Classifying GABAergic interneurons with semi-supervised projected model-based clustering

Bojan Mihaljević , Ruth Benavides-Piccione , Luis Guerra , Javier DeFelipe , Pedro Larrañaga , Concha Bielza

A B S T R A C T

Objectives: A recently introduced pragmatic scheme promises to be a useful catalog of interneuron names. We sought to automatically classify digitally reconstructed interneuronal morphologies according to this scheme. Simultaneously, we sought to discover possible subtypes of these types that might emerge during automatic classification (clustering). We also investigated which morphometric properties were most relevant for this classification.

Materials and methods: A set of 118 digitally reconstructed interneuronal morphologies classified into the common basket (CB), horse-tail (HT), large basket (LB), and Martinotti (MA) interneuron types by 42 of the world's leading neuroscientists, quantified by five simple morphometric properties of the axon and four of the dendrites. We labeled each neuron with the type most commonly assigned to it by the experts. We then removed this class information for each type separately, and applied semi-supervised clustering to those cells (keeping the others' cluster membership fixed), to assess separation from other types and look for the formation of new groups (subtypes). We performed this same experiment unlabeling the cells of two types at a time, and of half the cells of a single type at a time. The clustering model is a finite mixture of Gaussians which we adapted for the estimation of local (per-cluster) feature relevance. We performed the described experiments on three different subsets of the data, formed according to how many experts agreed on type membership: at least 18 experts (the full data set), at least 21 (73 neurons), and at least 26 (47 neurons).

Results: Interneurons with more reliable type labels were classified more accurately. We classified HT cells with 100% accuracy, MA cells with 73% accuracy, and CB and LB cells with 56% and 58% accuracy, respectively. We identified three subtypes of the MA type, one subtype of CB and LB types each, and no subtypes of HT (it was a single, homogeneous type). We got maximum (adapted) Silhouette width and ARI values of 1, 0.83, 0.79, and 0.42, when unlabeling the HT, CB, LB, and MA types, respectively, confirming the quality of the formed cluster solutions. The subtypes identified when unlabeling a single type also emerged when unlabeling two types at a time, confirming their validity. Axonal morphometric properties were more relevant that dendritic ones, with the axonal polar histogram length in the $[\pi, 2\pi)$ angle interval being particularly useful.

Conclusions: The applied semi-supervised clustering method can accurately discriminate among CB, HT, LB, and MA interneuron types while discovering potential subtypes, and is therefore useful for neuronal classification. The discovery of potential subtypes suggests that some of these types are more heterogeneous that previously thought. Finally, axonal variables seem to be more relevant than dendritic ones for distinguishing among the CB, HT, LB, and MA interneuron types.

1. Introduction

GABAergic interneurons of the cerebral cortex are key elements in many aspects of cortical function in both health and disease. Nevertheless, the classification of GABAergic interneurons is a difficult task and has been a topic of debate for a long time, since the

* Corresponding author. Tel.: +34 91 3363675; fax: +34 91 3524819.
  E-mail addresses: bmihaljevic@fi.upm.es (B. Mihaljević), rbp@cajal.csic.es (R. Benavides-Piccione), luispelayo84@gmail.com (L. Guerra), defelipe@cajal.csic.es (J. DeFelipe), mcbielza@fi.upm.es (C. Bielza).

pioneering work of Santiago Ramón y Cajal on the characterization and identification of interneurons [1]. The difficulty stems from the high variability of these cells according to morphological, electrophysiological and molecular features [2]. The scientific community lacks an accepted catalog of neuron names [3] which makes it difficult to organize and share knowledge [2]. There is some agreement on the set of morphological, molecular, and physiological features that can be used to distinguish among types of GABAergic interneurons [2]. However, a comprehensive classification according to those features is difficult to perform in practice [3]. A recent experiment enabled 42 expert neuroscientists from all around the world to classify interneurons by visual inspection and according to pre-selected neuron names [3]. It showed that the experts agree on the morphological definitions of some of the pre-selected types while disagreeing on the definitions of others. In particular, some types seemed to overlap in terms of the cells that were assigned to them by the experts. In [3], the authors also showed that supervised classification models can automatically categorize interneurons in accordance with the opinion of the majority of the experts.

Automatic classification of interneurons has mainly been done with (unsupervised) clustering; see, e.g., [4–8]. However, supervised approaches can be more accurate when there is prior knowledge about neuronal types [9]. In this study, such knowledge comes from the experts who participated in the experiment described in [3]. We can use this knowledge to guide classification and simultaneously discover subtypes using semi-supervised clustering, an approach that lies between the supervised and unsupervised approaches. In doing this, we follow the *cluster assumption* [10], i.e., we consider that the instances within a cluster are likely to belong to the same class whereas a class may consist of several clusters. In semi-supervised learning [10,11], some data instances are labeled whereas others are not. Since all our neurons were labeled by the experts, we fitted the semi-supervised scenario by removing the labels of (a) one type at a time; (b) two types at a time; and (c) half the instances of each type, simultaneously. By doing this we sought to discover possible subtypes and see if the types could be automatically discriminated. We used an adaptation of the semi-supervised projected model-based clustering algorithm (SeSProC) introduced in [12]. This is a probabilistic clustering algorithm which estimates the number of clusters and the relevance of each predictive feature for each of the clusters. The estimation of feature relevance within model-based clustering was introduced in [13].

We quantified the neurons with nine simple axonal and dendritic morphological variables, such as the axonal length close to the soma, and labeled them according to the choices of the expert neuroscientists. In [3] each instance was given up to 42 labels—coming from the 42 experts that concluded the study. Following a common practice in supervised learning [14], we reduced this vector of 42 labels to its mode (i.e., the most common value), thus obtaining a single label per neuron. However, since experts frequently disagreed, such labels were often not reliable, i.e., they were backed by few experts. To cope with the *label noise* [15,16] that expert disagreements may be introducing, we analyzed three subsets of our neuron population, each with a different minimum of 'label reliability', i.e., such that the label of each neuron in the subset was agreed upon by at least *th* experts, with *th* being a 'label reliability threshold'.

This paper is an extension of [17] and is the result of close collaboration between experts in neuroanatomy and machine learning.[1] We extend the mentioned paper by refining some of the predictor variables, adapting the SeSProC algorithm, and considering two

---

Table 1

Distribution of interneuron types with respect to label reliability threshold. Lowermost row shows total number of cells per dataset.

	th18	*th22*	*th26*
CB	49	24	9
HT	9	5	4
LB	27	19	12
MA	33	25	22
Total	118	73	47

additional experimental settings. The remainder of this paper is organized as follows: Section 2 describes the materials and methods we used; Section 3 reports and discusses the obtained results; while Section 4 provides conclusions.

2. Materials and methods

2.1. Data

We used 237 three-dimensional (3D) reconstructions of interneurons from several areas and layers of the cerebral cortex of the mouse, rat, and monkey. These neurons were used in [3], and were originally extracted from NeuroMorpho.Org [18]. From this population of neurons, we formed subsets by imposing minimums on the number of experts that agreed on the label of an included cell (i.e., a 'label reliability threshold'), considering that a higher threshold yields more confidence in the cells' labels. We used thresholds 18, 22 (half plus one out of the 42 experts), and 26 to build three databases: *th18*, *th22*, and *th26*, respectively. These data sets contained interneurons of four different types (classes): common basket (CB), horse-tail (HT), large basket (LB), and Martinotti (MA). Table 1 shows the distribution of different types at the three label reliability thresholds.

We characterized each neuron using nine features of axonal and dendritic morphology. While one may compute many morphological features (e.g., [3] used over 2000 features for classification), none are known, so far, as good predictors of interneuron type. Since detailed morphometric information on 3D reconstructed cortical interneurons is relatively scarce (a few hundred reconstructed neurons are available, comprising different types), it might be counterproductive to use many predictor variables. Therefore, we kept the number of variables low by defining variables which capture how, in our opinion, an expert classifies an interneuron upon visual examination.

We consider that an expert classifies an interneuron by estimating the distribution and the orientation of axonal and dendritic arborizations. We therefore measured the axonal and dendritic length according to the Sholl (5 features) and polar histogram (4 features) analyses from NeuroExplorer, the data analysis companion to Neurolucida [19]. Sholl analysis computes axonal and dendritic length at different distances from the soma whereas the polar histogram [20] describes the overall direction of dendritic growth; we only distinguished between two halves of the histogram, namely, the bifurcation angles falling in the $[0, \pi)$ interval and those falling in the $[\pi, 2\pi)$ interval. See Table 2 and Fig. 1 and for further details on predictor variables. We standardized all variables (transformed them so to have zero mean and unit standard deviation) prior to classification.

While an expert who classifies using a similar rationale can only roughly estimate these features, our classifier used exact values, thus possibly being more objective. This is important as some of the features that we use, such as the length of the axonal arbor at a certain distance from the soma, are rather hard for an expert to estimate.
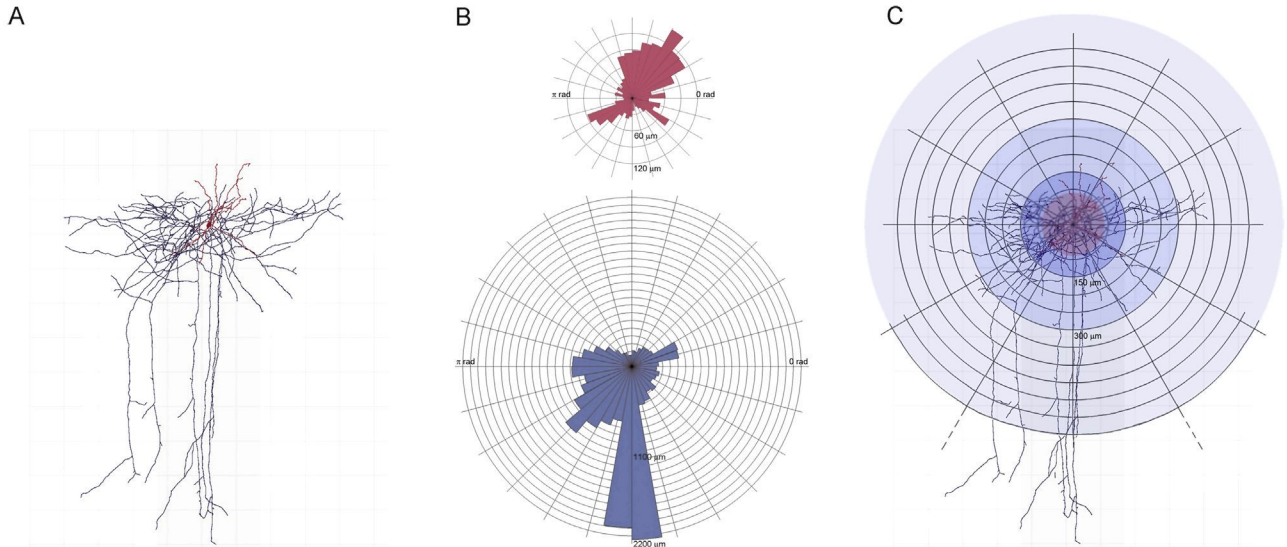
Fig. 1. A schematic representation of the nine morphological features of axonal and dendritic morphology that we used as predictor variables. (A) An example of a 3D reconstructed interneuron classified by expert neuroscientists in [3], showing its axonal (displayed in blue) and dendritic (red) arborizations. The gray vertical shadow indicates the extent of the cortical column (assumed to be 300 μm wide) whereas the dimensions of the squares are $100\,\mu m \times 100\,\mu m$. (B) Schematic representation of the overall direction of dendritic growth (polar histogram) for the dendrites (above; red) and the axon (below; blue). Features X_1 and X_2 encode axonal growth length in the angle intervals $[0, \pi)$ and $[\pi, 2\pi)$, respectively, whereas features X_6 and X_7 capture the dendritic growth length in the same angle intervals. (C) Schematic representation of the features encoding axonal and dendritic arborization lengths at different distances from the soma (Sholl analysis): features X_3, X_4, and X_5 encode, respectively, axonal lengths at 0–150 μm (marked with a dark blue circle), 150–300 μm (blue circle), and over 300 μm (outside of the blue circle) from the soma. Features X_8 and X_9 measure dendritic arborization at 0–90 μm (pink circle) and over 90 μm (outside of the pink circle) from the soma, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

2.2. Semi-supervised projected model-based clustering

We used the semi-supervised projected model-based clustering (SeSProC) algorithm, introduced in [12]. This algorithm allows clusters to exist in different feature subspaces, estimating the relevance of each feature for every cluster. We modified SeSProc's definition of feature relevance and its heuristic for initializing new clusters (see below for details on both modifications). SeSProC handles partially labeled data (i.e., some instances are labeled whereas others are not) by assigning unlabeled instances into either the a priori known clusters (given by the class labels) or to new clusters that it may have discovered.

Table 2
Predictor variables used in the present study. Predictors X_1–X_5 correspond to the axon whereas X_6–X_9 correspond to the dendrites.

Variable	Arbor type	Description
X_1	Axon	Polar histogram length (μm) for the $[0, \pi)$ radians interval
X_2		Polar histogram length (μm) for the $[\pi, 2\pi)$ radians interval
X_3		Sholl analysis length (μm) at less than 150 μm from the soma
X_4		Sholl analysis length (μm) at more than 150 and less than 300 μm from soma
X_5		Sholl analysis length (μm) at more than 300 μm from the soma
X_6	Dendrites	Polar histogram length (μm) for the $[0, \pi)$ radians interval
X_7		Polar histogram length (μm) for the $[\pi, 2\pi)$ radians interval
X_8		Sholl analysis length (μm) at less than 90 μm from the soma
X_9		Sholl analysis length (μm) at more than 90 μm from the soma

2.2.1. A finite mixture model with local feature selection

Let $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be observed data, with $\mathbf{x}_i \in \mathbb{R}^F, \forall i \in \{1, \ldots, N\}$, where F denotes the number of features. In model-based clustering, we assume that the data are generated from a finite mixture of K components. The density function for an instance $\mathbf{x}_i$ is

$$f(\mathbf{x}_i \mid \Theta) = \sum_{m=1}^{K} \pi_m f(\mathbf{x}_i \mid \boldsymbol{\theta}_m),$$

with $\pi_m \in [0, 1]$ and $\sum_{m=1}^{K} \pi_m = 1$, and $\boldsymbol{\theta}_m$ being the parameters of component m, with $\Theta = \{\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_K, \pi_1, \ldots, \pi_K\}$.

SeSProC encodes the clustering solution with a latent K-dimensional binary variable $\mathcal{Z}$, with $z_{im} = 1$ if $\mathbf{x}_i$ is assigned to component m, and $z_{im} = 0$ otherwise. In a semi-supervised setting, we observe the class labels c_i, $c_i \in \{1, \ldots, C\}$, of some instances $i \in \{1, \ldots, L\}$, $L < N$. We use the class labels to set the minimum number of components in the mixture, with one component per class label (i.e., $K = C$). We also use the labels to fix $z_{im} \equiv \mathbb{I}(c_i = m)$, where $\mathbb{I}(\cdot)$ returns 1 if $(\cdot)$ is true and 0 otherwise, for $i < L$. This means that the algorithm does not change z_{im} for the labeled $\mathbf{x}_i$—they always belong to clusters specified by their labels; the goal is to cluster the unlabeled instances $\mathbf{x}_i$, $L < i \leq N$.

Localized feature selection. Not all predictor variables are necessarily relevant for clustering. Following [13], we assume that a feature j is irrelevant if its distribution is independent on the components.[2] Such a feature then follows some distribution $f(\cdot \mid \lambda_j)$, independent of any particular component, instead of following a distribution $f(\cdot \mid \theta_{mj})$, dependent on a component m. SeSProc then models the distribution of a feature X_i as a mixture of $f(\cdot \mid \lambda_j)$ and

---

[2] This differs from the original SeSProC method which follows the notion of irrelevance given in [21]. Also, our model differs from the one used in [13] in that it considers the per-component relevance of each feature.

$f(\cdot \mid \theta_{mj})$. Assuming that the features are independent given the (hidden) mixture component, this transforms the density function into

$$f(\mathbf{x}_i \mid \Theta) = \sum_{m=1}^{K} \pi_m \prod_{j=1}^{F} (\rho_{mj} f(x_{ij} \mid \theta_{mj}) + (1 - \rho_{mj}) f(x_{ij} \mid \lambda_j)),$$

where ρ_{mj} is the probability of feature j being relevant for component m. ρ is encoded with an additional latent variable $\mathcal{V}$, with $v_{mj} = 1$ if feature j is relevant for component m and $v_{mj} = 0$ otherwise, and $\rho_{mj} = p(v_{mj} = 1)$. The full parameter set of the finite mixture of K components is then $\Theta = \{\theta_{mj}, \lambda_j, \rho_{mj}, \pi_m\}_{m=1,\dots,K; j=1,\dots,F}$.

2.2.2. EM for mixture learning

If $\mathcal{Z}$ and $\mathcal{V}$ were observed, the log-likelihood function would be

$$\log \mathcal{L}(\Theta \mid \mathcal{X}, \mathcal{Z}, \mathcal{V}) = \sum_{i=1}^{N} \sum_{m=1}^{K} \left(z_{im} \log \pi_m + \sum_{j=1}^{F} (z_{im}[v_{mj}(\log \rho_{mj} + \log f(x_{ij} \mid \theta_{mj})) + (1 - v_{mj})(\log(1 - \rho_{mj}) + \log f(x_{ij} \mid \lambda_j))]) \right).$$

However, they are not observed, and we cannot maximize the log-likelihood function analytically. Instead, we approximate the solution with the iterative expectation-maximization (EM) algorithm [22]. The EM produces a sequence of parameter estimates by alternating an E-step—'filling in' the missing observations with their expected values given the current parameters, and an M-step —using the completed data to compute maximum likelihood estimates of the parameters. At a given iteration t of the EM procedure, the expectation of the log-likelihood function is given by

$$\mathbb{E}_{\mathcal{Z},\mathcal{V}|\mathcal{X},\Theta^{t-1}}[\log \mathcal{L}(\Theta^{t-1} \mid \mathcal{X}, \mathcal{Z}, \mathcal{V})] = \sum_{i=1}^{N} \sum_{m=1}^{K} \gamma(z_{im}) \log \pi_m + \sum_{i=1}^{N} \sum_{m=1}^{K} \sum_{j=1}^{F} \gamma(z_{im}) \gamma(v_{mj})(\log \rho_{mj} + \log f(x_{ij} \mid \theta_{mj}))$$

$$+ \sum_{i=1}^{N} \sum_{m=1}^{K} \sum_{j=1}^{F} \gamma(z_{im})(1 - \gamma(v_{mj}))(\log(1 - \rho_{mj}) + \log f(x_{ij} \mid \lambda_j)),$$

where Θ^{t-1} are the parameters from iteration $t - 1$ and $\gamma()$ is the expectation function.

Assuming that both $f(\cdot \mid \theta_{mj})$ and $f(\cdot \mid \lambda_j)$ are Gaussian distributions, the parameters are updated in the M-step by maximum likelihood, as follows:

$$\pi_m = \frac{\sum_{i=1}^{L} z_{im} + \sum_{i=L+1}^{N} \gamma(z_{im})}{N},$$

$$\rho_{mj} = \frac{\sum_{i=1}^{N} \gamma(z_{im}) \gamma(v_{mj})}{\sum_{i=1}^{L} z_{im} + \sum_{i=L+1}^{N} \gamma(z_{im})},$$

$$\mu_{\theta_{mj}} = \frac{\sum_{i=1}^{N} \gamma(z_{im}) \gamma(v_{mj}) x_{ij}}{\sum_{i=1}^{N} \gamma(z_{im}) \gamma(v_{mj})},$$

$$\sigma_{\theta_{mj}}^2 = \frac{\sum_{i=1}^{N} \gamma(z_{im}) \gamma(v_{mj})(x_{ij} - \mu_{\theta_{mj}})^2}{\sum_{i=1}^{N} \gamma(z_{im}) \gamma(v_{mj})},$$

$$\mu_{\lambda_j} = \frac{\sum_{i=1}^{N} \sum_{m=1}^{K} \gamma(z_{im})(1 - \gamma(v_{mj})) x_{ij}}{\sum_{i=1}^{N} \sum_{m=1}^{K} \gamma(z_{im})(1 - \gamma(v_{mj}))},$$

$$\sigma_{\lambda_j}^2 = \frac{\sum_{i=1}^{N} \sum_{m=1}^{K} \gamma(z_{im})(1 - \gamma(v_{mj}))(x_{ij} - \mu_{\lambda_j})^2}{\sum_{i=1}^{N} \sum_{m=1}^{K} \gamma(z_{im})(1 - \gamma(v_{mj}))},$$

for $m = 1, \dots, K; j = 1, \dots, F$.

2.2.3. Estimating the number of components

Assuming that we have C classes in the data, the initial mixture contains C components. Its parameters are estimated with the EM procedure and its quality evaluated (see below) and recorded. Then, in the next iteration of SeSProC, this mixture is augmented with an additional component (see below), its parameters again estimated with EM and its quality computed and recorded. The process is repeated until a mixture $\mathcal{M}^K$ with K components is better than a mixture $\mathcal{M}^{K+1}$, and yields $\mathcal{M}^K$ as the final model. We evaluate the quality of a model with the AIC score [23],

$$AIC = -2 \log \mathcal{L} + 2R,$$

where R, the number of parameters in a model, is a function of the number of components K and the number of features F:[3]

$$R = 2KF + 2F + (K - 1) + KF.$$

Additionally, we halt mixture augmentation if $\mathcal{M}^K$ contains a component m with less than two instances (i.e., such that m is the most likely component for less than two instances), returning $\mathcal{M}^{K-1}$, unless $K = C$, in which case $\mathcal{M}^K$ is returned.

When starting the EM procedure for a model $\mathcal{M}^K$, we use the class labels to initialize the θ parameters for the classes, i.e., we estimate $\boldsymbol{\theta}_m = (\theta_{m1}, \dots, \theta_{mF})$, for $m \leq C$, from instances belonging to class m. If $K > C + 1$ (i.e., if there are already newly found components in the mixture), then we use the $\boldsymbol{\theta}_m$, $C < m < K$, from $\mathcal{M}^{K-1}$ as their initial values in $\mathcal{M}^K$, since there are no labels that could guide the estimation of their initial values. The initial $\boldsymbol{\theta}_K$ for the new component K are estimated from a number of unlabeled data points; the following paragraph describes how these data points are selected.

Initializing a new component K. We have modified SeSProC to use the following heuristic for initializing new components. Starting from the previous mixture, $\mathcal{M}^{K-1}$, we consider the neighborhood of each unlabeled point as (a part of) a potential new cluster. Thus, we take the ct nearest unlabeled neighbors (according to the Euclidean distance in full dimensionality), where ct is a parameter of the algorithm, to a point $\mathbf{x}_i$ and assign them to a new cluster, by setting $z_{jK} = 1$ and $z_{jm} = 0$, $m \neq K$ for all $\mathbf{x}_j$ in the neighborhood of $\mathbf{x}_i$; we then update all parameters (this includes the $\boldsymbol{\theta}_K$) by maximum likelihood (i.e., with an M-step) and compute the likelihood of the thereby obtained model. The neighborhood that yields the most likely model is then used to initialize $\boldsymbol{\theta}_K$ in the new model $\mathcal{M}^K$. All assignments to $\mathcal{Z}$ here described are then undone after $\boldsymbol{\theta}_K$ is initialized (i.e., these assignments were only temporary).

---

[3] The first addend in the formula for R corresponds to θ's, second to λ's, third to π's and fourth to ρ's.

2.3. Empirical setup

We tested SeSProC's capacity to discriminate among interneuron classes and explored the existence of their subtypes in three experimental settings, corresponding to three rules for hiding class labels. First, we unlabeled all the cells of a single class and then ran the algorithm once for each class. Here, there was initially a cluster for each of the other (labeled) types and the desired result was to assign unlabeled instances to a (one of) newly formed cluster(s), allowing us to explore the potential subtypes of each class separately. Second, we simultaneously unlabeled all cells of each pair of classes, yielding six clustering scenarios; this allowed to assess whether cells of different classes would be clustered together and whether, and to what extent, would the subtypes identified in the previous setting re-appear. Finally, we simultaneously unlabeled a portion of cells of each of the four classes. This allowed unlabeled cells to be placed in their 'true' cluster, other classes (i.e., be misclassified), or assigned to a new cluster, providing insight into the homogeneity of each interneuron class. Here we unlabeled half the instances of each class (rounding down when necessary). We selected unlabeled cells by random sampling at each label reliability threshold (thus a cell might have been unlabeled at $th26$ but not at $th18$, for example), and repeated the sampling ten times. We used these three unlabeling settings for each label reliability threshold, i.e., for $th18$, $th22$, and $th26$.

For the first two settings, we defined per-class discrimination accuracy as $acc_t = c_t/u_t$, where u_t is the number of cells of the unlabeled class t and c_t the cardinality of the subset of u_t assigned to a (one of) newly formed cluster(s) (and therefore not assigned to one of the other classes). In the second setting, we averaged acc_t across the three 'scenarios' in which t was unlabeled (e.g., HT was unlabeled together with CB, LB, and MA). For the third setting, we defined two measures —'error' and 'accuracy'—as follows: $err_t = \sum_{t' \neq t} a_{tt'}/u_t$, where $a_{tt'}$ is the number of unlabeled cells of class t assigned to class t' and u_t the number of unlabeled cells of class t, and $acc_t = a_{tt}/u_t$. 'Accuracy' considers the proportion of unlabeled cells of class t classified as t whereas 'error' does not penalize assignments to newly formed clusters.

When starting the EM procedure for a mixture model $\mathcal{M}^K$, one has to choose how to initialize the parameters. Furthermore, when $K > C$ one can keep or adapt the parameters from the previous model, $\mathcal{M}^{K-1}$. We initialize the parameters with the following heuristics:

- For $m \leq C$, estimate $\boldsymbol{\theta}_m$ from cells labeled as belonging to class m.
- Keep the θ and ρ parameters from $\mathcal{M}^{K-1}$ for $C < m < K$. That is, $\boldsymbol{\theta}_m^K = \boldsymbol{\theta}_m^{K-1}$, $\boldsymbol{\rho}_m^K = \boldsymbol{\rho}_m^{K-1}$, $C < m < K$ where $\boldsymbol{\rho}_m = (\rho_{m1}, \ldots, \rho_{mF})$.
- Estimate $\boldsymbol{\theta}_K$ from the instances selected as described in paragraph 'Initializing a new component' in Section 2.2.3.
- For components $m \in \{1, \ldots, C\} \cup \{K\}$, i.e., those of fixed classes and the newly introduced one, make all features equally relevant and irrelevant: $\rho_{mj} = 0.5, \forall m \in \{1, \ldots, C\} \cup \{K\}, \forall j \in \{1, \ldots, F\}$.
- Adapt π from $\mathcal{M}^{K-1}$ to give more weight to newly discovered components $m > C$ than to class components $m' \leq C$: $\pi_m^K = 2\pi_{m'}^{K-1}, \forall m > C, \forall m' \leq C$.
- Estimate λ as if $\rho_{mj} = 0.5, \forall m, \forall j$, i.e., as if each feature was equally relevant and irrelevant for every component, in order to fully 'reset' the λ estimates.

The EM procedure iterates until log-likelihood converges or up to 25 iterations. We set $ct = 5$ for initializing new components, because this value produced the best results in preliminary experiments.

Extreme probabilities due to maximum likelihood. Estimating the parameters of a Gaussian distribution by maximum likelihood can result in zero variance, driving an instance's density to infinity. To avoid this, we use 0.1 as a minimum variance for all Gaussian distributions, i.e., for all such distributions we set $\sigma^2 = \max\{0.1, \sigma^2\}$, where σ^2 is the estimated variance.

3. Results and discussion

3.1. Discriminating among classes

3.1.1. Unlabeling a single class
When unlabeling a single class, HT and MA cells were better distinguished from other classes when label reliability increased; the opposite happened for LB, while the discrimination accuracy for CB cells was rather unaffected (see Fig. 2a). At $th26$, HT and MA cells were identified with rather high accuracy.

HT was the most easily identified class: with perfect accuracy at $th22$ and $th26$, and high accuracy (0.89) at $th18$.

Although accurately identified at $th22$ and $th26$, MA cells were confused with all the other classes, particularly with HT at $th18$ and $th22$, and CB at $th26$. In this respect, the MA seemed to be the most heterogeneous interneuron type with respect to the used variables.

LB and CB cell types were often confused with each other but easily distinguished from other types (with the exception of CB at $th18$, where it was heavily confused with HT; see Table 3). This confusion is not surprising, as even expert neuroscientists often struggle to discern these two classes [3].

3.1.2. Unlabeling two classes
When hiding two classes at a time, discrimination accuracy generally increased with label reliability (see Fig. 3a). CB cells were better discriminated than when hiding a single class, HT cells equally well, whereas LB and MA cells better on some label reliability thresholds and worse on others.

At $th26$, which contained the most reliably labeled cells, all classes were identified more accurately than when unlabeling a single class (see Figs. 2a and 3a). Furthermore, the classes were well separated in the formed clusters: only six out of the 20 clusters formed at $th26$ (columns 'A', 'B', etc., in Table 4a–f) were 'mixed', i.e., contained instances of more than one class (shown in black). Thus, even though MA cells were misclassified as CB at $th26$ (when unlabeling the MA type), these two types were neatly separated when unlabeled simultaneously (see Table 4a). HT cells were almost never placed in clusters containing other cell types (only a single CB cell was assigned to a HT cluster; see Table 4b). LB cells were least separated in the formed clusters: they were mixed with both CB and MA cells in two clusters (see Table 4e and f).

3.1.3. Partially unlabeling all classes
In this setting, discrimination accuracy generally improved with label reliability (see Fig. 4a), except for MA and CB at $th22$, where it decreased due to more instances being assigned to new clusters (indicated by their low error at $th22$; Fig 4). Likewise, discrimination error generally decreased, except for MA and LB at $th26$.

At $th26$, the most accurately classified and most homogeneous types were HT and CB, as they had lowest error and highest accuracy. MA and LB, on the other hand, displayed low accuracies but relatively low errors, suggesting they were more heterogeneous than HT and CB at $th26$.

3.1.4. Summary
Summarizing the previous three sections, we can note that, despite some fluctuations—possibly due to small data samples—discrimination accuracy tended to increase with label reliability in all three experimental settings (e.g., accuracy was almost universally higher and error lower at $th26$ than $th18$, see, e.g., Figs. 2a, 3a, 4a and b). This might suggest that the degree of label
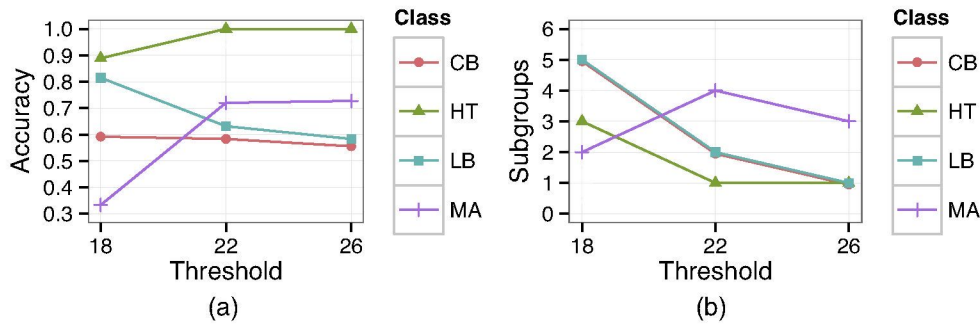
Fig. 2. Per-class discrimination accuracy and number of subtypes when unlabeling a single class, versus label reliability threshold.
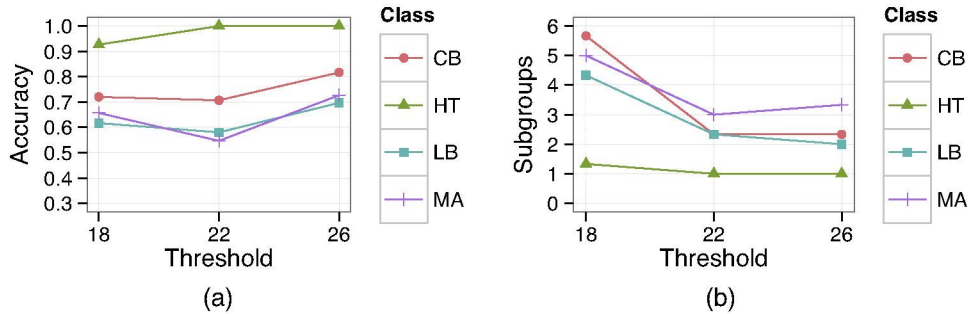

Fig. 3. Average per-class discrimination accuracy and number of subtypes when unlabeling two classes, versus label reliability threshold. The averages are taken across the three different settings in which a class is hidden, e.g., the averages for HT come from the settings: HT and LB hidden; HT and CB hidden; and HT and MA hidden.

noise decreased with label reliability threshold, and that therefore, the most reliable results were obtained at *th26.*

3.2. Potential subtypes

The number of subtypes generally decreased with label reliability (see Figs. 2b and 3b). This may indicate that there was more heterogeneity among less reliably labeled cells; nonetheless, this heterogeneity may simply be due to the higher number of instances at lower thresholds (especially for the CB, LB, and HT types). HT appeared as the most compact class as all of its cells were clustered together (in a single cluster) at *th22* and *th26*, in both the first and the second labeling scenario (see Figs. 2b and 3b and Table 4b–d). MA, on the other hand, seemed to be the most heterogeneous —at

Table 3
Clustering of unlabeled instances when hiding a single class. Each row corresponds to one 'labeling scenario', e.g., CB is the hidden class in the first row, whereas columns represent the classes to which the unlabeled cells were assigned. The cells assigned to newly formed clusters are considered as correctly classified and are thus displayed in blue color on the diagonal whereas the cells assigned to other classes are shown in red, outside the diagonal. Thus, for example, 16 MA cells were correctly classified, four were misclassified as CB, one as HT, and one as LB at *th26* (row four, rightmost table). Note that the number of formed subtypes is not shown—the 16 MA cells were placed in three clusters; the formed clusters are discussed in Section 3.2. Zeros were omitted.

(a) *th18*

	CB	HT	LB	MA
CB	29	7	11	2
HT		8		1
LB	3		22	2
MA	1	13	8	11

(b) *th22*

	CB	HT	LB	MA
CB	14		9	1
HT		5		
LB	6		12	1
MA		5	2	18

(c) *th26*

	CB	HT	LB	MA
CB	5		3	1
HT		4		
LB	4		7	1
MA	4	1	1	16

Table 4
Clustering of unlabeled instances when hiding two classes a time at *th26*. Rows represent the hidden classes whereas columns denote the clusters the instances were ascribed to. 'A', 'B', etc., denote newly formed clusters. Misclassified instances are shown in red, instances in 'pure' (not-mixed) clusters are shown in blue, and instances in mixed clusters in black. Zeros were omitted.

(a) Unlabeling MA and CB

	HT	LB	A	B	C	D	E
MA	2		6	5	5	4	
CB		3		1			5

(b) CB and HT

	LB	MA	A	B
CB			1	8
HT			4	

(c) HT and LB

	MA	CB	A	B
HT			4	
LB		4		8

(d) HT and MA

	CB	LB	A	B	C	D
HT				4		
MA	2	5	6		4	5

(e) LB and MA

	CB	HT	A	B	C
LB	6		5	1	
MA	8	1	5	2	6

(f) CB and LB

	HT	MA	A	B	C	D
CB		2		4	2	1
LB		1	6		2	3

Fig. 4. Per-class discrimination accuracy (acc_t) and error (err_t) when unlabeling half the instances of each class, versus label reliability threshold.

th26 MA cells were clustered into at least three subtypes (see Fig. 2b and Tables 4a, d and e). Thus, we focused on *th26* for analyzing the formed subtypes as it contained the most reliably labeled interneurons.

3.2.1. With a single hidden class

MA cells were clustered in three groups (see Fig. 5 for representative examples), with six, five, and five cells each, whereas a single distinct subtype was identified for the CB and LB classes, counting five and seven cells each, respectively. Since all HT cells were placed in a single cluster (see Fig. 2b), no potential subtypes of HT were identified.

Overall, MA subtypes showed relatively sparse axonal arbors, as indicated by their low or medium values for X_1 and X_2 (see Figs. 5a–c), with MA-A cells being less sparse than MA-B and MA-C ones. MA-A cells had plenty of axonal arborization far from soma (high X_5 values in Fig. 5a) and dendritic polar histogram length in the $[\pi, 2\pi)$ interval (X_7). MA-B cells exhibited medium values for all variables (Fig. 5b) whereas MA-C had the sparsest axons (low values for X_1 and X_2 in Fig. 5c) and, like MA-A, plenty of axon far from soma (X_5) and dendrites in the $[\pi, 2\pi)$ polar histogram interval (X_7). CB-A cells displayed relatively sparse axons (medium X_1 and low X_2 values in Fig. 5d), with little axon far from the soma (low X_5 values) and little dendritic arborization far from soma and in the $[\pi, 2\pi)$ polar histogram interval (low X_7 and X_9 values). LB-A cells exhibited the most dense axonal and dendritic arbors, with high or medium values for X_1, X_2, X_6 and X_7; the X_4 values (axonal length at medium distance from soma), was especially high (see Fig. 5e).

Validating the produced clusters. While the discovered subtypes are relatively small—their sizes ranging from five to seven cells, they may nonetheless be relevant in the domain of neuronal classification, where 3D neuronal reconstructions, and reliably classified reconstructions in particular, are scarce. Cluster quality indices [24,25] may thus help assess the goodness of the clustering solution.

One type of indices compares the obtained (crisp) clustering partition with the original one, given by class labels. We performed such an analysis in Section 3.1 when we computed accuracy, and now we report a measure more commonly used for this end, the Adjusted Rand index (ARI; [26,27]),

$$
\mathrm{ARI} = \frac{\sum_{c=1}^{C}\sum_{m=1}^{K}\binom{N_{cm}}{2} - \frac{\sum_{c=1}^{C}\binom{N_{c.}}{2}\sum_{m=1}^{K}\binom{N_{.m}}{2}}{\binom{N}{2}}}{\frac{1}{2}\left[\sum_{c=1}^{C}\binom{N_{c.}}{2} + \sum_{m=1}^{K}\binom{N_{.m}}{2}\right] - \frac{\sum_{c=1}^{C}\binom{N_{c.}}{2}\sum_{m=1}^{K}\binom{N_{.m}}{2}}{\binom{N}{2}}}
$$

where N_{cn} is the number of instances of class c assigned to cluster m. ARI rewards two types of agreements: (a) clustering a pair

of instances together (i.e., as members of a same cluster) in both partitions; and (b) clustering a pair of instances separately (i.e., as members of different clusters) in both partitions. It reaches its maximum value, 1, when the two partitions agree perfectly, and this occurs when unlabeling HT cells (see Table 5). We also achieved high ARI values when unlabeling CB and LB cells whereas we obtained a low one when unlabeling MA cells; this is due to the discovery of three MA subtypes in the setting, which was not favored by ARI because it increased the difference among the two partitions.

We cannot easily compute a second type of clustering quality indices, based on assessing properties such intra-cluster compactness and inter-cluster separation, because that would require computing distances among data points, something which is unclear how to do when points are located in different feature subspaces. When then assessed our clustering in terms of probabilistic concordance, considering that a clustering is good if cluster membership probabilities, $p(\mathbf{z}_i)$, are similar among the members of a same cluster and different among members of different clusters. We measured the similarities among cluster membership probabilities of two data instances with Jensen–Shannon divergence,

$$
d_{JS}(p(\mathbf{z}_i), p(\mathbf{z}_j)) = \frac{1}{2}(d_{KL}(p(\mathbf{z}_i), r) + d_{KL}(p(\mathbf{z}_j), r)),
$$

where $r = \frac{1}{2}(p(\mathbf{z}_i) + p(\mathbf{z}_j))$ and $d_{KL}(p(\mathbf{z}_i), p(\mathbf{z}_j))$ is the Kullback–Leibler divergence [28] between $p(\mathbf{z}_i)$ and $p(\mathbf{z}_j)$,

$$
d_{KL}(p(\mathbf{z}_i), p(\mathbf{z}_j)) = \sum_{m=1}^{K} p(z_{im}) \log\left(\frac{p(z_{im})}{p(z_{jm})}\right).
$$

Note that $p(z_{jm})$ is simply $\gamma(z_{im})$ computed in the final step of the EM algorithm. Unlike Kullback–Leibler divergence, Jensen–Shannon divergence is symmetric, does not require absolute continuity (i.e., that $p(z_{im}) = 0 \Rightarrow p(z_{jm}) = 0$), its square root is a metric, and it is bounded: $0 \leq d_{JS} \leq 1$ [29].

Using Jensen–Shannon divergence as the measure of distance among data instances, we computed the Silhouette width [30] clustering index, thus measuring intra-cluster compactness and

Table 5

Cluster validation indices when hiding a single class at *th26*. Columns denote the unlabeled class whereas rows correspond to different metrics.

	HT	MA	CB	LB
Adjusted Rand index	1.00	0.42	0.83	0.79
Silhouette	1.00	1.00	1.00	1.00

(a) `MA-A`

(b) `MA-B`
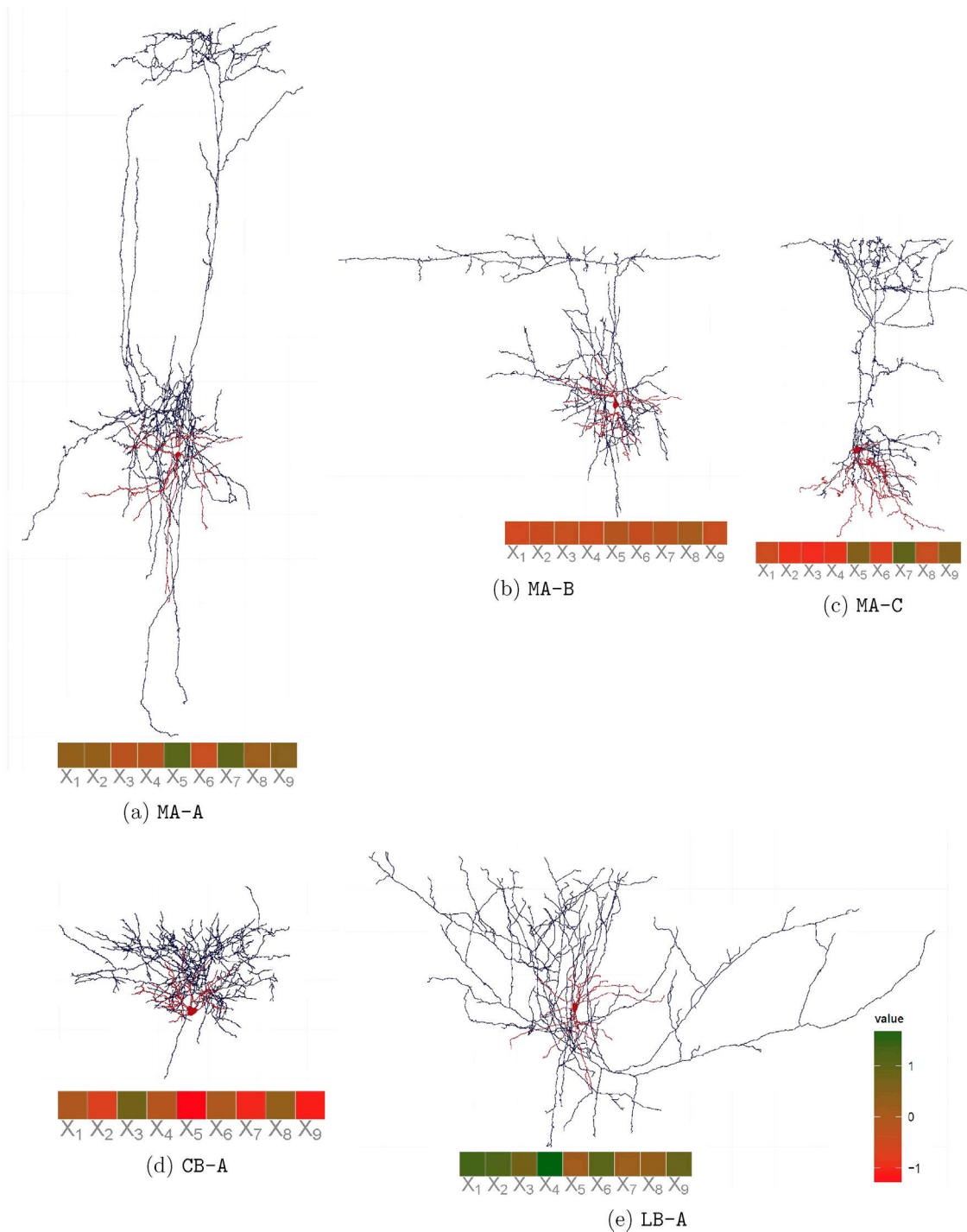
(c) `MA-C`

(d) `CB-A`

(e) `LB-A`

Fig. 5. Representative members of potential subtypes identified at *th26*. The heatmaps show the subtypes' mean values for all variables. Due to standardization, values are not comparable among different variables. Thus, for example, although $X_1 > X_2$ in the case of `CB-A`, it does not necessarily mean that `CB-A` cells have (on average) more axonal length in the $[0, \pi)$ than in the $[\pi, 2\pi)$ polar histogram interval; it means that they have more of the former *relative* to the values of the remaining (i.e., non-`CB-A`) cells at *th26*. The five depicted subtypes contain 28 out of the 47 cells at *th26*; `HT` cells and misclassified `MA`, `CB`, and `LB` cells are not represented. (For interpretation of the references to color in the text, the reader is referred to the web version of the article.)

inter-cluster separation in terms of this distance. The Silhouette width is given by

$$SW = \frac{1}{N} \sum_{i=1}^{N} \frac{b_i - a_i}{\max(b_i, a_i)},$$

where a_i is the average distance between $\mathbf{x}_i$ and other points in its cluster, while b_i is its average distance to the points in the closest cluster (defined as that yielding the lowest b_i). We achieved maximum Silhouette values (its values range from -1 to 1) for all labeling scenarios (see Table 5), showing that the cluster membership probabilities had converged reasonably.

A partial cause for high Silhouette widths was that we only unlabeled a subset of instances in each setting, and it was only those instances' $p(\mathbf{z}_i)$ that were estimated by our algorithm, while the rest instances' $p(\mathbf{z}_i)$ were fixed to degenerate distributions (probability
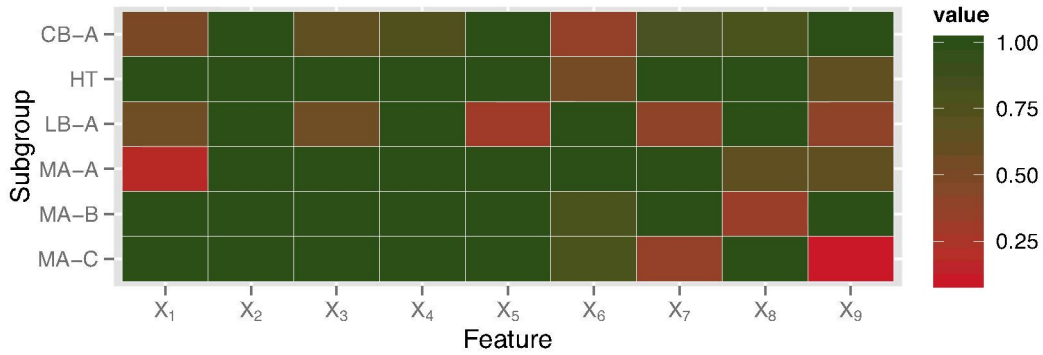
Fig. 6. Relevance (ρ) of features (X_1–X_9) for the HT, LB, CB, and MA subtypes identified at *th26* when hiding a single type. Each row corresponds to one subtype, e.g., the last three rows correspond to the three subtypes of MA. The displayed subtypes do not contain all the cells contained at *th26*, as some were misclassified (e.g., besides CB-A, CB cells at *th26* were also assigned to the LB and MA classes; the latter cells are not contained in any subtype represented in this graph). (For interpretation of the references to color in the text, the reader is referred to the web version of the article.)

Table 6
Average per-cluster Silhouette width. Rows correspond to the hidden (unlabeled) classes whereas the columns denote clusters. '-A', '-B', and '-C' denote newly formed clusters; for example, the intersection of the first row and fifth column (-A) corresponds to the HT-A cluster, whose average Silhouette width was 1.000.

	CB	HT	LB	MA	-A	-B	-C
CB		1.000	1.000	0.996	1.000		
HT	1.000		1.000	1.000	1.000		
LB	0.994	1.000		0.996	0.985		
MA	0.990	1.000	1.000		0.997	0.999	1.000

1 for the class labels' cluster; 0 for all other clusters) which were different among the clusters. Thus, e.g., when unlabeling the HT cells, we estimated the $p(\mathbf{z}_i)$ of only four instances, the overall Silhouette width therefore necessarily being high, due to the inter-cluster differences among the fixed $p(\mathbf{z}_i)$. Yet, as Table 6 shows, Silhouette widths were also high for the newly discovered clusters, not only for those corresponding to the known types (and thus containing many cells with fixed degenerate $p(\mathbf{z}_i)$).

Finally, in the next section we validated the discovered subtypes with a different experimental setting—the hiding of two classes, thus evaluating their 'stability', i.e., their robustness to different labeling scenarios.

3.2.2. With two hidden classes

The validity of the above-described subtypes of the MA, CB, and LB types, identified when hiding a single class, was confirmed when hiding two classes simultaneously. That is, in almost every labeling scenario in the second setting (i.e., for every pair of hidden classes), there was a cluster that greatly resembled the corresponding subtypes. So, for example, when hiding CB and MA, cluster E (Table 4a) was identical to subtype CB-A. Furthermore, four cells from the CB-A subtype were clustered together in every labeling scenario, that is, the intersection of CB-A and cluster E in Table 4a, cluster B in Table 4b, and cluster B in Table 4f consisted of four cells. This subset of CB-A cells emerged as its 'core' of highly similar instances, showing the robustness of this subtype (which totalled five cells).

A 'core' of the MA-C subtype (i.e., the MA-C cells that were always clustered together) emerged, consisting of four cells which formed the intersection of the pure MA clusters C, A, and C in Table 4a, d, and e, respectively (the latter two clusters being identical). Regarding MA-A, a 'core' of three cells emerged, defined by the intersection of clusters A, C and A in Table 4a, d, and e, respectively (the latter being a mixed cluster). Finally, clusters C and D in Table 4a and d contained four and three MA-B cells, respectively.

A LB-A core of five cells emerged, contained in clusters A, B, and A in Table 4c, e and f, respectively. The latter two contained a larger LB-A core of six cells.

3.2.3. Feature relevance

We focused on the first setting (i.e., hiding a single class) and *th26* to analyze the estimated relevance of predictor variables. Overall, all predictor variables seemed useful, as each one was very likely relevant (around 100% chance of being relevant; dark green boxes in Fig. 6) for at least two subtypes/classes identified at *th26*. Feature X_2—axonal polar histogram in the $[\pi, 2\pi)$ interval—appeared to be the most useful as it was very likely relevant for all the subtypes/classes (see Fig. 6). While its relevance for the HT class and MA subtypes is clear—an MA's axon grows predominantly upwards from the soma whereas the opposite holds for HT—it is interesting that it was relevant for the CB and LB subtypes as well. Features X_4 and X_5, which capture the length of axonal arborization at 150–300 μm and over 300 μm from soma, were relevant for five (out of six) subtypes. On the other hand, feature X_9—length of dendritic arborization at over 90 μm from soma—appeared as least useful as it was very likely relevant for only two subtypes (CB-A and MA-B). In general, axonal features (X_1 to X_5) were more likely to be relevant than dendritic ones (X_6 to X_9). For example, an axonal feature was, on average, very likely relevant for 4.6 subtypes/classes in Fig. 6 whereas a dendritic one was for 2.5. Likewise, a dendritic feature was, on average, probably irrelevant (below 50% chance of being relevant; red and brown boxes in Fig. 6) for more subtypes than an axonal feature.

Moreover, as shown by Fig. 6, the relevance of axonal features differed among the subtypes/classes. All axonal features were very likely relevant for all MA subtypes (except X_1 for MA-A: MA-A, unlike MA-B and MA-C, did not stand out regarding X_1—note the olive green box for X_1 in Fig. 5) and for the HT class. This suggests that the axonal arborizations of the MA subtypes and the HT class were distinct among themselves and from the CB-A and LB-A subtypes according to all axonal features. On the other hand, two and three (out of five) axonal features were relevant for the CB-A and LB-A subtypes, respectively, and, in particular, polar histogram in the $[0, \pi)$ interval and close to soma axonal arborization length (X_1 and X_3, respectively) were relevant with only 50% chance for both of those subtypes. This suggests that CB-A and LB-A were not particularly distinct according to those variables.

In summary, the results clearly show that the length of axonal polar histogram in the $[\pi, 2\pi)$ interval (X_2) is particularly relevant, followed by axonal length relatively near to the cell body (X_4) and the extent of the axon far from the cell body (X_5). In addition, the dendritic arborization was highly relevant for the

characterization of CB cells and MA cells. These findings may be useful for generating an accurate automatic classifier of 3D reconstructed interneurons.

4. Conclusions

Automatic classification of interneurons is a hard problem because data are scarce, the cells are morphologically, molecularly, and physiologically variable, and experts disagree on the definitions of features that distinguish them. Thus, we introduced a semi-supervised approach to this problem. Besides discriminating between types, this approach leads to the discovery of new types of neurons. It uses the available knowledge—in the shape of class labels—and also attempts to identify the relevance of each feature for each type and subtype.

We presented results on the classification of common basket, large basket, horse-tail, and Martinotti cells. We quantified the neurons with simple morphological features which describe the distribution and the orientation of axonal and dendritic arbors, seeking to mimic the way in which an expert visually classifies a neuron. The algorithm accurately discriminated among the different types when one and two types were unlabeled at a time and when half the instances of all types were unlabeled. Importantly, it identified potential subtypes of common basket, large basket, and Martinotti cells, suggesting that these types are more heterogeneous than previously thought. Although the identified subtypes are small, they may be indicative of the characteristics that differentiate cells belonging to the same interneuron type.

The defined morphological variables seemed useful for discriminating among the types. Axonal features seemed more useful than dendritic ones, with the axonal polar histogram length in the $[\pi, 2\pi)$ interval appearing to be the most useful feature. The defined variables might then be considered in future studies of interneuron classification. Still, it is possible that more complex variables, such as those considering both the distance and position with respect to the soma (e.g., over 300 μm from soma and above it) could further improve discrimination accuracy.

Overall, the results suggest that a semi-supervised approach may be helpful in neuronal classification and characterization. Further studies, with different morphological features and more neurons, would be needed in order to obtain more conclusive results. Although we focused on a subpopulation of GABAergic interneurons, the presented methodology could be applied to other types of cells as well.

The used algorithm, SeSProC, is open to further improvements, like the inclusion of uncertainty into labels. Instead of a single label—the class most voted by the experts—per data instance, a probability distribution could be learned from the experts' votes, thus taking all votes into account. Alternatively, an approach similar to [31] might be applied: a separate model could be learned for each set of experts with similar opinions, and these models then combined into a final, consensus model. Regarding the data, although it is generally thought that the same morphological types of neurons are found in all species, we cannot discard the possibility of inter-species variability. Thus, we plan to further analyze the data taking into account the different species in order to find types of neurons that may be representative of particular species.

Finally, it has been shown that GABAergic interneurons are affected in several brain diseases but not all of these interneurons are equally affected. For example, alterations of chandelier cells expressing the calcium binding protein parvalbumin have been associated with certain forms of epilepsy and schizophrenia. In addition, possible alterations of interneurons expressing somatostatin, neuropeptide Y and vasoactive intestinal peptide, probably including Martinotti and horse-tail (or double bouquet) interneurons, have been reported in schizophrenia (e.g., [32–35]). Since the algorithm used in the present study accurately discriminated between different interneuron types and new groups can be discovered, its application to study brain diseases may shed light on interneuron pathology in these diseases, providing a tool to more accurately determine which subtypes may be affected.

References

[1] DeFelipe J. Cortical interneurons: from Cajal to 2001. Prog Brain Res 2002;136:215–38.

[2] Ascoli GA, Alonso-Nanclares L, Anderson S, Barrionuevo G, Benavides-Piccione R, Burkhalter A, et al. Petilla terminology: nomenclature of features of GABAergic interneurons of the cerebral cortex. Nat Rev Neurosci 2008;9(7):557–68.

[3] DeFelipe J, López-Cruz PL, Benavides-Piccione R, Bielza C, Larrañaga P, Anderson S, et al. New insights into the classification and nomenclature of cortical GABAergic interneurons. Nat Rev Neurosci 2013;14(3):202–16.

[4] Tsiola A, Hamzei-Sichani F, Peterlin Z, Yuste R. Quantitative morphologic classification of layer 5 neurons from mouse primary visual cortex. J Comp Neurol 2003;461(4):415–28.

[5] Benavides-Piccione R, Hamzei-Sichani F, Ballesteros-Yáñez I, DeFelipe J, Yuste R. Dendritic size of pyramidal neurons differs among mouse cortical regions. Cereb Cortex 2006;16(7):990–1001.

[6] Cauli B, Porter JT, Tsuzuki K, Lambolez B, Rossier J, Quenet B, et al. Classification of fusiform neocortical interneurons based on unsupervised clustering. Proc Natl Acad Sci 2000;97(11):6144–9.

[7] Karagiannis A, Gallopin T, Dávid C, Battaglia D, Geoffroy H, Rossier J, et al. Classification of NPY-expressing neocortical interneurons. J Neurosci 2009;29(11):3642–59.

[8] McGarry LM, Packer AM, Fino E, Nikolenko V, Sippy T, Yuste R. Quantitative classification of somatostatin-positive neocortical interneurons identifies three interneuron subtypes. Front Neural Circuits 2010;4(12).

[9] Guerra L, McGarry LM, Robles V, Bielza C, Larrañaga P, Yuste R. Comparison between supervised and unsupervised classifications of neuronal cell types: a case study. Dev Neurobiol 2011;71(1):71–82.

[10] Chapelle O, Schölkopf B, Zien A, editors. Semi-supervised learning. Cambridge, MA: The MIT Press; 2006.

[11] Zhu X, Goldberg AB. Introduction to semi-supervised learning. Synth Lect Artif Intell Mach Learn 2009;3(1):1–130.

[12] Guerra L, Bielza C, Robles V, Larrañaga P. Semi-supervised projected model-based clustering. Data Min Knowl Discov 2013;28(4):1–36.

[13] Law MH, Figueiredo MA, Jain AK. Simultaneous feature selection and clustering using mixture models. IEEE Trans Pattern Anal Mach Intell 2004;26(9):1154–66.

[14] Raykar VC, Yu S, Zhao LH, Valadez GH, Florin C, Bogoni L, et al. Learning from crowds. J Mach Learn Res 2010;11:1297–322.

[15] Li Y, Wessels LF, de Ridder D, Reinders MJ. Classification in the presence of class noise using a probabilistic Kernel Fisher method. Pattern Recognit 2007;40(12):3349–57.

[16] Angluin D, Laird P. Learning from noisy examples. Mach Learn 1988;2(4):343–70.

[17] Guerra L, Benavides-Piccione R, Bielza C, Robles V, DeFelipe J, Larrañaga P. Semi-supervised projected clustering for classifying GABAergic interneurons. In: Peek N, Marín Morales R, Peleg M, editors. Artificial intelligence in medicine; vol. 7885 of lecture notes in computer science. Berlin: Springer; 2013. p. 156–65.

[18] Ascoli GA, Donohue DE, Halavi M. Neuromorpho.org: A central resource for neuronal morphologies. J Neurosci 2007;27(35):9247–51.

[19] Glaser JR, Glaser EM. Neuron imaging with Neurolucida – a PC-based system for image combining microscopy. Comput Med Imaging Graph 1990;14(5):307–17.

[20] McMullen NT, Glaser EM, Tagamets M. Morphometry of spine-free non-pyramidal neurons in rabbit auditory cortex. J Comp Neurol 1984;222(3):383–95.

[21] Li Y, Dong M, Hua J. Simultaneous localized feature selection and model detection for Gaussian mixtures. IEEE Trans Pattern Anal Mach Intell 2009;31(5):953.

[22] Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B (Methodol) 1977;39(1):1–38.

[23] Akaike H. A new look at the statistical model identification. IEEE Trans Autom Control 1974;19(6):716–23.

[24] Halkidi M, Batistakis Y, Vazirgiannis M. On clustering validation techniques. J Intell Inf Syst 2001;17(2–3):107–45.

[25] Handl J, Knowles J, Kell DB. Computational cluster validation in post-genomic data analysis. Bioinformatics 2005;21(15):3201–12.

[26] Rand WM. Objective criteria for the evaluation of clustering methods. J Am Stat Assoc 1971;66(336):846–50.

[27] Hubert L, Arabie P. Comparing partitions. J Classif 1985;2(1):193–218.

[28] Kullback S, Leibler RA. On information and sufficiency. Ann Math Stat 1951;22(1):79–86.

[29] Lin J. Divergence measures based on the Shannon entropy. IEEE Trans Inf Theory 1991;37(1):145–51.

[30] Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 1987;20:53–65.

[31] López-Cruz PL, Larrañaga P, DeFelipe J, Bielza C. Bayesian network modeling of the consensus between experts: an application to neuron classification. Int J Approx Reason 2014;55(1):3–22.

[32] DeFelipe J. Chandelier cells and epilepsy. Brain 1999;122(10):1807–22.

[33] Lewis DA. The chandelier neuron in schizophrenia. Dev Neurobiol 2011;71(1):118–27.

[34] Inan M, Petros TJ, Anderson SA. Losing your inhibition: linking cortical GABAergic interneurons to schizophrenia. Neurobiol Dis 2013;53:36–48.

[35] Joshi D, Fullerton JM, Weickert CS. Elevated ErbB4 mRNA is related to interneuron deficit in prefrontal cortex in schizophrenia. J Psychiatr Res 2014;53:125–32.