Machine learning for cell classification and neighborhood analysis in glioma tissue

Leslie Solorzano[1], Lina Wik[2], Thomas Olsson Bontell[3,4], Yuyu Wang[2], Anna H. Klemm[1,5], Johan Öfverstedt[1], Asgeir S. Jakola[6,7], Arne Östman[2], and Carolina Wählby[1,5]

[1]Dept. of Information Technology, Uppsala University, Sweden

[2]Department of Oncology-Pathology, Karolinska Institutet, Sweden

[3]Dept. of Clinical Pathology, Sahlgrenska University Hospital, Sweden

[4]Dept. of Physiology, Inst. of Neuroscience and Physiology, University of Gothenburg, Sahlgrenska Academy, Sweden

[5]Science for Life Laboratory, SciLifeLab, Sweden

[6]Dept. of Neurosurgery, Sahlgrenska University Hospital, Sweden

[7]Inst. of Neuroscience and Physiology, Dept. of Clinical Neuroscience, University of Gothenburg, Sahlgrenska Academy, Sweden

1 Abstract

Multiplexed and spatially resolved single-cell analyses that intend to study tissue heterogeneity and cell organization invariably face as a first step the challenge of cell classification. Accuracy and reproducibility are important for the downstream process of counting cells, quantifying cell-cell interactions, and extracting information on disease-specific localized cell niches. Novel staining techniques make it possible to visualize and quantify large numbers of cell-specific molecular markers in parallel. However, due to variations in sample handling and artefacts from staining and scanning, cells of the same type may present different marker profiles both within and across samples. We address multiplexed immunofluorescence data from tissue microarrays of low grade gliomas and present a methodology using two different machine learning architectures and features insensitive to illumination to perform cell classification. The fully automated cell classification provides a measure of confidence for the decision and requires a comparably small annotated dataset for training, which can be created using freely available tools. Using the proposed method,

we reached an accuracy of 83.5% on cell classification without the need for standardization of samples. Using our confidence measure, cells with low-confidence classifications could be excluded, pushing the classification accuracy to 94.47%. Next, we used the cell classification results to search for cell niches with an unsupervised learning approach based on graph neural networks. We show that the approach can re-detect specialized tissue niches in previously published data, and that our proposed cell classification leads to niche definitions that may be relevant for sub-groups of glioma, if applied to larger datasets.

2 Introduction

Spatially resolved single cell analysis allows for cell characterization in tumor microenvironment (TME) studies, and can be used by itself or in combination with other methods to find subgroups of disease that are associated with distinct clinical outcomes [1]. From the numerous multiplex staining methodologies (nicely reviewed in [2]), we want to mention two fluorescence based platforms that have complementary characteristics: i) co-detection by indexing (CODEX) [3] and ii) multiplexed immunofluorescence (mIF). While technologies like CODEX allow the simultaneous study of up to fifty biomarkers, it is low in throughput as it requires repeated rounds of staining and de-staining. In contrast, while mIF allows for the simultaneous staining of only up to eight markers, its high throughput nature allows to scan and analyze more samples. Previously, mIF has been used to quantify marker expression [4, 5, 6, 7], and in conjunction with other techniques like flow cytometry, H&E, IHC and single-cell RNAseq [8, 9, 10]. In [11] mIF stained cell populations in the colorectal cancer tumour microenvironment are identified using QuPath[12]. In [13] mIF stained cells are classified based on manual thresholding and Random Forests using the HALO Highplex and the InForm software. All these approaches include a manual adjustment of thresholds or normalization on a per-image basis, risking the introduction of bias. In [14] a more advanced pipeline is applied, including cell classification using a one-vs-all multi-class binary classification with support vector machines using InForm and Matlab.

To classify cells and study cell niches in tumor tissue, commonly, cells are first identified by segmentation of cell nuclei, followed by dilation to approximate a cell boundary, as implemented in popular open source software such as CellProfiler [15] and QuPath [12]. Private software for image analysis in mIF includes Akoya's InForm, HALO and Aperio [2]. Closed software has a disadvantage of limited possibilities for sharing and reproducing projects and results. On the other hand, QuPath is open, offers the possibility of using Random Forests but doesn't yet allow deep learning classifiers natively. However it includes scripting capabilities which makes it possible to extract the outlines of segmented cells and work with feature extraction and cell classification using other tools.

Typically, the next step in cell classification is marker quantification. Per cell marker quantification composes a profile that can be used to assign a class to a cell from an expected list of classes associated to a known marker panel. Fluorescent stains can have some degree of spectral overlap and bleed-through and each sample may have intensity variability due to sample-, staining- and scanning artefacts such as uneven illumination. To approach these challenges, a typical workflow of cell classification using mIF involves manual/visual adjustment and/or thresholding of images or per-cell measurements to define which cells are positive and negative for each specific marker. These adjustments are often made for each image and stain separately, making it a tedious process that is difficult to reproduce and may introduce bias. Stain normalization in histology is an intensely studied area [16], but few solutions work across large and multiplexed datasets where samples come from multiple sources, and the task becomes more challenging as more stains are added.

We present a methodology for fully automated cell classification that is able to provide a confidence measure, and compare two very popular and established methods in Machine Learning: i) an ensemble of fully connected neural networks (FCNNs) and ii) Extreme Gradient Boosting (XGBoost) [17]. FCNNs create a weighted linear combination of the features which is able to take into account the variation within a class. Each of the neurons in each layer is connected to the adjacent layers and in this way all the knowledge is propagated. XGBoost is an advanced version of RF, an ensemble of decision trees, which, at each round of training, learns to focus on the hardest points to classify. Additionally, XGBoost uses the second order derivative of the objective function to gather more information. Both methods use gradient information to arrive to a solution. But they are also very different, create different solution landscapes and have very different parameters and behaviors. We additionally explore using ensembles of classifiers, and propose a set of new marker intensity features that have less variation under uneven illumination, improving classification accuracy.

To validate our methodology we trained our machine learning methods on features extracted from manually classified cells from 11 cores randomly selected from a collection of TMAs. The methods were evaluated on a separate set of 10 cores randomly selected from the same collection. Cell segmentation was performed using QuPath and class labels for training and testing were created selecting a threshold to create a "single measurement classifier" per marker.

Following cell classification we proceeded to use the results for spatial analysis and cell niche characterization with an unsupervised graph neural network approach, spage2vec [18], originally developed for characterization of spatially resolved gene expression. We present a brief validation of its use on previously published spatially resolved single cell data obtained by CODEX [3]where we can compare to other niche detection approaches. Finally, we apply spage2vec to our 21 TMA cores creating maps of local tissue niches with similar cellular neighborhoods.

3 Methods

3.1 Tissue collection and mIF

Lower grade gliomas were verified histopathologically, operated and embedded in paraffin between 2007 and 2016 at the Department of Neurosurgery, Sahlgrenska University Hospital, Gothenburg, Sweden. All procedures involving human specimens in this study were approved by the regional ethical committee in Gothenburg (Ep Dnr: 1067-16), and informed consent was given by the patients. Diagnoses were retrospectively confirmed and corrected by molecular classification. Tissue microarrays (TMAs) of tumor cores with a diameter of 1.2 mm were constructed at the Human Protein Atlas, Department of Immunology, Genetics and Pathology, Uppsala University, Sweden. Each TMA also contained two control cores from each of white and grey matter. TMAs underwent octaplex immunofluorescence staining using an 7-plex Opal kit and additional Opal 480 and Opal 780 reagent kits (Akoya Biosciences, Marlborough, US) according to the manufacturer's instructions. Opal 480, Opal 520, Opal 540, Opal 570, Opal 620, Opal 650, Opal 690 and Opal 780 were used to detect antibodies against the biomarkers: Iba1, Ki-67, TMEM119, NeuroC, MBP, mutIDH1, CD34, GFAP. Nuclei were counterstained with spectral DAPI provided with the kit for a 5-minute incubation at room temperature. TMAs were coverslipped using Prolong Gold antifade mountant (Thermo Fisher Scientific). Images were obtained after scanning TMAs through Vectra Polaris (Akoya Biosciences) at 20X magnification with optimized exposure time lengths, mapping each core with Phenochart image analysis software (Akoya Biosciences) and performing spectral unmixing and autofluorescence substraction using an image of unstained tissue. Prior to cell classification, core edges as well as tissue artifacts with poor marker quality were masked away.

3.2 Ground truth for model training and evaluation

To create ground truth for model training and methods evaluation we selected 21 random cores from a cohort of TMAs. Then, based on the mean marker intensity in each cell, extracted using QuPath as described above and shown in Figure S1, we manually selected a single threshold per marker and per image to indicate which cells in the image express the marker.

Cell type identification is not straight forward, and relies on more than one biomarker per cell type. Additionally, several cell types may express overlapping markers. Therefore, we set up a classification scheme using a lookup table in order to systematically classify each cell based on the combined positivity/negativity for the markers in our biomarker panel. Rules for classification were based on existing knowledge about marker expression. Markers for astrocytes (GFAP), neurons (NeuN, betaIII-tubulin, NF-H, MAP2), myeloid derived macrophages and microglia

(Iba1), endothelium (CD34), IDH1 mutated gliomas (IDH1 R132H) and proliferation (Ki-67) are well established. TMEM119 was recently shown to be specifically expressed in brain resident microglia and was used to distinguish between these cells and myeloid derived macrophages [19]. The lookup table is illustrated in Figure S2.

We then proceeded to use the lookup table to assign a definitive training label based on the combination of markers in each cell. As a result each cell received a single class label corresponding to one of six cell classes expected to be in the tissue: Astrocyte, Glioma, Neuron, Microglia, Macrophage, Endothelial. A small group of cells had combinations of markers that could not be adequately mapped to an expected cell type, and another sub group did not surpass any threshold resulting in no class. These two subgroups were deemed as "ambiguous" and were not used for training or evaluating the final automated classification results.

The subset of 21 cores labeled by expert 1 was further divided into two subsets, 11 cores for training and parameter optimization, and 10 cores for independent testing, taking care that the proportions of each cell class was as similar as possible in the two subsets which was limited due to class imbalance (we avoided having 0 cells representing a class in either testing or training). We made sure that cores are completely separate, none of the annotated cells used for training appears in the test set. Manual/visual determination of thresholds is challenging and highly subjective. We therefore let a different person (later referred to as Expert 2) independently determine thresholds for the 10 cores used for method evaluation. In total, Expert 1 classified 54953 cells from 21 cores, while leaving 8280 (15%) as ambiguous, and Expert 2 classified 22293 cells in 10 cores, leaving 7992 (35%) as ambiguous.

3.3 Comparison with cell classification by average marker thresholds

As benchmarking for automating cell classification, we averaged all the manual thresholds for each marker across cores. We then applied these fixed thresholds to obtain a simple automated classification of the cells in the test set. This experiment functioned as a benchmark for comparison to the more advanced automated classification approaches described below.

3.4 Cell segmentation and feature extraction

Using QuPath and the DAPI channel as input, we segmented the cell nuclei in each core. Then we used dilation to define an area around each nucleus representing a region for quantifying cytoplasmic markers. It should be noted that, although we here on refer to this as a cell segmentation, this is in fact not a proper cell segmentation, which would be far more complicated due to the high variability of cell morphology in the tissue. All segmentation parameters were kept fixed for the full experiment as well as for creating ground truth.

After performing cell segmentation we quantified basic statistics of intensities (mean, min, max, and standard deviation) using QuPath. These features may however be influenced by errors in cell segmentation, e.g., small parts of neighboring cells reaching inside the very approximate cell outline may heavily influence a measure of mean intensity. They are also dependent on core-to-core variations in intensity.

We therefore exported the cell outlines and created a separate pipeline for further feature extraction and processing using Python Pandas library for data treatment and Scikit-Image [20] for image analysis. Using this pipeline we first measured the 90th percentile of intensity per cell and marker. Our reason for choosing the 90th percentile was to avoid noise from markers in neighboring cells. The 90th percentile also includes more information than the median (50th percentile) which would be close to zero when the stain is present only in the outer rim of the object, and not in the nucleus. With the 90th percentile we include more of the relevant information provided by the markers in the cell.

Next, we wanted to define a set of cell features with reduced core-to-core variation. While the intensities per marker and core may vary, the difference between markers remains similar from core to core. Hypothesizing that per-marker differences provide a valuable feature to circumvent core normalization we extracted the differences of 90th percentiles of intensity (here on referred to as d90s) for every pair of different markers. We reasoned that given that each class is determined by a combination of markers (as opposed to a single one), we provide the d90s to the models instead of them having to learn it. Leaving out DAPI and without repeating combinations of the remaining eight markers there area total of 28 d90s. Concatenating these d90s to the regular intensity measurements: mean, minimum, maximum, standard deviation of the eight stains we obtain a feature vector per cell of length 68. We also experimented with other features from the intensities per cell such as the difference of median, mutual information and linear correlation (the former two being between pairs of markers inside the cell per pixel). These features were able to distinguish core similarities but they were not useful for a localized cell classification.

3.5 Machine learning models and cell classification confidence measures

3.5.1 Data augmentation

Regardless of machine learning model, we augmented the training data to combat the imbalanced cell classes, with the glioma class containing about 100 times more cells than the other classes (Astrocyte: 1667, Glioma:23168, Neuron:695, Microglia:230, Macrophage:2498, Endothelial:685). The augmentation procedure consisted of generating new "cell entries" by taking real cell feature values and adding Gaussian noise with 0 mean and a standard deviation equal to each feature's standard deviation. In this way every epoch of FCNN or XGBoost was trained with a new

augmented training set which was created with a balanced number of cells per class (n=20000). Each model was given a set of cell labels and per-cell features for training. We compared two sets of features: i) the regular set of features per marker per cell (a total of 40 for ten channels (eight cell markers plus DAPI and autofluorescence) and four measurements: mean, min, max, and std dev) and ii) adding the 28 d90s, leading to a total of 68 features. The cell classes were Astrocyte, Glioma, Neuron, Microglia, Macrophage and Endothelial.

3.5.2 Fully connected neural networks (FCNN)

We created a fully connected neural network (FCNN) architecture for cell classification consisting of an input layer with the size of the cell feature vector (size 68) and as output a feature vector of size six corresponding to the six different cell types. The model had three hidden fully connected layers in the middle, of sizes 100, 200 and 300 with rectified linear unit (ReLU) activations [21]. We used batch normalization [22] after the second and third hidden layer, before the activation functions, to reduce internal covariate shift and avoid convergence problems. To train the network, we used AdaBelief [23] as optimizer. This optimizer arrived to the solution twice as fast as a stochastic gradient descent optimizer [24] while providing similar classification accuracy. We trained each FCNN for 15 epochs.

3.5.3 FCNN ensemble and confidence measure

Ensembles of machine learning models are well known to improve accuracy and robustness [25, 26]. To answer the question of whether an ensemble is needed and its advantages over a single model with the lowest (best) training loss, we ran a bootstrapping experiment and trained 100 unique models. Each new model was trained with a new seed, and each epoch was trained with a different augmentation of the training set.

We further hypothesized that a surrogate measure of confidence for the assigned class could be approximated from the votes by the ensemble of models. For each cell, we selected the mode of the votes as the true class and the percentage of models that voted for that class as a measure of how confident the ensemble as a whole was of that particular prediction.

All networks were trained using PyTorch 1.7.1 [27].

3.5.4 Extreme gradient boosting (XGBoost)

For XGBoost [17] we used most of the default parameters, of note, seed per iteration True, tree depth of 7, eta 0.1, regularization alpha 0, regularization lambda 1, learning rate 0.05, objective 'multi:softprob' and number of rounds 200. This resulted in an ensemble of 1200 trees (given by the product of the number of classes and the number

of rounds) which are weak classifiers, each tree being more specialized on the difficult training examples than the previous trees.

In order to obtain a measure of confidence for the XGBoost method, analogously to the FCNNs, we require an ensemble of models. We observed that even though an XGBoost model consists of an ensemble of decision tree models, they are neither independent nor equally weighted and therefore their agreement is not suitable as a measure of confidence. Instead we trained 100 XGBoost models and once more used the mode of their votes as the assigned class and the percentage of votes for the assigned class as a measure of confidence.

All XGBoosts were trained using the XGBoost Python API implementation version 1.1.1.

3.5.5 Model implementation

FCNNs and XBGoost are not limited to a single programming language or software; what we present is a methodology which can be implemented in any existing tool. We would also like to mention that QuPath has some alternatives for the initial labeling, instead of using several single measurement classifiers, Random Forests can be used on cores individually. However using Random Forests inside the software does not allow the user to determine which features to use for final classification.

If customized code is used outside of any particular software then more freedom of exploration is gained at the cost of increased technical difficulty. For the creation of the FCNN ensamble and XGBoost we used the popular programming language Python and the Pytorch library for Deep Learning. To obtain the d90s features we also use Python's specialized image processing library scikit-image and Numpy. If desired, the d90s features could be possibly obtained inside QuPath through the use of scripting.

3.6 Cell niche detection by spage2vec

Spage2vec [18] is a graph neural network-based approach that learns a manifold of local object neighborhoods. The method takes object coordinates and labels as input, meaning that we could directly use coordinates and cell classes as output from the above presented methods. The learning is unsupervised, meaning that no a-priori information on expected neighborhoods was needed. To validate the approach for niche detection, we used the publicly available cell class data provided in [3] and compared to previously presented niche results.

Spage2vec allows for the use of several layers of any chosen size. We used the same parameters for our data and for CODEX data: two layers, sizes 50 and 2. This means that the output is a 2D point in space. We used an attention aggregator which results on the placement of these 2D points in a circle. Each point is separated from the others based on its difference in neighborhood. Distances are measured in terms of angles and cosine distance.

The minimum distance that connects any 2 cells is chosen to be the 90th percentile of all the distances between each nearest cell in the dataset. This results in neighborhoods ranging from 1 to 10 cells.

4 Results

4.1 Comparison of classification accuracy

To evaluate classification performance, we used the designated test subset of 10 cores manually annotated with labels by two expert annotators as described in the previous sections. We regard annotations by expert 1 as ground truth and compare accuracy i.e. correctly labeled over the total labeled, to the fully automated classification results produced by the FCNN and XGBoost ensembles, as well classification results obtained by average thresholds. We also compare to classification results obtained by manual classification by a second person, referred to as expert 2, see Figure 1.

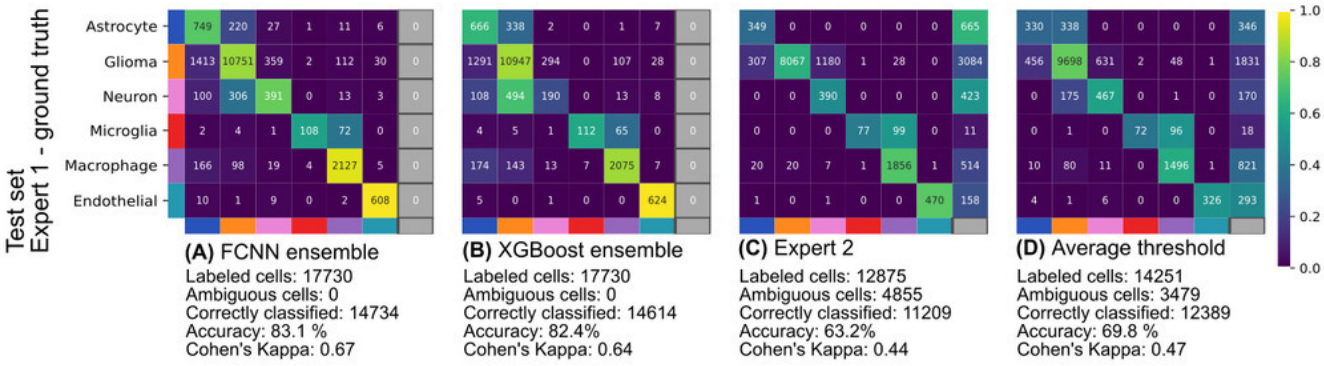
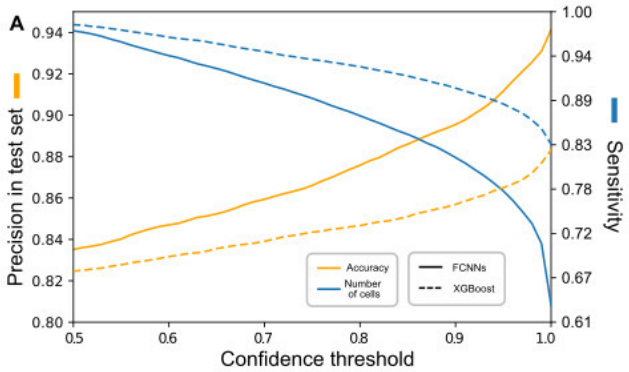


Figure 1: Confusion matrices comparing ground truth (as defined by expert 1) versus A) FCNN ensemble, B) XGBoost ensemble, C) human expert 2, and D) classification by average thresholds. The right-most column of each confusion matrix shows the number of cells not assigned to a class. The color coding in the confusion matrices reflect the proportion of the total number of cells in each class, including cells marked as ambiguous by expert 1. Accuracy measurements and Cohens Kappa are calculated only for cells in the test set that were assigned a class by expert 1.

When taking into account all the cells in the test set and using the full set of features including d90s, the FCNN ensemble reached 83.1% of accuracy while the XGBoost ensemble achieved 82.4% as compared to expert annotator 1. Classifying cells by applying averages of thresholds from the training set resulted in an accuracy of 69.8% on the test set (as compared to non-ambiguous cells in the ground truth by expert 1), and as many as 3479

out of the 17730 of the cells in the test set were left without an assigned class, as shown the confusion matrices in Figure 1. It can also be observed that FCNN ensemble, as well as XGBoost ensemble, assigns a class to every cell (leaving no ambiguous cells) while expert 2 as well as classification by averaged thresholds (1 C and D) result in many ambigous cells, and thus reduced recall. The bright diagonals in both 1 A and B indicate that FCNNs and XGBoost ensembles have high accuracy and recall, but also that they have some difficulty separating glioma cells from neurons and perhaps microglia from macrophages.

4.2 Confidence thresholding increases precision at the cost of sensitivity



B

	Model	Precision (%)	Precision in maximum confidence (%)
with d90s	FCNN ensemble	**83.1**	**94.47**
	XGBoost	82.4	88.3
without d90s	FCNN ensemble	80.8	-
	XGBoost	82.0	-

Figure 2: Confidence scores and effect of d90s features. A) Precision and sensitivity at varying cell class confidence thresholds. B) Summary of precision achieved with and without d90s features and when using only maximum confidence.

Using ensembles of classifiers allowed us to provide a measure of the confidence of the class voted by the ensemble. Note that all cells above the confidence threshold will be assigned a class, which may be a true (TP) or a false (FP) as compared to the class assigned by expert 1. Cells that fall below the confidence threshold are regarded as false negatives (FN). Figure 2A shows that applying an increasing confidence threshold increases the classification precision (TP/(TP+FP)), while at the same time decreasing sensitivity (TP/(TP+FN)). Thus, the confidence score can be used as a criterion to exclude or to weight cells and their contributions to further downstream analysis of cell niches. In Figure 2A it can be observed that the FCNN ensemble achieved its highest precision value of 94.47% when the confidence threshold was set to the maximum (1.0) while the XGBoost ensemble reaches 88.3% meaning that it has less discriminative power and regardless of the cost, no higher precision can be achieved.

4.3 Feature engineering

We studied the benefit of including the d90s features in relation to the performance of both machine learning models. We trained 100 components per model, one including the d90s and one that only included common marker intensity features. Each ensemble architecture used the same parameters for both sets of features. Both ensembles increased in precision when presented the d90s features. Without d90s, the FCNNs had a reduced precision of 80.75% and the XGBoost remained close at 82% (with only a drop of 0.4%). This showed that the d90s features helped the classification mostly for the FCNNs. Figure 2B summarizes the precision of each ensemble with and without d90s features.

4.4 Ensembles perform better than single models

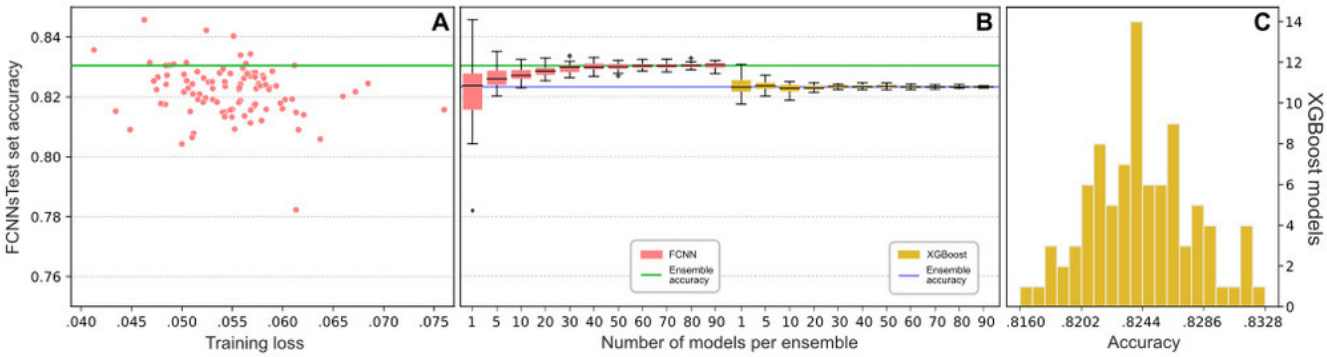


Figure 3: A) Scatterplot relating the training loss versus test accuracy of one hundred unique FCNNs trained to explore the solution landscape. B) Boostrap experiment to estimate the expected test accuracy using different amount of models in the ensembles. C) Histogram of accuracies amongst XGBoost models.

FCNNs and XGBoost are very different architectures and even if they achieved similar performance they do so for different reasons. We explored how a single model arrives to a conclusion and what expected value of accuracy can be reached if we could train an infinite number of models.

Figure 3A shows the relationship between training loss and test set accuracy for each of 100 trained FCNNs. Using the annotated test set we could conclude that there was no correlation between training loss and test set accuracy, meaning that the FCNN with the lowest training loss can not be expected to perform well on the test set (correlation coefficient of -0.22). Additionally, the density of networks above the ensemble accuracy is very low, meaning that the likelihood of obtaining an FCNN that can outperform the ensemble is very low.

To estimate the effect of varying the number of models in the ensemble, we selected 30 random ensembles for each considered number of models (30 ensembles of sizes 1, 5, 10 etc) and quantified their accuracy, displaying the resulting accuracy distributions in Figure 3B. We observed that an ensemble of 30 FCNNs or more yields an expected value of 83.1% accuracy on the test set, and that increasing the number of networks further does not increase the accuracy significantly. For XGBoost the expected value does not change significantly and remains around 82%. Based on training loss, XGBoost always achieved 0 loss so we cannot choose a model based on training loss, instead, we show in 3C a histogram of the number of models per accuracy. Low and high accuracies in the XGBoost range have lower probability of appearing so we can expect a total accuracy of 82% with little to no variability which means that almost all XGBoost models arrive to the same conclusions which is expected given the nature and limitations of the architecture.

4.5 Towards cell niche analysis using spage2vec

To verify spage2vec as a method for detecting cell niches we performed an exploratory analysis and comparison with previously studied cells and tissue in [3]. The results, presented in Supplementary Figure S4 show that spage2vec is able to find representative cell niches as those identified in the original publication.

Having confirmed that spage2vec works on single cell data we proceeded to use the method and the output from our cell classification to find cell niches in our TMA cores. Resulting common and structured cell niches can be observed in Figure 4, where similar colors correspond to similar niches. The angle histogram of the spage2vec values shows that some niches are more common than others. The peaks in the spage2vec histogram coincide with peaks in the mean class representation in the niche histogram. A detailed example is shown in Figure S3. This indicates that spage2vec is able to describe a niche with a single value and is able to organize niches as evenly as possible, capturing distinct class compositions. Whenever no clear peak is observed in spage2vec it shows that the niche composition has a high standard deviation and therefore is not a homogeneous representation of classes in the niche and probably should not be a cluster of its own.

5 Conclusions and discussion

We have presented a methodology for cell classification using two popular machine learning architectures FCNN and XGBoost. Accurate cell classification is necessary for further analysis of tissue architecture and cell niches. We also presented the use of the spage2vec[18] method for the analysis of spatial cellular composition of tissue on the results from our classification.

High throughput image analysis of TMAs generally requires sample standardization or working on samples individually. The proposed methodology using ensembles is able to learn sample distributions, thus circumventing standardization difficulties. We presented the d90s features that capture information about relative cells marker profiles, leading to an increase in accuracy. From all XGBoost models, some of the d90s features are in the top 20 (out of 68) features in the measures of gain, coverage and weight (Figure S7), showing that they are indeed important features for decision making.

We show that ensembles of classifiers arrive to similar conclusions as our two expert annotators which encourages us to continue using this method to classify larger cohorts of TMA cores and avoid manual thresholding. However, Figure S5, cores 1 and 9 show a confusion between the glioma class (orange) and neurons (pink). It is important to take biologically relevant information into account, and it is not expected for neurons to be present in such a high amount in this dataset. Even though many cells were labeled as neuron given the presence of the NeuroC marker, visual inspection confirmed that they were not all neurons. More specific markers would be required to correctly distinguish some of the cell classes.

We showed that using an ensemble is better than using a single model and that the voting scheme can be converted to a measure of cell classification confidence. We also showed that within the highest confidence classifications there is indeed higher classification precision.

Confidence thresholding could potentially be used to flag cells that do not belong to the cell class expected in the tissue. If high precision is preferred at the cost of removing cells from analysis (i.e. lower sensitivity) then the confidence is a good parameter for thresholding. If a higher number of classified cells are preferred, the confidence threshold can be lowered and more cells are included. When the confidence is visualized as a colored map (supplementary FigureS6) we observed that the confidence is distributed along all cores and within the cores; it is not localized. This shows that our confidence measure is independent on whether a core belongs to training or testing sets and it does not follow any spurious pattern, increasing our trust on the map and the measure itself.

After comparing FCNN and XGBoost and exploring what both methods offer one could consider adapting strategies from both architectures. XGBoost gradually focuses on hard examples. A similar strategy could be used in FCNNs by implementing a weighting scheme where sample weights change as samples become easier or harder to classify. Also after feature exploration with XGBoost one could perhaps rethink the choice of features for the FCNN and possibly increase accuracy.

Finally we show that our cell classification allows us to perform further spatial analysis of the cell niches and allows us to create a grouping of TMA cores. Near future work includes using the proposed cell classification and spatial analysis techniques to explore larger cohorts of TMA cores and relate them to clinical data.

6 Acknowledgements

Prof. Anja Smits, MD, PhD, Inst. of Neuroscience and Physiology, Dept. of Clinical Neuroscience, University of Gothenburg, Sahlgrenska Academy, Sweden, is acknowledged for facilitating the study and for contributing to creation of the tissue collection.

7 Funding

Financed by the European Research council via ERC: Ref 682810 to C.W. and the Swedish Foundation for Strategic Research [grant BD150008]. Swedish Research Council (Vetenskapsrådet), Swedish Cancer Society (Cancerfonden).

8 Conflict of interest

The authors declare no conflict of interest

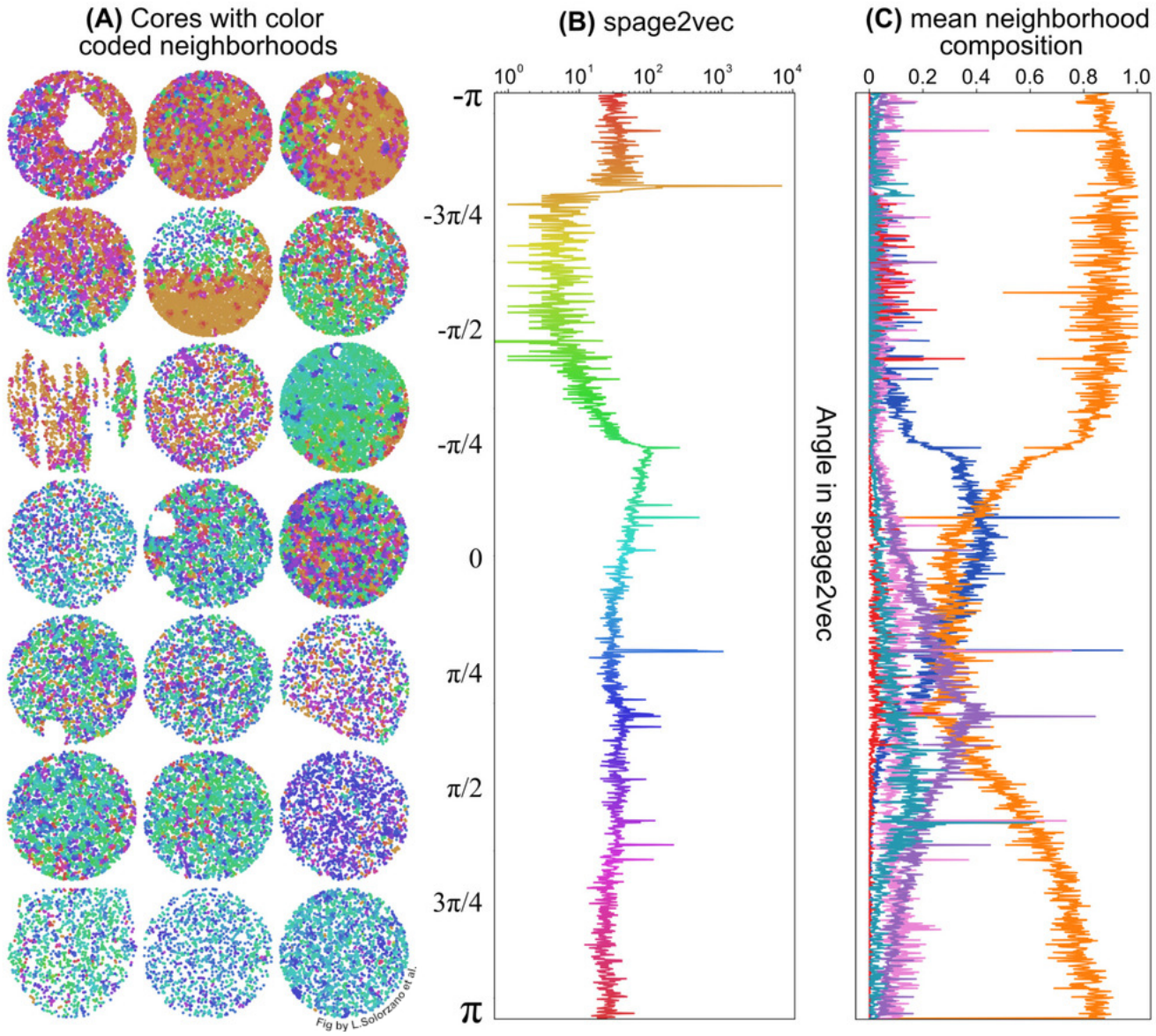


Figure 4: Cell niches defined by spage2vec on glioma TMA cores. A) Each cell is colored with the representation of the niche around it. B) Spage2vec angle histogram. Each cell's niche is projected to a 2D circle space using spage2vec, its angle, location and distance to other points in the circle, represent differences between neighborhoods. C) Normalized mean niche composition in a bin centered in the same angle as spage2vec. The results is an organized list of possible niche types from which significantly different niches can be identified by the peaks in spage2vec.

References

[1] Hartland W. Jackson, Jana R. Fischer, et al. The single-cell pathology landscape of breast cancer. *Nature*, 578(7796):615–620, Feb 2020.

[2] Edwin Parra, Alejandro Francisco-Cruz, et al. State-of-the-art of profiling immune contexture in the era of multiplexed staining and digital analysis to study paraffin tumor tissues. *Cancers*, 11(2):247, feb 2019.

[3] Yury Goltsev, Nikolay Samusik, et al. Deep profiling of mouse splenic architecture with CODEX multiplexed imaging. *Cell*, 174(4):968–981.e15, aug 2018.

[4] Mark A. J. Gorris, Altuna Halilovic, et al. Eight-color multiplex immunohistochemistry for simultaneous detection of multiple immune checkpoint molecules within the tumor microenvironment. *The Journal of Immunology*, 200(1):347–354, nov 2017.

[5] Laurence Buisseret, Sandra Pommey, et al. Clinical significance of cd73 in triple-negative breast cancer: multiplex analysis of a phase iii clinical trial. *Annals of Oncology*, 29(4):1056 – 1062, 2018. Quantification of CD73 and its significance Epigenetic modifiers as immunomodulatory therapies in solid tumours.

[6] Guanzhang Li, Yiming Li, Xing Liu, et al. Aldh1a3 induces mesenchymal differentiation and serves as a predictor for survival in glioblastoma. *Cell Death & Disease*, 9(12):1190, Dec 2018. Immunofluorescence, used to explore ALDH1A3 bio-functions in Glioblastoma. Co-expression with CD44.

[7] Qian Chen, Xiao-Wei Zhou, et al. Actn1 supports tumor growth by inhibiting hippo signaling in hepatocellular carcinoma. *Journal of Experimental & Clinical Cancer Research*, 40(1):23, Jan 2021. Imunofluorescence performed to identify underlying molecular mechanism of previously segmented cells.

[8] Jarem Edwards, James S. Wilmott, et al. Cd103+ tumor-resident cd8+ t cells are associated with improved survival in immunotherapy-naïve melanoma patients and expand significantly during anti–pd-1 treatment. *Clinical Cancer Research*, 24(13):3036–3045, 2018.

[9] Katherine Sanchez, Isaac Kim, et al. Multiplex immunofluorescence to measure dynamic changes in tumor-infiltrating lymphocytes and PD-l1 in early-stage breast cancer. *Breast Cancer Research*, 23(1), jan 2021. FMIHC to find immune cells and use this as validation for other immuno-oncology biomarkers.

[10] Itay Tirosh, Benjamin Izar, et al. Dissecting the multicellular ecosystem of metastatic melanoma by single-cell rna-seq. *Science*, 352(6282):189–196, 2016.

[11] Amélie Viratham Pulsawatdi, Stephanie G. Craig, et al. A robust multiplex immunofluorescence and digital pathology workflow for the characterisation of the tumour immune microenvironment. *Molecular Oncology*, 14(10):2384–2402, sep 2020.

[12] Peter Bankhead, Maurice B. Loughrey, et al. QuPath: Open source software for digital pathology image analysis. *Scientific Reports*, 7(1), dec 2017.

[13] Benjamin R Challoner, Katharina von Loga, et al. Computational image analysis of t-cell infiltrates in resectable gastric cancer: Association with survival and molecular subtypes. *JNCI: Journal of the National Cancer Institute*, 113(1):88–98, apr 2020.

[14] Chung-Wein Lee, Yan J. Ren, et al. Multiplex immunofluorescence staining and image analysis assay for diffuse large b cell lymphoma. *Journal of Immunological Methods*, 478:112714, mar 2020.

[15] Claire McQuin, Allen Goodman, et al. CellProfiler 3.0: Next-generation image processing for biology. *PLOS Biology*, 16(7):e2005970, jul 2018.

[16] Francesco Ciompi, Oscar Geessink, et al. The importance of stain normalization in colorectal tissue classification with convolutional networks. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. IEEE, apr 2017.

[17] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA, 2016. Association for Computing Machinery.

[18] Gabriele Partel and Carolina Wählby. Spage2vec: Unsupervised representation of localized spatial gene expression signatures. *The FEBS Journal*, 2020.

[19] Mariko L. Bennett, F. Chris Bennett, et al. New tools for studying microglia in the mouse and human CNS. *Proceedings of the National Academy of Sciences*, 113(12):E1738–E1746, feb 2016.

[20] Stéfan van der Walt, Johannes L. Schönberger, et al. scikit-image: image processing in Python. *PeerJ*, 2:e453, 6 2014.

[21] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011.

[22] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 448–456. PMLR, 07–09 Jul 2015.

[23] Juntang Zhuang, Tommy Tang, et al. AdaBelief optimizer: Adapting stepsizes by the belief in observed gradients. *NeurIPS*, 2020.

[24] Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145 – 151, 1999.

[25] Lars Kai Hansen and Peter Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001, 1990.

[26] Padraig Cunningham, John Carney, et al. Stability problems with artificial neural networks and the ensemble solution. *Artificial Intelligence in Medicine*, 20(3):217 – 225, 2000.

[27] Adam Paszke, Sam Gross, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32:8026–8037, 2019.

9 Supplementary

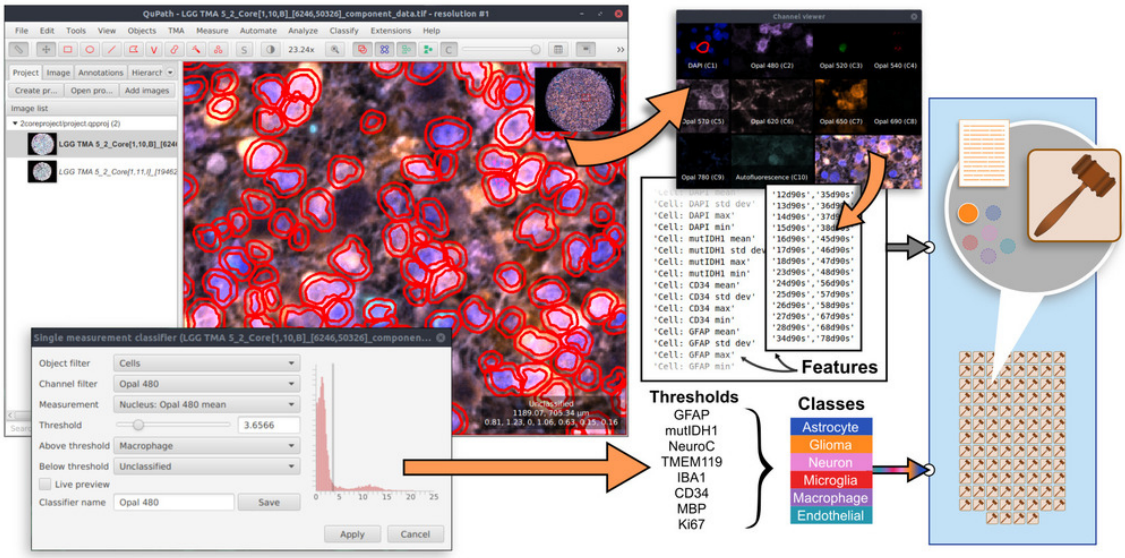


Figure S1: Creating training data: Example of the QuPath interface to perform cell segmentation of a single core and manually select thresholds per marker to generate cell classes. The cell segmentation masks are also exported to enable extraction of the d90s features outside of QuPath. These features are input to the FCNN and the XGBoost ensembles.

Main cell classes	GFAP	mutIDH1	NeuroC	TMEM119	IBA1	CD34	MBP	Ki-67	Possible sub-classes
Astrocyte	o							x	2
Glioma	x	o	x					x	8
Neuron			o						1
Microglia				o	x			x	4
Macrophage					o			x	2
Endothelial						o		x	2

Figure S2: Marker to class lookup table. “O” means main marker while “x” means possible additional marker. A threshold must cross the “O” marker and any “x” marker to belong to the marked class.

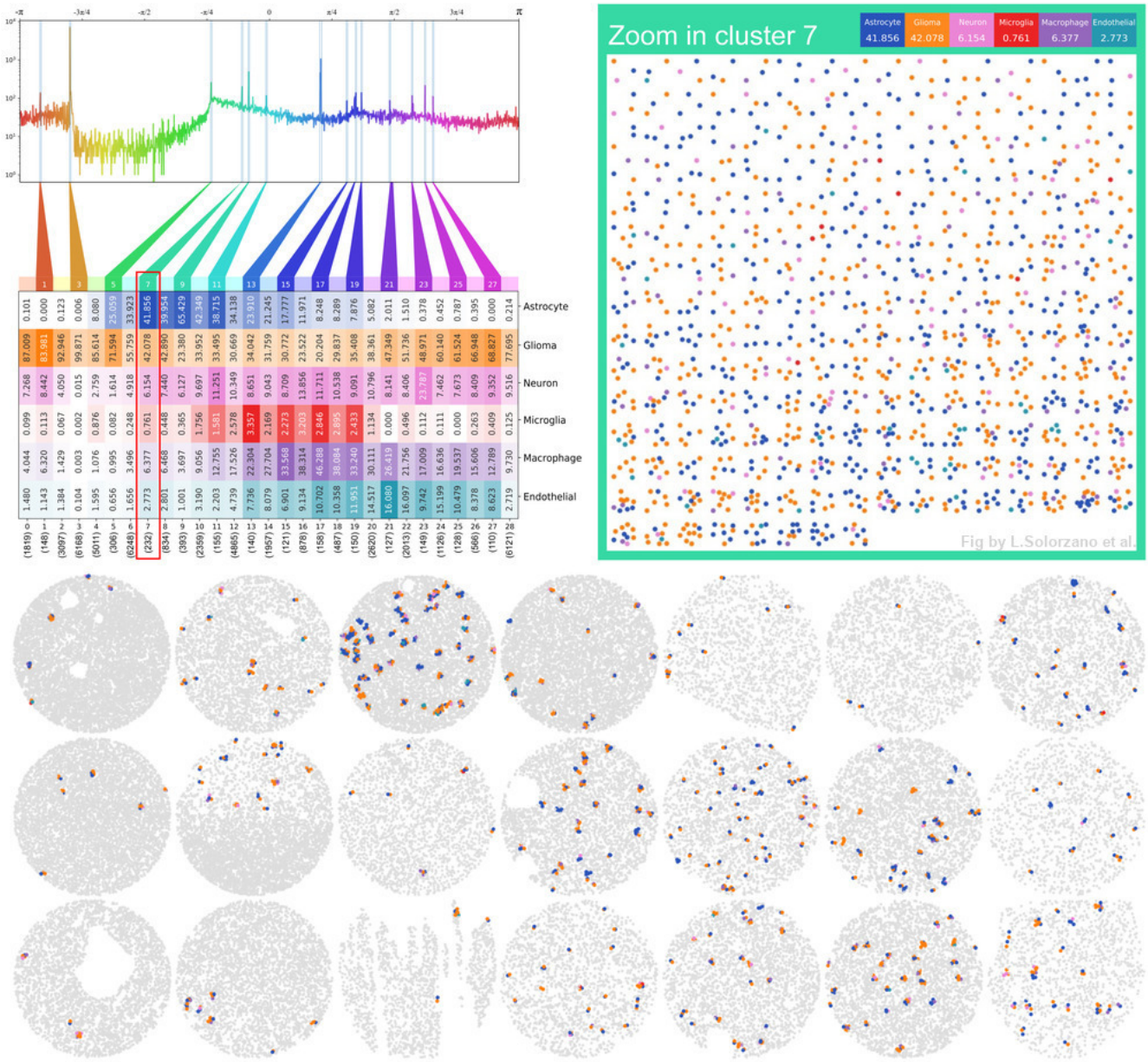


Figure S3: Spage2vec cluster zoom-in for glioma data. Peaks in the spage2vec angle histogram can signal clusters that stand out. Zoom into cluster 7 which contains a peak that present neighborhoods that have 41% chance of having astrocytes, 42% chance of having glioma, 6% of having neurons, 6% of having macrophages, 2.7% of having endothelial and 0.7% of having microglia. Clusters are shown individually and in their spatial location in the cores.

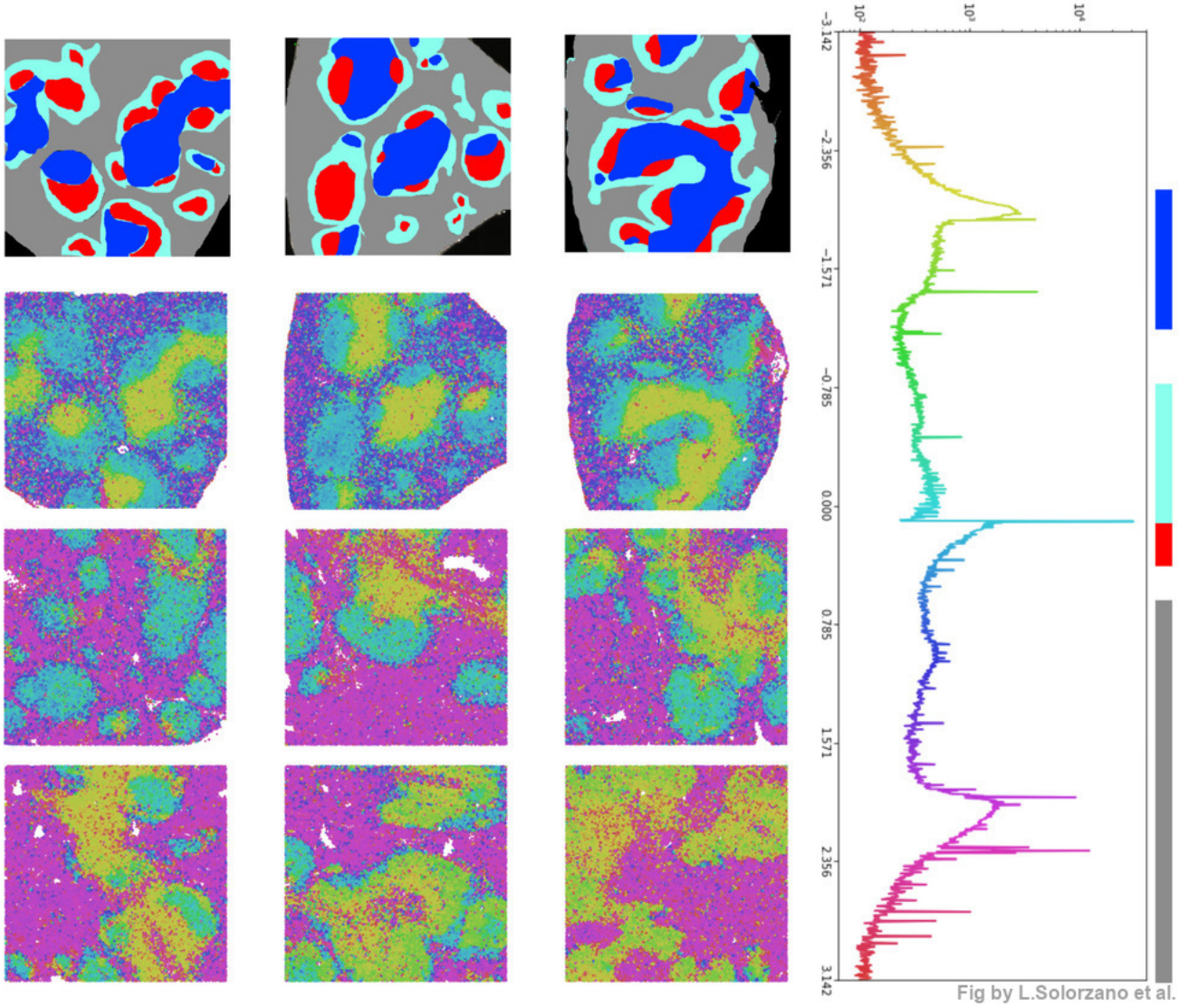


Figure S4: Spage2vec on CODEX data. Same groups of tissue compartments are found. Cells are grouped according to their neighborhood. Neighborhoods are placed in a polar space. The histogram of angles (represented by hue) shows populations per angle, displaying several peaks which correspond visually to groups found in the image. First row of tissue compartment images belongs to the authors of the CODEX paper in [3].

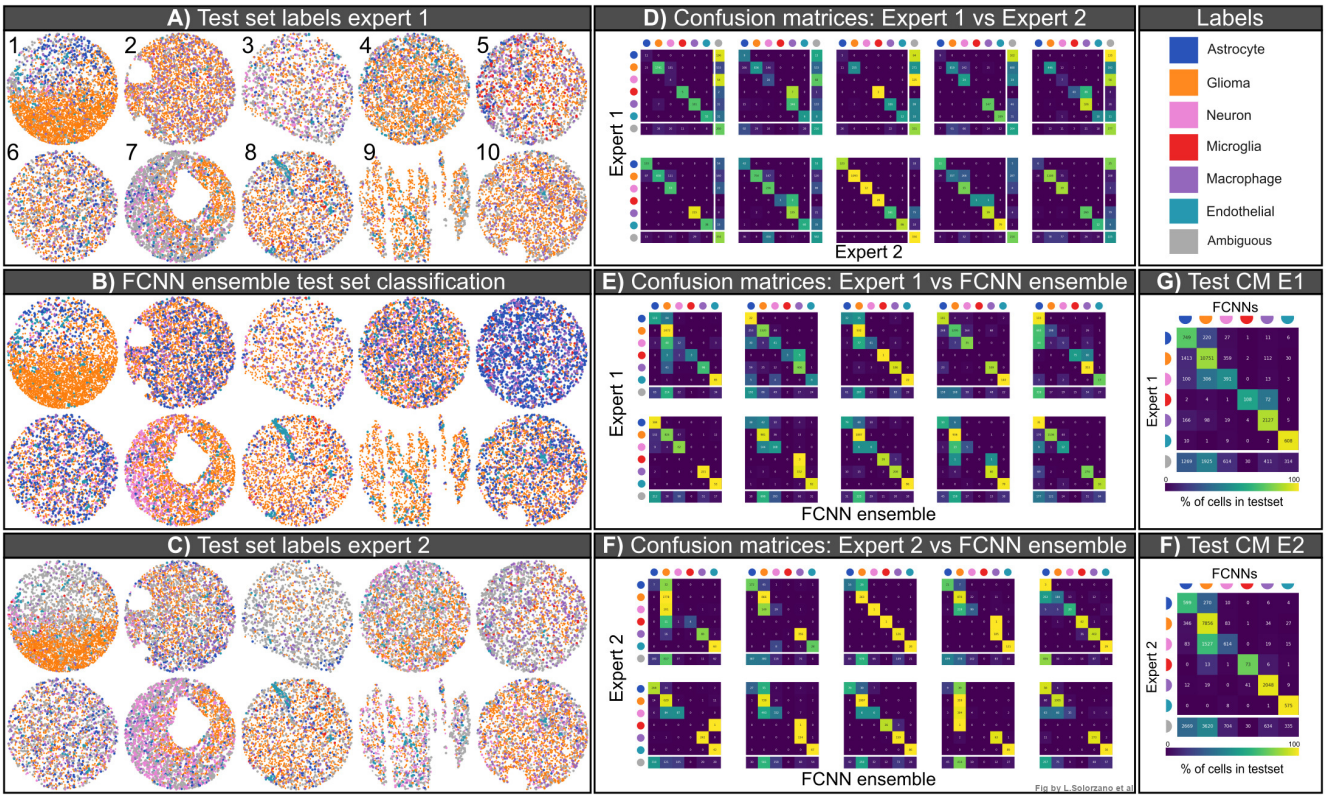


Figure S5: Cell classification results on the 10 independent test cores compared amongst experts and the FCNN ensemble. A) Cell classes from manual/visual annotation by expert 1. B) Cell classification results by the FCNN ensemble. C) Cell classes from manual/visual annotation by expert 2. D) Confusion matrices per core showing the agreement between expert 1 and expert 2. Notice the number of ambiguous cells provided in the bottom row in each confusion matrix, and how they reduce the number of agreed cells. E) Confusion matrices between expert 1 and the FCNN ensemble. F) Confusion matrices between expert 2 and the FCNN. G) Summary confusion matrix between expert 1 and the FCNN. F) Summary confusion matrix between expert 2 and the FCNN.

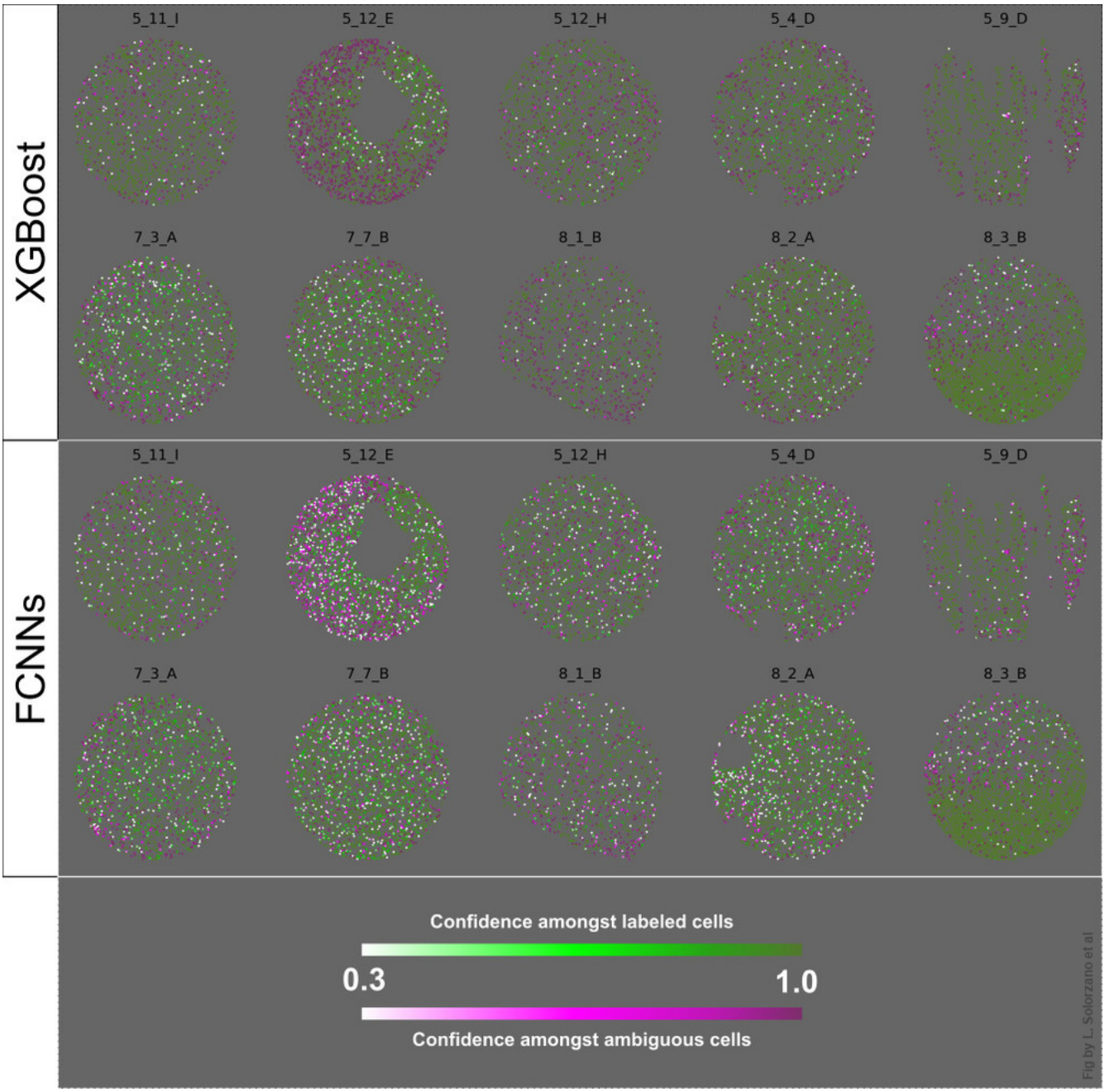


Figure S6: Colored confidence map for test cores. The color LUT is on purpose highlighting lower confidence cells. It can be observed confidence is spread across cores, for both methods. Green represents labeled cells by expert 1, and purple represents ambiguous cells according to expert 1.

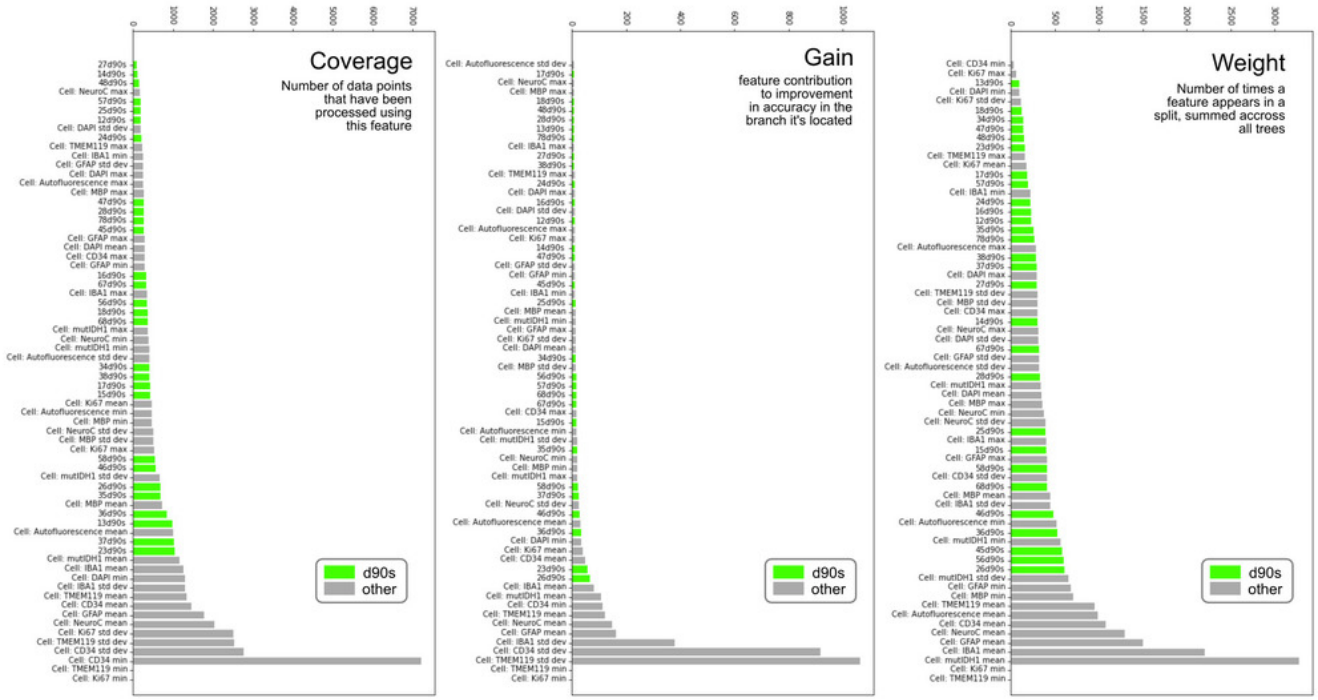


Figure S7: XGBoost measures of feature importance plotted for the average of each feature amongst all the trained 100 models. The importance of d90s can be observed in green, it is spread, not all d90s are equally important. Note that in all gain, weight and cover, the means of the markers have generally more importance than any other feature.