This article presents a quantitative and objective approach to cat ganglion cell characterization and classification. The combination of several biologically relevant features such as diameter, eccentricity, fractal dimension, influence histogram, influence area, convex hull area, and convex hull diameter are derived from geometrical transforms and then processed by three different clustering methods (Ward’s hierarchical scheme, K-means and genetic algorithm), whose results are then combined by a voting strategy. These experiments indicate the superiority of some features and also suggest some possible biological implications.
Thus, the determination of a meaningful set of features related to the neural behavior of the different neural classes can help to segregate neurons into coherent classes [7,10,13]. These sets of features should not only be related just to the geometrical properties of the neural processes (i.e. dendrites and axons) of the neurons, but also to the types of fields of influences affecting those processes, because they are essential in order to better understand the spatial cover exhibited by cells.

The primary cells used in many morphological investigations are retinal ganglion cells, because their nearly planar dendritic arborization can be conveniently reduced to two dimensions. Indeed, important evidences of the relationship between shape and function have been verified for the specific case of cat retinal ganglion cells. Wässle [4] verified that their receptive fields are related to the convex hull area of the dendritic arborization, and related findings were verified by several researches [14–18], which defined a correspondence between the three morphological (α, β and γ) and functional (i.e. Y, X and W) groups. Of these three functional groups, the W cells are the least known.

The present work considers a group of α and β cat retinal ganglion cells and investigates several biologically relevant morphological measures that can be particularly helpful to characterize each of those groups. The objective is to try to find the best combination among the adopted features that can properly classify these neurons inside each group with the smallest possible mistake. The considered features are related to the shape and influence fields of neural cells, including fractal dimension, symmetry, diameter, eccentricity and convex hull. An analysis of the correlations between the considered features is also included.

The extraction of the visual information is a complex process that involves several different phases. One of the most common paradigms involves a hierarchy of image transformations from the low to the high levels of vision. The definition and the selection of suitable features play a fundamental role during the whole process [19,20], since this phase may affect the understanding of the image data at higher levels of abstraction. Such feature spaces can be further transformed into new ones by applying suitable operators. The two-dimensional (2D) Fourier transform is an example of image map from a spatial to a frequency space. The choice of the image transformation depends on the kind of image properties that we want to put in evidence, depending on the problem to be solved.

Geometrical entities (e.g. edges, comers and surfaces) and spatial relations are natural candidates to be used in shape classification problems (syntactic approach). However, both the computation of these entities and the evaluation of their spatial relations are often too complex and the design of the shape classifier is consequently affected by the correctness of syntactic rules that are not always standard or general. The representation of an image through numerical features allows the problem to be treated from a statistical point of view.

In the following, a quantitative approach that represents neural cells in new multi-dimensional spaces is described. It must be pointed out that the new representation space may introduce some drawbacks requiring a careful analysis of the generated features. For instance, redundancy and ambiguity may emerge, which can be partially solved by a careful choice of the features themselves. The classification phase will be performed in the new feature space by considering three different classifiers: Ward’s hierarchical grouping, K-means and genetic algorithm, integrated by a weighted vote technique [21].

Shape-based Features

Ganglion retinal cells can be mathematically understood as sets of connected points in a two or three-dimensional (3D) space F, which can be approximated in a discrete binary image space. Neuron classification performed directly on F is a hard task that could require \( O(N^2) \) comparisons, assuming that each image has \( N \) pixels.

The representation of an image can be modified by applying suitable image transformations (IT) mapping from F to a new, and typically smaller, feature space \( F' \). An IT is said to be sound if \( F' \) is invariant for translation, rotation and scaling. This property reduces the size of the search space, therefore making the classification easier. Translation invariance is usually satisfied, while scale and rotation invariance are not always fully satisfied. However, normalized features, i.e. that range in a fixed interval, are most of the time scaling invariant. Rotational invariance needs the definition of isotropic ITS. In the following, three image transform families used in the present approach are described in detail.
**Influence area transform**

Measures of the influence field of a cell, which is directly related to its size, complexity, and spatial coverage, present good potential for the identification of neural cells. A description of these features, considered in the present work, is presented in the following.

**Influence area.** The influence area (IA) indicates the area around a cell, defined by the respective dendritic arborization [7,22], which is relevant to the type of influence field under consideration. For instance, in case we are interested in synaptic contacts and know that these occur up to a maximum distance of 1 μm around any point of any dendrite, the region thus defined is called the influence area of that cell with respect to that type of interaction. A natural geometric way to represent IAs is in terms of Minkowski Sausages [22]. Given a specific shape corresponding to the contours of a 2D neural cell, it is possible to obtain, by using dilations [12,23], a series of Minkowski sausages for successive radius values, as illustrated in Figure 1 for sausage radii equal to 3, 6, 9 and 12 for each cell, respectively. Interesting feature vectors expressing the influence pattern of the cell, henceforth called influence area histograms, can therefore be obtained by estimating the area of each sausage for each considered radius. Figure 1(k) presents the influence area histogram for the images. Observe that, if necessary, it is also possible to normalize the areas by previously mapping the cells onto a fixed size box (or dividing by the diameter of the cell), in such a way as to obtain scale invariance.

**Influence histograms.** While the above-presented influence area histogram considers a uniform pattern of influence around the cell processes, it is also possible to allow the influence to be graded, leading to the concept of influence histograms (IH). Such graded influences arise naturally in several relevant neural contexts, such as the interaction of the membrane of the cell with a surrounding electric field. The IH, first proposed in [22], presents good potential for adequately characterizing the spatial covering degree occupied by cells and has been shown to be an interesting feature to classification. The first step to obtain an influence histogram is to define the point-spread function defining the type of interaction under study. For instance, in case we are interested in the electric potential around the cell, the potential around a point charge should be used as point-spread function. Once this function is defined, it is convolved with the neural shape in order to obtain the whole field of influence around the cell. Now the influence histograms are obtained by histograming the field intensities obtained around the cell.

In the current work, we consider Gaussian distributions as point-spread functions. Let \( f(x, y, z) \) be a multivariate and symmetric Gaussian distribution, function defining the distribution of an influence factor around cell. Considering that the shape of a neural cell is represented by binary data \( I(x, y, z) \), the influence histogram can be calculated by convolved \( f(x, y, z) \) with the neural cell, i.e. \( T(x,y,z) = f(x,y,z) * I(x,y,z) \), where \( T(x,y,z) \) is the resulting image. The histogram of such a resulting image, which is computed only up to a maximum distance to the shape in order to allow more uniform results, has the potential to provide relevant information about the degree of space coverage exhibited by the cells. For instance, a more complex and dense cell will tend to present a histogram with a larger average than a simple cell. The process of deriving an influence histogram is illustrated in Figure 2.

**Fractal dimension.** Complexity measures have been frequently used for shape classification, including the classification of neural cells. Boycott and Waessle [14] were one of the first to show the differences among the ganglion cells of the retina of the cat, showing that the cell complexity is an important measure to characterize these cells. One of the most used complexity measures is the fractal dimension (FD) [22,24–26]. In this paper, we considered the Box-Counting (BC) and the Minkowski sausage (Mi) [22,27] methods, but with the difference that only the portion of the log–log curve with largest fractality in the FD estimation was considered (a further extension of this concept has been described in [12]). This is done in order to compensate the problem of the limited fractality exhibited by natural objects [27]. The calculation of the FD by Box-Counting considered the average values of the numbers of boxes for inclinations of the grid (the rotated grid in different angles and the average number of boxes for all inclinations was considered [13]) so that calculation is more precise. The Minkowski sausages method is based on the fact that more complex shapes tend to produce Minkowski sausages with areas that increase less steeply. In other words, such shapes impose more constraints on the growing sausages. Computationally, the fractal dimension can be estimated by this method by first obtaining the influence area histogram as described in the on Influence area section organizing this histogram.
Figure 1. Minkowsky sausages of the neuron in (a) with respect to sausage radii equal to 3 (b), 6 (c), 9 and 12 (d), and of the neuron in (b) for the same radii used in (a). Observe that the original contour can be understood as the sausage for radius 0. The influence area histogram for the image (a) and (f) for radii 1–20 is presented in (k).
as a log–log function (i.e. logarithm of the area in terms of the logarithm of the radius), and interpolating a straight line around the region presenting the smallest derivative. The fractal dimension is taken as 2 minus the slope of this line.

**Axial moments**

An object is said to exhibit symmetry if the application of certain isometries, called symmetry operators, leaves it unchanged while parts are permuted. For instance, the
Figure 2. Influence histograms (e) obtained for the two distinct complexity neural cells (a, b). The histograms express the distributions of the graded influence fields (assuming a Gaussian point-spread function) around these cells (c, d).
letter “A” remains unchanged under reflection, the letter “H” is invariant under both reflection and half-turn, the circle, “○” has an annular symmetry around its center. Symmetry operators have already been applied to represent and describe object-parts [28] and to perform image segmentation [29].

The definition of our symmetry transform (ST) starts from the computation of a set of normalized first-order axial moments (AMs) of the gray-level intensities $g$ around the center of gravity of each cell $C$ [30]:

$$AM_k = \sum_{(m,n)\in C} |m \sin \varphi_k - n \cos \varphi_k| g(m,n),$$

where $k = 1, \ldots, K$ and $\varphi_k = k\pi/K$.

For the purpose of this work, we have set the number of axes to $K=32$, according to previous experimental results. It is easy to prove that all AMs will have the same value in the case of an isotropic density distribution, hence they can be used as a circular symmetry measure:

$$ST = 1 - \sqrt{\frac{\sum_k AM_k^2}{K} - \left(\frac{\sum_k AM_k}{K}\right)^2}.$$

The ST is invariant for translation, rotation and scaling; therefore it is sound. The first property derives directly from the analytical definition of the AMs. The second property derives from the fact that object rotation implies the circular permutation of the AMs. The third property is satisfied thanks to the normalization.

An example of the AM feature is shown in Figure 3. It is interesting to note that all AMs can be quickly computed in parallel as a sum of convolution filters. This approach can also be generalized for 3D object recognition [31].

Three shape parameters are derived from the AMs: their mean value (MV), their standard deviation (SD),
and their eccentricity \((EC = \frac{AM_{\text{min}}}{AM_{\text{max}}})\), where \(AM_{\text{min}}\) and \(AM_{\text{max}}\) are the minimum and maximum values of the AMs. The MV and SD features are normally distributed in first approximation. We assumed it to be true also in the case of the feature EC which is the ratio of two normally distributed quantities. Table 1 reports the mean \((\mu)\) and standard deviation \((\sigma)\) parameters, while Figure 4 shows the probability distribution.

**Convex hull**

The size of a neural cell provides an important resource for its identification, being often used by neuroscientists as one of the most important features for the classification of neurons. The size of a shape can be defined in several ways, including the convex hull and the shape diameter, which are addressed in the following.

**Convex hull area.** The convex hull (CA) is considered as the smallest convex polygon that totally contains a specific neuron. It has been verified experimentally [4] that the receptive field of a cell can be predicted with good precision as being approximately 25\% larger in area than the convex hull of the dendritic arborization (Figure 5).

**Diameter.** The diameter \((Di)\) is another feature used to globally characterize the neuron size and spatial influence. It can be obtained from the CA of the cell as the largest distance between any of its two points, as illustrated in Figure 5.

**The Classifiers**

Since we are interested in investigating how the morphological cell classes are defined, an unsupervised classification has been adopted. Three classification

---

**Table 1. Mean \((\mu)\) and standard deviation \((\sigma)\) of MV, SD, EC**

<table>
<thead>
<tr>
<th>Feature</th>
<th>(\mu)</th>
<th>(\sigma)</th>
<th>(\mu)</th>
<th>(\sigma)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MV</td>
<td>0.936</td>
<td>0.26</td>
<td>0.026</td>
<td>0.081</td>
</tr>
<tr>
<td>SD</td>
<td>0.007</td>
<td>0.003</td>
<td>0.017</td>
<td>0.011</td>
</tr>
<tr>
<td>EC</td>
<td>0.979</td>
<td>0.010</td>
<td>0.942</td>
<td>0.039</td>
</tr>
</tbody>
</table>

---

Figure 4. Probability distribution for MV, SD, and EC.
algorithms (Ward’s hierarchical grouping (WG), K-means clustering (KM) and genetic algorithm (GA)) have been used and are briefly described in the following.

This work has considered a set of 50 cat retinal ganglion cells [14–16,32–36], 23 being of α type and 27 of β type. All these cells present distance to the fovea smaller than 3° and were normalized such that they all present the same diameter.

The New Feature Space

Figure 6 describes the combination of the transform introduced above as a tree, where the root is the input feature space F, the leaves represent the new features of F, and the internal nodes are the IT’s.

The features considered here are listed in Table 2. They are related to measures normally used to biologically classify these cells. The subscripts of the IH feature indicate the box number in the histograms. We have considered a histogram with 11 boxes (numbered 1–11). The subscripts of the IA feature indicate the radius of each sausage considered. We have considered 20 areas of the sausage (radii varying 1–20).

Ward’s hierarchical grouping

Ward’s hierarchical grouping, which is one of the most frequently adopted clustering methods, relies on progressively agglomerating the data in such a way as to minimize the dispersion (variance) inside each group [37,38]. It tends to produce clusters that are easily distinguished from other clusters and which tend to be tightly packed. It operates by reducing the number of clusters one at a time starting from one cluster per object and ending with one cluster comprising all the objects. At each cluster reduction, the method merges two or more objects. The confusion matrices of the WG classification are given in Table 3.

K-means clustering

K-means clustering involves an iterative scheme that operates over a fixed number of clusters, such that each class has a center which is the mean position of all the samples in that class and each sample is in the class whose center is closest to it [39]. The confusion matrices of the KM classification are given in Table 3.

Genetic algorithm clustering

The genetic algorithm clustering [40] is a variant of ISODATA [41]. It shows a faster convergence to the solution by considering the clustering as a global optimization problem, where the optimum solution corresponds to the minimization of the within-cluster scatter matrix (maximization of the between-cluster

<table>
<thead>
<tr>
<th>IT</th>
<th>Morphological feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>IA</td>
<td>BC</td>
</tr>
<tr>
<td>IA</td>
<td>Mi</td>
</tr>
<tr>
<td>IA</td>
<td>IH&lt;sub&gt;1&lt;/sub&gt;–11</td>
</tr>
<tr>
<td>IA</td>
<td>IA&lt;sub&gt;1&lt;/sub&gt;–20</td>
</tr>
<tr>
<td>ST</td>
<td>AM</td>
</tr>
<tr>
<td>ST</td>
<td>MV</td>
</tr>
<tr>
<td>ST</td>
<td>SD</td>
</tr>
<tr>
<td>ST</td>
<td>EC</td>
</tr>
<tr>
<td>CH</td>
<td>CA</td>
</tr>
<tr>
<td>CH</td>
<td>Di</td>
</tr>
</tbody>
</table>
scatter matrix) [42]. Initially, the sampling strategy allows us to span the solution space in a random way. The evolution is fast driven (on average by 123 iterations) toward an optimal (or near-optimal) solution, according to the schema theorem [43].

The feature vector of each cell and its cluster-label are coded through a chromosome of 24 and 8 bits, respectively. This method starts by generating an initial population of such chromosomes and then evolves these binary strings by crossover (replacing with a probability of 90% two of them by a pair of their offspring) and by mutation (altering with a probability of 1% some of their bits). The fitness function is based on the Euclidean distance computed between each chromosome and the center of gravity of each cluster. The evolution allows to rearrange the assignment of the cells by minimizing the internal variance of each class. The selection of the new population is performed by using the ranking technique. The confusion matrices of the GA classification have been reported in Table 3.

Table 3. Confusion matrices of the classifiers with respect to their best results

<table>
<thead>
<tr>
<th>Feature sets</th>
<th>K-means</th>
<th>Ward</th>
<th>Genetic algorithm</th>
<th>Vote strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. (AM, MV, BC, Mi, IH)</td>
<td>[19 4]</td>
<td>[22 1]</td>
<td>[22 1]</td>
<td>[22 1]</td>
</tr>
<tr>
<td>2. (AM, MV, Mi, IH)</td>
<td>[21 2]</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[22 1]</td>
</tr>
<tr>
<td>3. (AM, BC, CA, Di)</td>
<td>[22 1]</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[22 1]</td>
</tr>
<tr>
<td>4. (AM, SD, EC, BC, IH)</td>
<td>[22 1]</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[22 1]</td>
</tr>
<tr>
<td>5. (SD, EC, BC, Mi, IH)</td>
<td>[22 1]</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[22 1]</td>
</tr>
<tr>
<td>6. (AM, MV, BC, CA, Di)</td>
<td>[23 0]</td>
<td>[18 5]</td>
<td>[23 0]</td>
<td>[23 0]</td>
</tr>
<tr>
<td>7. (AM, Di)</td>
<td>[23 0]</td>
<td>[18 5]</td>
<td>[23 0]</td>
<td>[23 0]</td>
</tr>
<tr>
<td>8. (AM, MV, CA, Di)</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[23 0]</td>
<td>[23 0]</td>
</tr>
<tr>
<td>9. (MV, SD, BC, Mi, IH)</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[23 0]</td>
<td>[23 0]</td>
</tr>
<tr>
<td>10. (MV, SD, EC, Mi, IH)</td>
<td>[21 2]</td>
<td>[20 3]</td>
<td>[23 0]</td>
<td>[23 0]</td>
</tr>
<tr>
<td>11. (AM, MV, SD, EC, BC, Mi, CA, Di, IH)</td>
<td>[20 3]</td>
<td>[20 3]</td>
<td>[22 1]</td>
<td>[20 3]</td>
</tr>
</tbody>
</table>

The Classifier Integration

The above-described classifiers can be combined to improve the classification accuracy. The main motivation for considering combined technique is that human decision models also use more than one evaluation paradigm, and usually a complex decision is taken by more than one expert [21,44,45].

The integration of decisions is becoming a very powerful approach in data analysis systems. The application of a class of classifiers $Cl = \{Cl_1, Cl_2, \ldots, Cl_s\}$ on a given data set $X$ produces a set of partitions $M = \{M_1, M_2, \ldots, M_s\}$, where $M_i$ is the confusion matrix of the $i$th classifier $Cl_i$. An indicator $\pi_i$, related to the accuracy of the method, can be assigned to each partition $M_i$, thus defining the distribution $\Pi_{Cl} = \{\pi_1, \pi_2, \ldots, \pi_s\}$ with $0 < \pi_i < 1$. $\Pi_{Cl}$ can be evaluated on the basis of the judgment of an expert or by means of a calibration procedure during the training phase; sometimes a theoretical evaluation of the
accuracy of a classifier can be used. Intuitively, \( \Pi_i \) weights the goodness of \( C_l_i \). Note that, without a priori knowledge, \( p_i = 1/s \).

Integrated decision techniques can be described by using graph-theoretical approaches. Graphs are weighted and directed; labeled nodes represent: the classifiers \( C_l_i \), the input dataset \( X \) and the partition \( M \). The topology of the graph is related to the decision strategies adopted. In particular, we adopted the parallel topology shown in Figure 7 because our classifiers do not exchange any information during their computation. Each classifier performs the computation independently: it receives the two inputs \( (X, \Pi_i C_l) \) and produces the result \( M \).

The evolution of the integrated classifier is determined by the following system:

\[
M = G(M_1, M_2, \ldots, M_s, \pi_1, \pi_2, \ldots, \pi_s)
\]
\[
\pi = m(\pi_1, \pi_2, \ldots, \pi_s)
\]

where the functions \( G \) and \( m \) are usually polynomials. For example,

\[
M = \frac{\sum \pi_i M_i}{\sum \pi_i}, \quad \pi = \frac{1}{s} \sum \pi_i
\]

The parameter \( \pi \) is usually considered as a global measure of performance. In our case, the values of \( \pi_i \) are computed from

\[
\pi_i = \frac{(M_{xx})_i + (M_{yy})_i}{50}
\]

since we studied 50 cells. Each entry of \( M_i \) indicates the agreement between two methods. This allows to compare the decision of the classifier \( C_l_i \) against a human expert: a perfect agreement holds if \( M_i \) is diagonal. Table 4 reports \( \pi_i \) for the three classifiers we used. The weight of each classifier has been derived on the basis of its accuracy. In order to obtain the most correct grading, we applied a voting strategy among the results provided by the classifiers. In particular, a given cell \( x \) is assigned to the class \( \alpha \) if \( \sum x_i \pi_i / \sum \pi_i < 0.5 \), to class \( \beta \) otherwise. This formula is a weighted mean, where \( x_i \) represents the assignment by the classifier \( C_l_i \) of \( x \) to \( \alpha (x_i = 0) \) or to \( \beta (x_i = 1) \).

**Methodology**

In order to validate the potential of the adopted measures for the characterization and classification of neural cells, 50 images of cat retinal ganglion cells [14–16,33–36,39] of types \( \alpha \) and \( \beta \), all presenting distance of the fovea smaller than 3', have been considered.

Influence histograms assuming a Gaussian profile with a standard deviation of 4, and containing 11 uniformly distributed bins, were generated for each image. Influence areas with \( \text{Dist} \) varying from 1 to 20 are also considered for each neuron. In order to investigate the influence of the size of the cells, the diameter of each cell was obtained from the respective convex hulls of the cells. To measure the complexity of each image, the fractal dimension was calculated by using both Box Counting and Minkowski sausage methods. Note that to improve the clustering results, every feature was normalized through a statistical transformation in such a way as to present null average and unitary standard deviation.

The calculation of central and axial moments has introduced new parameters with complementary information as compared to those obtained by other shape indicators. Moreover, their soundness allowed us to improve the discrimination ratio of the two types of cells.
Results and Discussion

The linear interrelationship between neural features can be analyzed by the correlation matrix shown in Table 5. This table shows only three out of the 11 bins of influence histogram and five out of the 20 bins of influence area. It should be recalled that value 1 indicates a perfect correlation among two features, which only happens, in this case, for the same features. A negative value indicates an anti-correlation. Marked correlations (absolute values larger than 0.5) are considered strong. Classifications have been performed using all possible combinations of 2, 3, 4, 5 and 9 features, as well as by just using one and all features. All classification methods have been applied for such combinations. Although IA has presented a strong correlation with many other features, it turned out to be the worst feature. Alone or combined with other features, it always tends to undermine the classification. Therefore, it is excluded from the following comments.

The features were grouped into the three major features groups: (1) symmetry features (AM, MV, SD, and EC); (2) complexity features (BC and Mi); and (3) size features (CA, Di, and IH), that considered the size of the neuron. Except for group (Di, IH), that presented satisfactory results (error ≤ 10% by all classification methods), any combination using only features of the same groups led to a poor result. This is a consequence of the fact that both features are related to the cell size (mainly Di), one of the features used by neuroscientist to classify neural cells. The group (AM, Di) showed as being suitable for the classification (presented error equal to 6%). This was the only group with two or three features that presented such a small error. Another interesting result was obtained by using all the features, except IA. The error presented was 6%.

Most of the best classifications were obtained using groupings of 4 or 5 features together. Good results were obtained when we took at least two symmetry features, at least one complexity feature and IH. The IH combined with others also showed to be a good feature.

Most of the worst results were obtained for groups of 1, 2 or 3 features. In such cases, the groups contained only symmetry features, or symmetries and complexity or still only a symmetry feature. The error in these cases was larger than 20%.

The number of cells assigned to each class with respect to the best classification results is shown in Table 3. In all these cases, there was at least 1 wrongly classified cell.

It should also be observed that the relatively high misclassification rates obtained are very likely explained by the fact that the class assignment used as a comparison standard was produced subjectively by diverse authors, using different criteria.

Conclusions

A quantitative approach to neural cell classification, concentrating on cat ganglion cells, has been reported considering several geometrical features, and their combinations have been investigated with respect to three clustering algorithms. The obtained results indicated the superiority of some specific features, as well as a few incompatibilities with the original classifications.
Table 5. Correlation matrix among the features considered. The PHASES considered N = 50 (Casewise deletion of missing data).

<table>
<thead>
<tr>
<th>AM</th>
<th>MV</th>
<th>SD</th>
<th>EC</th>
<th>BC</th>
<th>Mi</th>
<th>CA</th>
<th>Di</th>
<th>IH1</th>
<th>IH6</th>
<th>IH11</th>
<th>IA1</th>
<th>IA5</th>
<th>IA10</th>
<th>IA15</th>
<th>IA20</th>
<th>IA25</th>
</tr>
</thead>
<tbody>
<tr>
<td>AM</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MV</td>
<td>1.0000</td>
<td>0.8133</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SD</td>
<td>0.5913</td>
<td>0.7140</td>
<td>0.9905</td>
<td>1.0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EC</td>
<td>0.6594</td>
<td>0.7897</td>
<td>0.9905</td>
<td>1.0000</td>
<td>0.7307</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BC</td>
<td>0.7347</td>
<td>0.8536</td>
<td>0.6901</td>
<td>0.9905</td>
<td>1.0000</td>
<td>0.7307</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mi</td>
<td>0.1694</td>
<td>0.0046</td>
<td>0.1239</td>
<td>0.0046</td>
<td>0.7307</td>
<td>1.0000</td>
<td>0.9905</td>
<td>1.0000</td>
<td>0.7307</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CA</td>
<td>0.4585</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.1239</td>
<td>0.9905</td>
<td>1.0000</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Di</td>
<td>0.8782</td>
<td>0.7752</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IH1</td>
<td>0.4200</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IH6</td>
<td>0.3069</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IH11</td>
<td>0.4069</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IA1</td>
<td>0.4492</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
<td></td>
</tr>
<tr>
<td>IA5</td>
<td>0.4253</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>0.9905</td>
</tr>
<tr>
<td>IA10</td>
<td>0.4470</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>IA15</td>
<td>0.4510</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>IA20</td>
<td>0.4610</td>
<td>0.1694</td>
<td>0.6832</td>
<td>0.5170</td>
<td>0.9905</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

References