 is also included (namely, 10^{-12}), the impossible counts (i.e. the values of x such

Algorithm 4: BOUNDEDGRANULARCOUNT

```

Data:  $\mathbf{R}, i, A$ 
Result:  $N \in [0, 1]^m$ 
/*  $m$  is the number of observations */
1  $\mathbf{r} \leftarrow [\mathbf{R}_{ji}]$  for  $j = 1, 2, \dots, m$ ;
2  $\bar{\mathbf{r}} \leftarrow [\max_{k \neq i} \mathbf{R}_{jk}]$  for  $j = 1, 2, \dots, m$ ;
/*  $N_l, N_u$  are lower and upper bounds */
3  $N_l \leftarrow [0, 0, \dots, 0]$  ( $m + 1$  times);
4  $N_u \leftarrow [1, 1, \dots, 1]$  ( $m + 1$  times);
5 for  $\alpha \in A$  do
6    $x_{\min} \leftarrow 0$ ;  $x_{\max} \leftarrow 0$ ;
   /* Compute  $\alpha$ -cut */
7   for  $k = 1, 2, \dots, m$  do
8     if  $\mathbf{R}_{ki} \geq \alpha$  then
9        $x_{\max} \leftarrow x_{\max} + 1$ ;
10      if  $\bar{\mathbf{r}}_k < \alpha$  then
11         $x_{\min} \leftarrow x_{\min} + 1$ ;
12      end
13    end
14  end
  /* Update bounds */
15  for  $x \in x_{\min}, \dots, x_{\max}$  do
16     $N_l[x] \leftarrow \max\{N_l[x], \alpha\}$  ;
17  end
18  for  $x \in 0, \dots, x_{\min} - 1$  do
19     $N_u[x] \leftarrow \min\{N_u[x], \alpha\}$  ;
20  end
21  for  $x \in x_{\max} + 1, \dots, m$  do
22     $N_u[x] \leftarrow \min\{N_u[x], \alpha\}$  ;
23  end
24 end
25 return  $N_l, N_u$ 

```

that $\pi_N(x) = 0$) are also detected. Finally, since the values of α are equally spaced in $]0, 1]$, the lengths of the bounding intervals are constant.

It also possible to set n_α in order to achieve a desired precision. By looking at Algorithm 2, it is possible to observe that the values of α are equally spaced at distance

$$\Delta\alpha = \frac{1 - \varepsilon}{n_\alpha - 1}$$

where the value $\Delta\alpha$ coincides with the maximum length of the intervals computed by the bounded granular counting algorithm. By reversing the problem, it is possible to set the value of n_α so that the maximum length is less than a desired threshold β . Since $\varepsilon \approx 0$, it suffices to set

$$n_\alpha = \left\lceil \frac{1}{\beta} \right\rceil + 1$$

Fig. 4. Bounded granular count of r_1 for $n_\alpha = 5$. Middle dots represent the values of exact granular count.

4 Experimental Results

4.1 Efficiency Evaluation

The evaluation of efficiency has been performed on synthetically generated data. In particular, a number of random possibilistic assignment tables have been generated by varying the number of observations on a geometrical progression with common ratio 10 but keeping the number of referents fixed to three.³

For each possibilistic assignment table, both exact and bounded granular counting algorithms (with $n_\alpha = 10$) have been applied on the first referent, and the time required to complete operations has been recorded.⁴ Each experiment has been repeated 7 times and average time has been recorded. For each repetition, the experiment has been looped for 10 times and the best timing has been retained. The average execution time is reported in Table 2 and depicted in Fig. 5.

A linear regression in the log-log scale confirms the quadratic trend of the time required for exact granular counting and the linear trend for bounded granular counting algorithm. Noticeably, the change of complexity is most exclusively due to the way the set of α -values have been generated: the selection of all values occurring in the possibilistic assignment table—which is required for exact granular counting—determines a significant reduction of the overall efficiency.

³ Each possibilistic assignment table has been generated by taking care that each row corresponds to a normal possibility distribution.

⁴ Experiments have been executed on a machine equipped by an Intel i7 CPU, 16GiB RAM, Linux SO. Scripts were written in Python 3.7 and executed in Jupyter Notebook. The NumPy library has been used for fast numerical computations, but the scripts were not implemented with the objective of maximizing performance.

Table 2. Average execution time on synthetic data (time in secs.)

m	Bounded g.c	Exact g.c.
10	$140 \mu\text{s} \pm 1.06 \mu\text{s}$	$206 \mu\text{s} \pm 3.26 \mu\text{s}$
100	$750 \mu\text{s} \pm 3.22 \mu\text{s}$	$12.5 \text{ ms} \pm 995 \mu\text{s}$
1,000	$7.14 \text{ ms} \pm 182 \mu\text{s}$	$1.14 \text{ s} \pm 27.6 \text{ ms}$
10,000	$72 \text{ ms} \pm 569 \mu\text{s}$	$1 \text{ min } 55 \text{ s} \pm 639 \text{ ms}$
100,000	$825 \text{ ms} \pm 65 \text{ ms}$	$3 \text{ h } 22 \text{ min}$

Fig. 5. Average execution time on synthetic data

On the other hand, bounded granular counting takes profit of the advantages of approximate granular counting for light-weight computations but, at the same time, it offers certified bounds on the possibility degrees of the exact granular count.

4.2 Application: Gene Expression Estimation

In Bioinformatics, RNA-Seq is a protocol that allows to examine the gene expression in a cell by sampling fragments of RNA called “reads”. When RNA-Seq output is mapped against a reference database of known genes, a high percentage of reads—called *multireads*—map to more than one gene [10]. Multireads are a source of uncertainty in the quantification of gene expression, which should be managed in order to provide significant results. To this end, the mapping procedure provides a quality index that is a biologically plausible estimate of the possibility that a read can be associated to a gene [11]. However, a high

Fig. 6. Bounded granular counting of reads mapping to a sample gene

quality index does not mean certainty in association: two or more genes can be candidate for mapping a read because they can be mapped with similar high quality.

Granular counting finds a natural application in the specific problem of counting the number of reads that are possibly associated to a gene. (Reads are considered as observations, while genes are referents.) However, the amount of data involved in such process may be overwhelming. For example, the public dataset SRP014005 downloaded from NCBI-SRA archive⁵, contains a case-control study of the Asthma disease with 55,579 reads mapped on 14,802 genes (16% are multireads). Nonetheless, accurate granular counting can be achieved by the use of the proposed algorithm. As an example, in Fig. 6 the bounded granular count has been computed for gene OTTHUMG00000189570—HELLPAR with $n_\alpha = 10$. It is noteworthy observing how imprecise is the count of this gene, which is due to a large number of multireads (with different quality levels).

5 Conclusions

The proposed bounded granular counting algorithm is an extended version of approximate granular counting where efficient computation is combined with the ability of bounding the exact granular count within intervals whose granularity can be decided by the user. In most cases, it is more than enough that the exact possibility degrees of exact granular count are assured to be within a small range from some approximate values. When such type of imprecision is tolerated, a significant speed-up in calculations can be achieved, thus opening the door of granular counting to big-data problems.

⁵ <ftp://ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByStudy/sra/SRP/SRP014/SRP014005>.

Acknowledgments. This work was partially funded by the INdAM - GNCS Project 2019 “Metodi per il trattamento di incertezza ed imprecisione nella rappresentazione e revisione di conoscenza”.

References

1. Boukhelifa, N., Perrin, M.-E., Huron, S., Eagan, J.: How data workers cope with uncertainty: a task characterisation study. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 3645–3656. ACM (2017)
2. Aggarwal, C.C., Yu, P.S.: A survey of uncertain data algorithms and applications. *IEEE Trans. Knowl. Data Eng.* **21**(5), 609–623 (2009)
3. Hüllermeier, E., Beringer, J.: Learning from ambiguously labeled examples. In: *Advances in Intelligent Data Analysis VI*, p. 739 (2005)
4. Lin, T.Y., Cercone, N.: *Rough Sets and Data Mining*. Springer, Boston (1996). <https://doi.org/10.1007/978-1-4613-1461-5>
5. Vannoorenberghe, P.: Reasoning with unlabeled samples and belief functions. In: *The 12th IEEE International Conference on Fuzzy Systems. FUZZ 2003*, vol. 2, pp. 814–818. IEEE (2003)
6. Dubois, D., Prade, H.: Possibility theory. In: *Computational Complexity*, pp. 2240–2252. Springer, Boston (2012). <https://doi.org/10.1007/978-1-4684-5287-7>
7. Mencar, C., Pedrycz, W.: Granular counting of uncertain data. *Fuzzy Sets Syst.* **387**, 108–126 (2019)
8. Mencar, C., Pedrycz, W.: GrCount: counting method for uncertain data. *MethodsX* **6**, 2455–2459 (2019)
9. Dubois, D., Prade, H.: Interval-valued fuzzy sets, possibility theory and imprecise probability. In: *Proceedings - 4th Conference of EUSFLAT-LFA 2005*, pp. 314–319 (2005)
10. Consiglio, A., Mencar, C., Grillo, G., Liuni, S.: Managing NGS differential expression uncertainty with fuzzy sets. In: Angelini, C., Rancoita, P.M.V., Rovetta, S. (eds.) *CIBB 2015. LNCS*, vol. 9874, pp. 42–53. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44332-4_4
11. Consiglio, A., Mencar, C., Grillo, G., Marzano, F., Caratozzolo, M.F., Liuni, S.: A fuzzy method for RNA-Seq differential expression analysis in presence of multireads. *BMC Bioinform.* **17**(S12:345), 167–182 (2016). <https://doi.org/10.1186/s12859-016-1195-2>