**Preprint**
**June 04, 2025**

# On the Computation of Large Spatial Datasets With the M function

Eric Marcon[1] Florence Puech[2]

**Abstract**
Increasing access to large geo-referenced datasets, coupled with the development of computing power, has encouraged the search for suitable spatial statistic tools. Distance-based methods have been largely developed in many scientific fields to detect spatial concentration, dispersion or independence of entities at any distance and without any bias. In a recent article, Tidu et al. (2024) highlight the qualities of the Marcon and Puech's *M* function, a relative distance-based measure, but also express reservations for the computation times required. In our article, we propose a methodological work that seeks to specify the processing of large spatialised datasets with the *M* function by using R software. We appraise the computational performance of *M* in two ways. At first, a precise evaluation of the computational time and memory requirements for geo-referenced data is carried out using the *dbmss* package in R by means of performance tests. Then, as suggested by Tidu et al., we consider an approximation of the geographical positions of the entities. The extent of the deterioration of *M*'s results is estimated and discussed, as the gains in computation time it provides. We give evidence that the individual location approximation generates information loss at very small distances, implying a trade-off between the smallest distance at which spatial interactions can be detected and computing performance. The R code used in the article is given for the reproducibility of our results.

**Keywords**
Distance-based method, M-function, Performance test, R Package dbmss

[1]AgroParisTech, UMR AMAP, CIRAD, CNRS, INRAE, IRD, Univ Montpellier, Montpellier, France.

[2]Université Paris-Saclay, INRAE, AgroParisTech, Paris-Saclay Applied Economics, F-91120 Palaiseau, France.

***Corresponding author**: eric.marcon@agroparistech.fr,

## Contents

**Introduction** 1
**1 The *M* function** 2
1.1 Main idea . . . 2
1.2 Definition . . . 3
**2 Data simulation** 3
2.1 Drawing the points . . . 3
2.2 Gridding the space . . . 3
**3 Computing *M* with the *dbmss* package** 4
3.1 Necessary data . . . 4
3.2 Point pattern . . . 4
3.3 Distance matrix . . . 4
**4 Computational performance** 5
4.1 Computing time . . . 5
4.2 Memory . . . 5
**5 Effects of approximating the position of points** 5
5.1 Case of an aggregated distribution (Matérn) . . . 5
5.2 Case of a completely random distribution (CSR) . . . 5
**6 Discussion and conclusion** 7
**Appendix** 8
**Acknowledgements** 8

## Introduction

Increasing access to large spatial datasets and greater computing power have encouraged the development of statistical analysis tools for processing such data in the best possible way (Baddeley et al., 2016). Empirical studies at very detailed geographical levels have thus been proposed in recent years for large datasets. Particular attention has been paid to detect the spatial structures (attraction, repulsion, independence) of individual spatialised data using analyses that are no longer based on zoned data but on geo-located data. This type of approach has the advantage of preserving the exact positions of the entities analysed. It has been proven that any distance-based method (by considering space as continuous) circumvents statistical bias associated with the Modifiable Areal Unit Problem – MAUP (Arbia, 1989; Openshaw and Taylor, 1979) due to discretising space into separate units. A great number of studies have shown how important it is to use this type of methodology in social sciences (Arbia, 1989; Sweeney and Feser, 1998; Marcon and Puech, 2003; Sweeney and Arabadjis, 2022) or in exact sciences (Cressie, 1993; Lentz et al., 2011; Dray et al., 2021).

In a recent article, Tidu et al. (2024) highlight the interest of a particular statistical measure, the *M* function proposed by Marcon and Puech (2010). This measure, which we will refer to as *M* in the remainder of the article, makes it possible to highlight spatial structures within a spatialised distribution (attraction, repulsion, independence) from a study based on the distances separating the entities analysed. How-

ever, while this measure preserves all the richness of individual geo-located data, it requires a longer calculation time than other distance-based measures, since it is a relative measure (see Marcon and Puech, 2017, for a literature review on the advantages and limitations of a dozen existing distance-based measures).

Tidu et al. (2024) propose to limit $M$ calculation times by introducing a voluntary positioning error for the entities analysed. For example, in their study, industrial establishments in Sardinia (Italy) are no longer located at their exact postal address but at the centroid of their municipality. This repositioning reduces calculation times, as the number of possible distances between establishments is in fact limited to the distances separating the centroids of the municipalities. This approach is similar to that of Scholl and Brenner (2015) who proposed, for the $K_d$ function (Duranton and Overman, 2005), to approximate the distances between pairs of entities by grouping them into classes. The method of Scholl and Brenner (2015), implemented in the *dbmss* package (Marcon et al., 2015) for R (R Core Team, 2025) provides a considerable gain in computational performance with little loss of accuracy. However, the information loss due to the approximation of the location of objects should imply a loss of accuracy in the estimation of their interactions at the same scale, that must be assessed.

**In our paper, we propose to test the effectiveness of Tidu et al. (2024)'s method and help the researchers to choose the appropriate method to characterise the spatial structure of quite large datasets.** First, we show the advantages to estimate $M$ on datasets with an order of magnitude of 100,000 points or less, and we show that the computation times become excessive beyond that, on a personal computer. We then study the effect of the geographical approximation of the locations of the entities analysed. This methodological work, based on a deliberately limited number of entity locations, enables us to quantify the extent of the deterioration in information that this approach creates. These performance tests provide a precise answer to the computational advantages and limitations of the $M$ function as a function of the size of the datasets.

The layout of the article is as follows. The two first sections present the $M$ function and the necessary data generated for the tests. Large point sets (in the order of several tens of thousands of points) that are either completely random or geographically concentrated are drawn. The third section details the use of the *dbmss* package to calculate $M$ and its confidence interval from a table giving the position and characteristics of the points or a matrix of distances between them. The fourth section measures the performance of *dbmss* as a function of the size of the set of points, in terms of computing time and memory requirements. The fifth section tests the spatial approximation which consists of grouping them together at the centre of the cells of a grid, following the approach of Tidu et al. (2024) which positions them at the centre of the administrative units in which they are located. In the last section, we conclude and discuss the advantages and the limits of an approximation of the locations on the results as well as on the computing time.

## 1. The *M* function

### 1.1 Main idea

Marcon and Puech (2010) introduced the $M$ function that evaluates the dependence between geo-located points without relying on a specific zoning of space. As any distance-based method, the calculation of $M$ is based on distances that separate entities under study (establishment, shops…). The idea of $M$ is simple: it compares two proportions of neighbours of interest, a local one to a global one. The local one is defined as the proportion of neighbours of interest within a distance $r$. The global one is the same proportion but defined on the whole territory. This comparison of ratios allows the detection of:

- spatial concentration (attraction) of entities if the proportion of local neighbours is greater than the one observed on the entire territory,
- spatial dispersion (repulsion) of entities if the relative proportion of local neighbours is lower than the one observed all over the territory,
- independence between entities if the local distribution of neighbours does not differ from the global one.

This comparison of proportions of neighbours defines $M$ as a *relative* distance-based measure in a strict sense (Marcon and Puech, 2017). The term *topographic* distance-based measures is preferred for those that use the surface area as a benchmark, as the well-known Ripley's $K$ function (Ripley, 1976, 1977). The $M$ function is also defined as a *cumulative* distance-based method because the local environment is appraised within a distance $r$ rather than at a distance $r$. The possibility to detect exactly at which distance(s) the spatial concentration or dispersion appears coupled with the interpretation of the results opens the way to describe very precisely the distribution of entities under study. Moreover, an easy-computation of $M$ is possible thanks to the *dbmss* R package (Marcon et al., 2015).

$M$ was at first introduced in the field of economics. Marcon and Puech (2010) proved that this function satisfies all of the requirements of Duranton and Overman (2005) for the evaluation of spatial distribution of industries. Since its introduction, various studies have described the spatial locations of industries by using $M$; for example, Jensen and Michel (2011) studied the location of shops at a urban level, Coll-Martínez et al. (2019) analysed that of the creative industries at a metropolitan level etc. This methodology has also rapidly been applied in other sciences including biology (Fernandez-Gonzalez et al., 2005), geography (Deurloo and De Vos, 2008), ecology (Marcon et al., 2012) or seismology (Nissi et al., 2013). The $M$ func-

tion is now included in general textbooks of spatial statistics such as Arbia et al. (2021), but it is less popular than $K_d$'s function of Duranton and Overman (2005); see Chain et al. (2019).

### 1.2 Definition

The $M$ function is based on the point process theory (Møller and Waagepetersen, 2004; Baddeley et al., 2016). This distance-based method was developed to analyse interactions among entities in a context of heteregeneous space. It means that within this statistical framework, we consider that any entity analysed does not have the same probability to locate everywhere on the territory (the *first-order property of the point pattern* is its intensity). Then, after controlling for space heterogeneity, we are able to identify interactions and thus detect spatial concentration or dispersion (the *second-order property of the point pattern*). Space heterogeneity is a consistent assumption for studying agglomeration of industries (see the discussion of Duranton and Overman, 2005, on that subject).

The definition of $M$ it as follows. It compares the relative proportion of entities of interest up to each distance $r$ to the same ratio but defined over the entire territory under study. In this article, we only consider the intra-type version of $M$: we study the spatial structure of neighbouring points of the same type (called *points of interest*) as the points at the centres of the disks of radius $r$. In a mathematical terms, let us denote:

- $x_i^s$, the location of point $i$ of the reference type $s$, at the centre of the disk (the point whose neighbourhood is to be analysed),
- $x_j^s$, the location of a neighbour $j$ of the same type as point $i$,
- $x_j$, the location of a neighbour $j$ of $i$, whatever its type,
- $w(.)$, the weight of a given neighbour. In that sense, $w(x_j)$ defines the weight of a neighbour $j$ of $i$.
- $W_s$, the total weight of the points $x_j^s$,
- $W$, the total weight of all points of the dataset, whatever their type,
- $\mathbf{1}\left(\left\|x_i^s - x_j\right\| \leq r\right)$, the indicator function equal to 1 if $x_j$ is in the neighbourhood of $x_i^s$, e.g., the distance between $x_i^s$ and $x_j$ is at most equal to $r$, 0 otherwise.

The intra-type $M$ function is defined as:

$$\hat{M}(r) = \sum_i \frac{\sum_{j \neq i} \mathbf{1}\left(\left\|x_i^s - x_j^s\right\| \leq r\right) w\left(x_j^s\right)}{\sum_{j \neq i} \mathbf{1}\left(\left\|x_i^s - x_j\right\| \leq r\right) w(x_j)} \Big/ \sum_i \frac{W_s - w(x_i^s)}{W - w(x_i^s)}$$

A number of remarks must be made. The first one is that the benchmark value of $M$ is equal to 1, whatever the distance considered. It means that for any radius $r$:

- if the estimated $M$ result is above 1, the local value of the ratio is greater than the global one: a spatial concentration of entities of type $s$ within that radius is thus detected.
- if the estimated $M$ result is under 1, the local value of the ratio is lower than the global one: a spatial dispersion of entities of type $s$ within that radius is thus detected.

The second remark concerns the significance of the results. A confidence interval can be generated thanks to Monte Carlo simulations following Marcon and Puech (2010). A risk level is chosen (for example 5%) as well as the number of simulations (the greater the number of simulations, the longer is the duration of the calculation of $M$). Third, the package *dbmss* (Marcon et al., 2015) on the R software (R Core Team, 2025) can be used to compute the $M$ function. The Euclidean distance is generally preferred for the calculation of $M$ but the *dbmss* package can also to use network distances.

## 2. Data simulation

The datasets we will consider in this article are obtained by simulation. The R code is given in the appendix, which allows perfect reproducibility of the examples treated.

### 2.1 Drawing the points

A set of points is drawn by a Poisson process (whose expectation of the number of points is 5,000) in a square window of side 1. Each point is assigned a qualitative mark: "Case" or "Control". 95% of points are Controls. 5% are Cases, whose spatial structure is studied. The weight of the points is drawn from a gamma distribution with free shape and scale parameters.

In this example, the drawing of points is completely random (*complete spatial randomness*: CSR), i.e. there is no simulation of attraction or dispersion of points which could generate spatial concentrations of points (aggregates) or, on the contrary, spatial regularities (dispersions). Sets of aggregated points can be drawn in a Matérn (1960) process.

The Cases are shown in figure 1: the aggregates are clearly visible. The Controls are distributed completely randomly.

### 2.2 Gridding the space

Let's consider the simulation of the Cases obtained by the Matérn process and cut the window into a 20-by-20 square grid. This partition simulates the approximation of the position of the points of an administrative unit to the position of its centre. It is important to underline that the choice of the optimal level of the grid remains an open question, as Arbia et al. (2021) noticed (p.109): *"Unfortunately, the choice of the partitioning scheme is usually arbitrary and an optimal criterion to guide this choice is not available."*

The approximated position of points is shown on the map in figure 2. Each cell now contains only one point of each type, whose weight is the sum of the weights of the individual points.

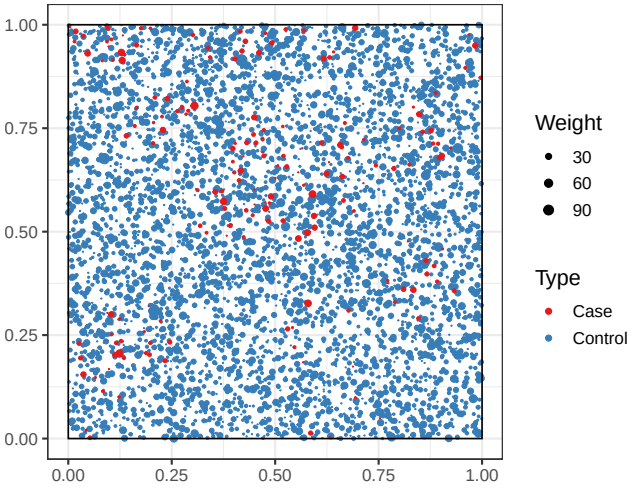


**Figure 1.** Random draw of a set of points where the Cases (in red) are aggregated and the Controls (in blue) are distributed completely randomly. The size of the points is proportional to their weight.

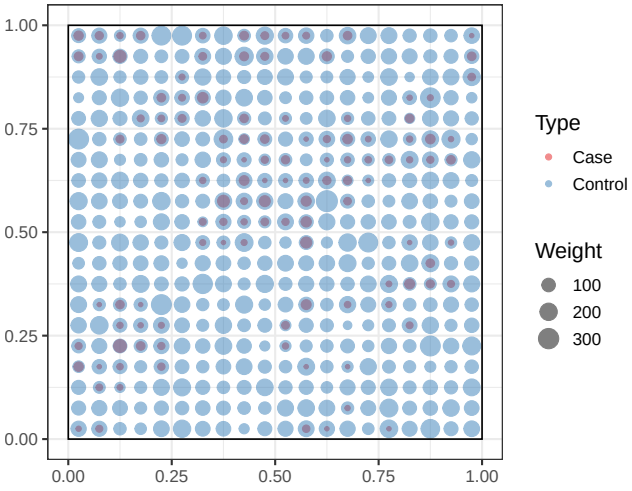


**Figure 2.** Repositioning of points in an arbitrary grid. The absence of Cases in a cell is easily detected (single-colour blue dot), as is the strong presence of Cases in a cell (two-colour dot, but predominantly red).

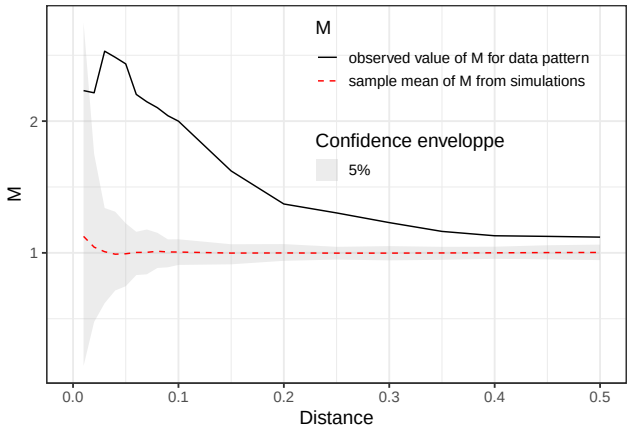


**Figure 3.** Value of $M$ as a function of distance from the reference point. The 95% confidence envelope, obtained from simulations, appears in grey and is centered on the value 1.

The values of $M$ can now be calculated from the original point set or its approximation.

## 3. Computing *M* with the *dbmss* package

### 3.1 Necessary data

In the *dbmss* package, data are a set of points or a distance matrix. The set of points in figure 1 is used. The distance matrix between all the pairs of its points is calculated to form the data on which the performance tests will be carried out.

### 3.2 Point pattern

The `Mhat()` function in the *dbmss* package is used to estimate the $M$ function. The theoretical reference value for $M$ is 1, as this function relates the proportion of Cases up to a distance $r$ to that observed over the entire window. The aggregation of Cases will be highlighted by values of $M$ greater than 1 (the relative presence of Cases is greater locally than over the whole window) and the dispersion of Cases by values less than 1. We observe (figure 3) that $M$ detects an agglomeration of Cases, which is in line with the simulation of this type of point (the controls having a completely random location on the window). The advantage of a function based on distances is clearly visible: it allows us to detect exactly at which distance(s) the attraction phenomena occur and are the most important (for functions whose values can be compared at different radii, such as $M$). In addition to estimating the $M$ function, the `Menvelope()` function can be used to calculate its global confidence interval (Duranton and Overman, 2005) under the null hypothesis of random point location. It allows parallelising the necessary simulations. The result is shown in figure 3.

### 3.3 Distance matrix

Matrices can be used to process non-Euclidean distances (transport time, road distance, etc.) which can-

not be represented by a set of points. The `Mhat()` and `MEnvelope()` functions are the same, and provide the same results whatever the form of the data (point set or distance matrix).

## 4. Computational performance

The use of $M$ to characterise the spatial structure of large sets of points may be limited by its computing time or the memory required.

### 4.1 Computing time

Calculating the distances between all pairs of points is necessary to estimate $M$. The calculation time is therefore expected to increase as the square of the number of points. The calculation time required for the exact calculation is evaluated for a range of numbers of points (figure 4a).

**The calculation time is related to the size of the set of points by a power law.** It increases less quickly than the square of the number of points. It can be estimated very precisely ($R^2 = 0.98$) by the relation $t = t_0(n/n_o)^p$ where $t$ is the estimated time for $n$ points (e.g.: 3.17 seconds for 100,000 points) knowing the time $t_0$ for $n_0$ points and $p$ is the power relation (here: 1.5).

Using a distance matrix may seem an efficient way of saving computation time, but in reality calculating distances is extremely fast and the whole process from a matrix is ultimately more time-consuming. The median execution time is equal to 14 milliseconds for estimating the $M$ function from a set of 5,000 points but 25 milliseconds for the corresponding distance matrix.

### 4.2 Memory

The memory used is evaluated for the same data sizes (figure 4b). **The memory required increases linearly with the number of points and is never critical for point set sizes that can be processed in reasonable times.** This highlights Tidu et al. (2024)'s conclusion about the power and computation time required when using $M$ on large datasets. The memory used by `Dtable` objects to calculate $M$ from a distance matrix is much greater: it is that of a numerical matrix, amounting to 8 bytes multiplied by the square of the number of points, i.e. 800 MB for 10,000 points only. Since the calculation time is not reduced by this approach, its use should be reserved for non-Euclidean distances.

## 5. Effects of approximating the position of points

Unambiguously, approximating the position of the points results in a loss of information: in each grid cell, the distance between all the points is set to zero, and the distance between two points in different cells is approximated by the distance between the centroids of the two cells. We therefore suspect a severe error in the estimation of $M$ on a small scale (of the order of magnitude of the size of the cells) and an error that decreases with distance, when the relative size of the cells decreases. The effect of the location approximation is first tested on a set of aggregated points, similar to the real Tidu et al. (2024) data. Secondly, the case of an unstructured set of points is considered.

### 5.1 Case of an aggregated distribution (Matérn)

100 sets of aggregated points (5,000 points with 5% of Cases) are simulated. To evaluate the effect of the approximation, the exact calculation and the calculation on the grid points are performed on each set of points.

The mean values of the estimates of $M$ are presented in figure 5a. The size of the grid cells is equal to 0.05. All neighbours at distances less than this threshold are placed at zero distance: the estimate of the function is constant up to this threshold and small-scale aggregation is underestimated. The correlation between the $M$ values estimated by each method is calculated at each distance in figure 5b.

Two results can be drawn from the estimated correlation's levels. Firstly, the correlation can be quite low under the size of the grid. The information on interactions at very short distances, i.e. within each grid cell, is lost, or, more precisely, approximated by its value at the grid scale. As a result, under the size of the grid the approximation of locations is not optimal. Secondly, as soon as the distance taken into account exceeds the grid cell, the correlation is very close to 1, and the estimated values are very similar. In that case, if the interactions between points are studied beyond the size of the grid, the approximation in the position of the locations may be considered.

**In the case of clustered distributions, a very careful use of location approximation is recommended, particularly for studies that suspect localised interactions at very small distances, such as the existence of information externalities or contagion phenomena. The approximation may or may not be acceptable depending on the grid size chosen.**

### 5.2 Case of a completely random distribution (CSR)

The same simulations are run with a completely random set of points. The exact calculation and the calculation on the grid points are carried out on each set of points.

The average values are shown in figure 6a. The mean value of $M$ is equal to 1 at all distances by construction: Cases and Controls are distributed completely randomly. The approximations are relatively small in value (a few percent) but artefactual aggregation is generated at small scale. As the real value of $M$ varies little around 1, the correlations are much weaker (figure 6b) in the absence of spatial structure than in the aggregated case. To sum up, **within a CSR framework, the approximation of locations appears to be acceptable.**

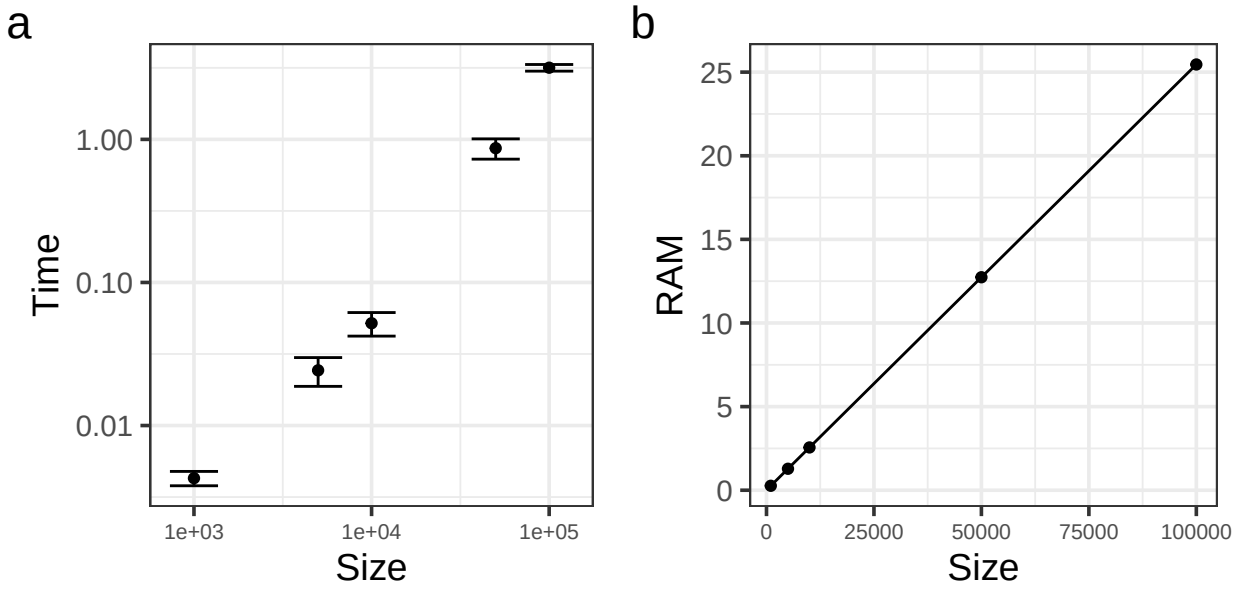


**Figure 4.** Calculation time (a) in seconds and memory required (b) in MB to estimate $M$ as a function of the size of the set of points. The bars represent the ±1 standard deviation interval.

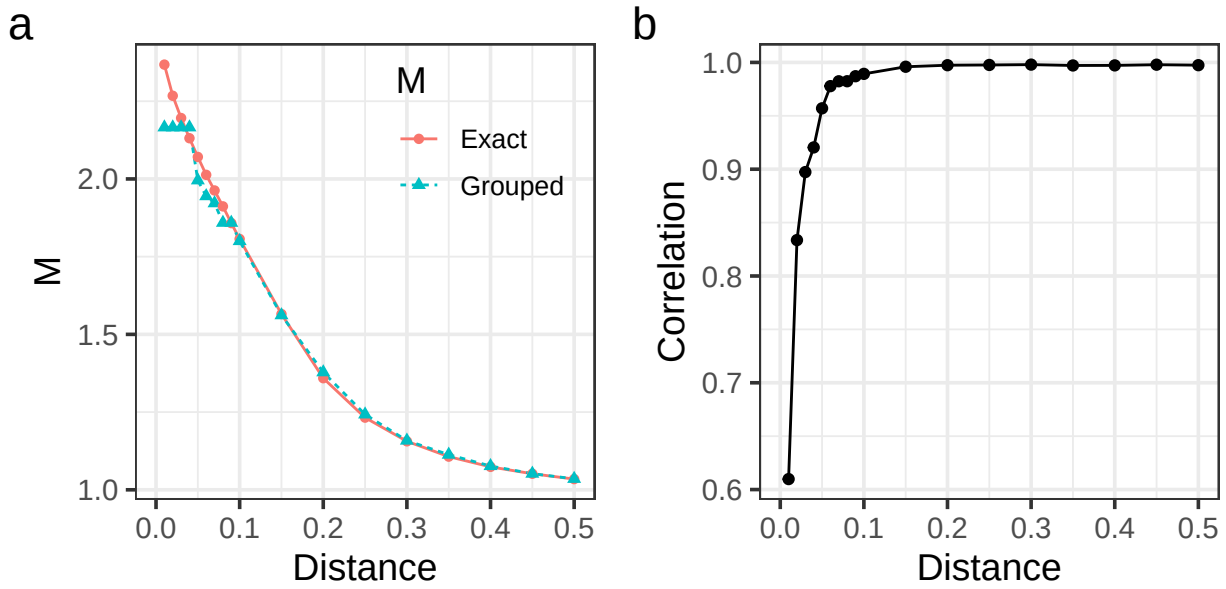


**Figure 5.** Average estimate of $M$ from the exact position of the points compared with the values obtained by grouping the points (a) and correlation between them (b). Cases form aggregates of radius 0.1.

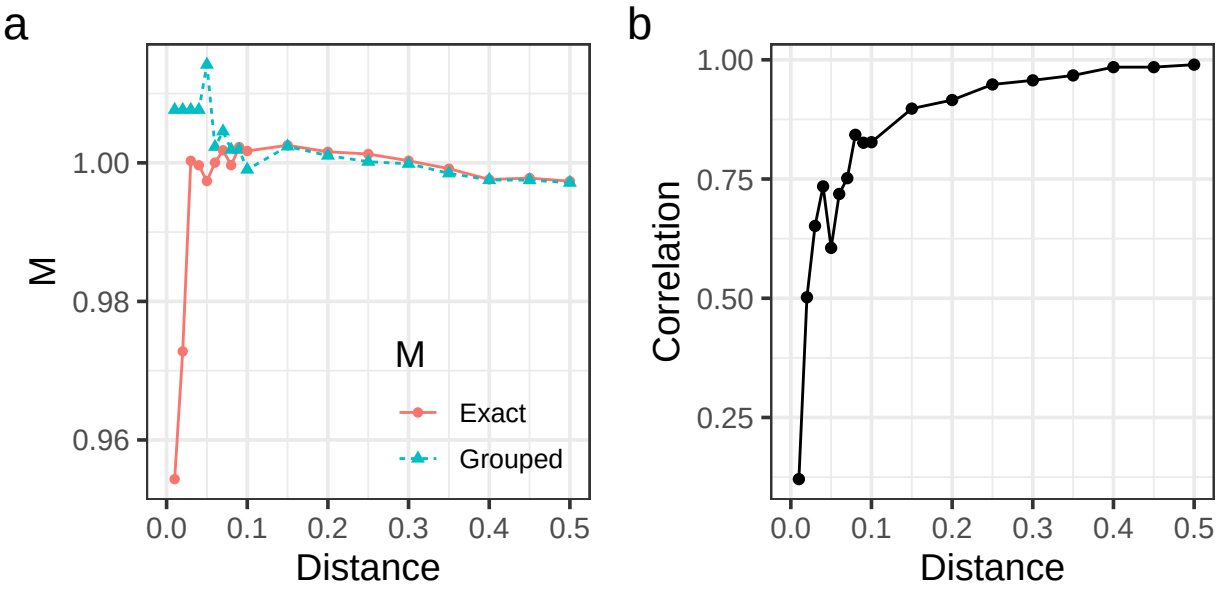


**Figure 6.** Average estimate of the $M$ function from the exact position of the points compared with the values obtained by grouping the points (a) and correlation between them (b). Both Cases and Controls are drawn in a Poisson process.

## 6. Discussion and conclusion

**The computation burden of estimating $M$ on large datasets may be an issue.** The calculation time for $M$ is below 4 seconds for a set of 100,000 points on a modern computer[1] and requires 25 MB of RAM. Calculating a confidence interval from 1,000 simulations therefore takes less than 67 minutes. For a set of five million points, the power law predicts around 11 minutes of computing time. 1,000 simulations would then take around 8 days.

Thanks to parallelization, a calculation server would drastically increase performance, but at the cost of a complexity of implementation that limits its use. If we limit ourselves to the computing power of a personal computer, **exact calculation is fully justified for data of the order of $10^5$ points**: less than an hour is enough to calculate confidence intervals. Since parallelising the simulations is offered with no effort by the *dbmss* package, this time can be reduced by a factor depending on the available hardware, say 2 to 6 with modern multicore CPU's. Beyond that, approximating the location reduces the size of the set of points to the number of locations selected. Again, it may be up to $10^5$ locations to keep computing time acceptable, whatever the size of the original dataset. The choice of the size of the grid (or the administrative scale of the aggregation of points in Tidu et al., 2024) must be done according to the scale of the interactions under study: they can not be characterized correctly at distances below it.

With regard to the errors generated in $M$'s estimates when the approximation of location is used, our findings somehow support Tidu et al. (2024)'s article, which mentions strong correlations between $M$ values computed from exact and approximated Italian company location data. In our article, a strong correlation is found, but not systematically. Since the spatial structure of their data is probably an intermediate case between the two cases dealt with in our article (aggregated and random theoretical distributions), the results provided by our two contributions are complementary. The problem with the spatial approximation comes from a possible weak correlation under the size of the grid. The loss of information can be somewhat important if interactions appear at a distance lower than the size of the grid. This situation should be analysed very attentively. Our analysis was motivated to discuss Tidu et al. (2024)'s results but a prior study of Arbia et al. (2017) has also investigated the subject of applying distance-based methods to spatial datasets that include positional errors. They proposed a first evaluation of the consequences of a positional error not *for all* of the studied entities, but only *for some of them*. This positional uncertainty for a given number of entities only, is associated to an *"unintentional positional error"*. In that situation, these uncertain geo-localised entities are placed at the centroid of the zone considered, exactly as in Tidu et al. (2024). Arbia et al. showed on a real case (Italian manufacturing firms) that the error measurement is less severe as one's can expected. Their explanation rests on the definition $M$, a relative measure: a compensation effect of positional errors is suspected between the local ratio and the global one.

To conclude, **an approximation of the spatial locations may be considered to save computation time, given the strong correlation observed between the values of $M$ on exact and approx-**

[1]The results presented here were obtained on a GitHub-hosted runner under Mac OS with a virtual 3-core Apple M1 (Virtual), similar to a fast laptop computer.

**imated data, but the scale of the grid must be fine enough to be informational at small distances even if a quite important error in the estimates may happened**. As we noticed, the choice of the optimal level of the grid is challenging, which calls for a certain degree of wariness in the use of the approximated locations. Choosing the approximation scale is a trade-off between accuracy, i.e. a small distance threshold above which results are accurate, and speed with a coarse grid.

## Appendix

R code is available at the following address: https://ericmarcon.github.io/MLargeDataSets/Appendix.pdf

## Acknowledgements

Eric Marcon benefited from an "Investissement d'Avenir" grant managed by the Agence Nationale de la Recherche (LABEX CEBA, ref. ANR-10-LBX-25) and Florence Puech gratefully acknowledges financial support from INRAE.

## References

Arbia, G. (1989). *Spatial Data Configuration in Statistical Analysis of Regional Economic and Related Problems*. Dordrecht: Kluwer.

Arbia, G., G. Espa, and D. Giuliani (2021). *Spatial Microeconometrics* (First ed.). Routledge Advanced Texts in Economics and Finance. London and New York: Routledge, Taylor & Francis Group.

Arbia, G., G. Espa, D. Giuliani, and M. M. Dickson (2017). Effects of missing data and locational errors on spatial concentration measures based on Ripley's K -function. *Spatial Economic Analysis 12*(2-3), 326–346.

Baddeley, A., E. Rubak, and R. Turner (2016). *Spatial Point Patterns: Methodology and Applications with R*. Chapman & Hall/CRC Interdisciplinary Statistics Series. Boca Raton London New York: CRC Press.

Chain, C. P., A. C. D. Santos, L. G. D. Castro, and J. W. D. Prado (2019). Bibliometric analysis of the quantitative methods applied to the measurement of industrial clusters. *Journal of Economic Surveys 33*(1), 60–84.

Coll-Martínez, E., A.-I. Moreno-Monroy, and J.-M. Arauzo-Carod (2019). Agglomeration of creative industries: An intra-metropolitan analysis for Barcelona. *Papers in Regional Science 98*(1), 409–432.

Cressie, N. A. (1993). *Statistics for Spatial Data*. New York: John Wiley & Sons.

Deurloo, M. C. and S. De Vos (2008). Measuring segregation at the micro level: An application of the M measure to multi-ethnic residential neighbourhoods in Amsterdam. *Tijdschrift voor economische en sociale geografie 99*(3), 329–347.

Dray, N., L. Mancini, U. Binshtok, F. Cheysson, W. Supatto, P. Mahou, S. Bedu, S. Ortica, E. Than-Trong, M. Krecsmarik, S. Herbert, J.-B. Masson, J.-Y. Tinevez, G. Lang, E. Beaurepaire, D. Sprinzak, and L. Bally-Cuif (2021). Dynamic spatiotemporal coordination of neural stem cell fate decisions occurs through local feedback in the adult vertebrate brain. *Cell Stem Cell 28*(8), 1457–1472.e12.

Duranton, G. and H. G. Overman (2005). Testing for localisation using micro-geographic data. *Review of Economic Studies 72*(4), 1077–1106.

Fernandez-Gonzalez, R., M. Barcellos-Hoff, and C. Ortiz-de-Solorzano (2005). A tool for the quantitative spatial analysis of complex cellular systems. *IEEE Transactions on Image Processing 14*(9), 1300–1313.

Jensen, P. and J. Michel (2011). Measuring spatial dispersion: Exact results on the variance of random spatial distributions. *The Annals of Regional Science 47*(1), 81–110.

Lentz, J. A., J. K. Blackburn, and A. J. Curtis (2011). Evaluating Patterns of a White-Band Disease (WBD) Outbreak in Acropora palmata Using Spatial Analysis: A Comparison of Transect and Colony Clustering. *PLoS ONE 6*(7), e21830.

Marcon, E. and F. Puech (2003). Evaluating the geographic concentration of industries using distance-based methods. *Journal of Economic Geography 3*(4), 409–428.

Marcon, E. and F. Puech (2010). Measures of the geographic concentration of industries: Improving distance-based methods. *Journal of Economic Geography 10*(5), 745–762.

Marcon, E. and F. Puech (2017). A typology of distance-based measures of spatial concentration. *Regional Science and Urban Economics 62*, 56–67.

Marcon, E., F. Puech, and S. Traissac (2012). Characterizing the relative spatial structure of point patterns. *International Journal of Ecology 2012*, 619281.

Marcon, E., S. Traissac, F. Puech, and G. Lang (2015). Tools to characterize point patterns: dbmss for R. *Journal of Statistical Software 67*(3), 1–15.

Matérn, B. (1960). Spatial variation. *Meddelanden från Statens Skogsforskningsinstitut 49*(5), 1–144.

Møller, J. and R. P. Waagepetersen (2004). *Statistical Inference and Simulation for Spatial Point Processes*, Volume 100 of *Monographs on Statistics and Applied Probabilities*. Chapman and Hall.

Nissi, E., A. Sarra, S. Palermi, and G. Luca (2013). The application of m-function analysis to the geographical distribution of earthquake sequence. In A. Giusti, G. Ritter, and M. Vichi (Eds.), *Classification and Data Mining*, Chapter 32, pp. 271–278. Springer Berlin Heidelberg.

Openshaw, S. and P. J. Taylor (1979). A million or so correlation coefficients: Three experiments on the modifiable areal unit problem. In N. Wrigley (Ed.), *Statistical Applications in the Spatial Sciences*, pp. 127–144. London: Pion.

R Core Team (2025). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Ripley, B. D. (1976). The foundations of stochastic geometry. *Annals of Probability 4*(6), 995–998.

Ripley, B. D. (1977). Modelling spatial patterns. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 39*(2), 172–212.

Scholl, T. and T. Brenner (2015). Optimizing distance-based methods for large data sets. *Journal of Geographical Systems 17*(4), 333–351.

Sweeney, S. and S. Arabadjis (2022). Spatial point patterns. In S. J. Rey and R. Franklin (Eds.), *Handbook of Spatial Analysis in the Social Sciences*, pp. 262–276. Edward Elgar Publishing.

Sweeney, S. H. and E. J. Feser (1998). Plant size and clustering of manufacturing activity. *Geographical Analysis 30*(1), 45–64.

Tidu, A., F. Guy, and S. Usai (2024). Measuring Spatial Dispersion: An Experimental Test on the *M*-Index. *Geographical Analysis 56*(2), 384–403.