Predicting lipid abundance in a murine brain section from spatial gene expression

Lusine Khachatryan, Jules Perrin, Viola Renne
Tutor: Luca Fusar Bassini, Halima Hannah Schede
Supervisor: Gioele La Manno
Laboratory of Brain Development and Biological Data Science, EPFL, Switzerland

***Abstract*—Understanding how lipids distribution is regulated in the brain is a key task for the analysis of the brain functioning in health and disease. In this work, we investigate the predictability of various brain lipids distribution using spatial gene expression data. We present a method to aggregate non-uniformly distributed gene expression data around the lipid measurement area and test obtained features using various Machine Learning techniques. Interpretation of built models allows to determine the impact of genes, lipid category, and brain region on the predictive quality for analysed lipids.**

I. Introduction

Lipids are key molecules for membrane formation, cell signalling, energy storage, and homeostasis maintenance [1]. The concentration of various lipids is especially high in the brain, where lipid molecules play an important role in organ functioning. Many studies demonstrated a connection between brain lipids deregulation and the progression of many different neurological pathologies [2]. Thus, understanding how brain lipids are regulated is crucial for the early diagnosis and prognosis of brain diseases as well as for investigating the biological functions of the brain and its specific regions.

Spatial transcriptomics is a gene expression profiling method allowing one to quantify the total counts for hundreds of RNA molecules at a given time point in space in a tissue sample and to create detailed molecular maps reflecting changes in gene expression levels. Integrating brain spatial transcriptomics data with brain lipidomics using **M**achine **L**earning (**ML**) techniques could reveal how gene activity modulates the distribution of different lipids across the brain.

In this study our objective is to accurately predict the abundance of 156 brain lipids, each measured in 89,395 points across the mouse (*Mus musculus*) brain using the spatial expression of 500 different genes measured in 186,090 points (here and after called genes nuclei). We conclude that the abundance of most analysed lipids can be accurately predicted using spatial transcriptomics data. Prediction quality is strongly associated with the number of nuclei selected for feature aggregation and with the aggregation strategy. We test several popular ML techniques and different data generation strategies to obtain robust and interpretable models. Finally, we analyse obtained models to reveal the impact of individual genes, lipid categories, and brain regions on the prediction quality of certain lipids.

II. Models and Methods

A. Data

Analysed data is tabular, for its handling pandas [3] DataFrames format is used. The abundance of 156 lipids are generated using mass spectrometry in 89,395 points across the brain section from a single *Mus musculus* brain. For some lipids, the level of technical noise is higher than the signal level. These lipids (18 in total) are detected by analysis of standard deviation per lipid (< 0.00011) and eliminated from the further analysis (see Appendix for detailed explanation). The final number of analysed lipids is 138.

Gene expression data is obtained with spatial transcriptomics analysis targeted on the 500 most variable genes expressed in the brain. These genes are measured in 186,090 points across the same brain section as for lipidomics data. Measured points for gene expression are non-uniformly distributed across the brain section.

For the lipidomics data, the absence of the measurement in a particular point is replaced with a -9.21 value, obtained data is exponentiated. Gene expression measurements are log-transformed. These normalisation strategies are selected after controlling prediction performance using several initial ML fittings (see ML methods II-C for more details) on a random subgroup of lipids.

B. Feature Engineering

The strategy used for gene nuclei selection is based on constructing a cKDTree [4], a spatial data structure optimised for fast neighbour querying. This approach focuses on selecting a constant number of neighbouring nuclei around each lipid, based on their spatial coordinates. This method is favoured due to its simplicity in iterative optimisation and the fixed number of nuclei it provides for analysis, allowing for a consistent and streamlined evaluation process.

The following averaging techniques are implemented to aggregate gene expression data obtained from all selected nuclei:

- Simple average
- Weighted average using Gaussian decay with distance from the lipid centroid

- Weighted average using Negative Logarithmic (NL) decay with distance from the lipid centroid

C. ML methods

The task is a multi-regression problem, as it involves predicting values on a continuous scale. Therefore, the Coefficient of Determination, R^2, is used to evaluate the prediction quality of a model.

Models initial fitting, evaluation, and tuning are performed strictly using a training subset (70%) of data. Testing data subset (30%) is saved only for the final evaluation of the selected models.

1) Model fitting and optimising using PyCaret: PyCaret [5] package is used for the initial fitting of various ML approaches and tuning selected ones. PyCaret ML model fitting is performed separately for each of the analysed 138 lipids.

Function *setup* with random seed 42 divides the initial dataset into training and testing subsets, adhering to a 70/30 split. Subsequent *compare_models* function performs the initial fit of 25 popular ML techniques (see the complete list in the Appendix). To control the influence of overfitting, 5 fold Cross Validation (CV) 5 fold is used. The returned models are fitted on the entire train dataset.

The hyper-parameter optimisation for the selected PyCaret ML approaches is performed using Optuna [6] framework with 5 CV folds and custom tuning grid (Table I).

ML Model	Parameters	Tuning Grid
CatBoost	LR	0.1, 0.05, 0.01, 0.005, 0.001
	Depth	2, 4, 6, 8, 10
	L2 Reg	1, 3, 5, 7, 9
KNN	# neighbours	$(2, 3, ..., 20)$
MLP	LR	0.01, 0.001, 0.0001
	Initialization	HeNormal, GlorotUniform
	Activation	sigmoid, tanh, ReLU, GELU
	# of layers	2, 3, 4

Table I: *Hyper-parameters tuning summary.*

2) MLP: A Multi-Layer Perceptron (MLP) is developed utilising Keras [7] TensorFlow [8]. It is a fully connected feed-forward neural network comprising 3 hidden layers and a total of 79,018 parameters. The network has 500 nodes in the input layer, corresponding to the genes, and 138 nodes in the output layer, corresponding to the lipids. The lipid data is normalised through the application of a min-max scaler, so that during back-propagation the neural network allocates equal significance to every output node, given the varied magnitudes of lipids abundances. Sigmoid activation function is used for both hidden layers and output layer since once the data are scaled with the min-max scaler, they are in the range of 0-1. Each layer's weights are initialised with GlorotUniform [9]. During training, the learning rate (0.001) is reduced after each epoch. The tuning parameters are shown in I (see Appendix for more details).

D. Model interpretation and analysis

1) Features importance analysis: Features importance obtained for each lipid using CatBoost are used to calculate:

- pairwise Jaccard distance [10] between lipids using per-lipid most important features (comprising 50% of the model importance).
- pairwise correlations using all features importance (calculated with *pdist* [11] function of scipy package).

2) R^2 residuals analysis: Initial CatBoost models are used to predict lipid abundances for the test subset of data. R^2 residuals for each test lipid point are stratified per region according to Allen Brain Atlas [12], summed up and divided by the total sum of the residuals and the number of points associated with the region. This is implemented to mitigate biases arising from variations in region sizes and the varying prediction quality of different lipids, consequently affecting the total sum of residuals across the test subset of data.

III. Results

A. Defining an optimal strategy for feature engineering

Since gene expression nuclei are non-uniformly distributed throughout the brain section, the points at which lipid abundance should be predicted may exhibit variability in the density of nearby gene expression spots. Managing this variability emerges as a crucial aspect of tackling the prediction task. We use *n*-nearest neighbours strategy changing *n* from 1 to 10,000 with the following gene expression data aggregation across all the neighbours. We compare three aggregation approaches: simple average, Gaussian decay, and NL decay.

The performance for each combination of the number of neighbours and aggregating approaches is estimated by running an initial fitting of several ML methods and selecting the best observed R^2. This analysis allows the selection of the optimal *n*-aggregation combination and demonstrates which of the initially fitted ML methods should be considered for further analysis.

As shown in Figure 1, the overall performance grows for all aggregation techniques with the increase of *n*, with the plateau phase starting around *n* equal 1,000. Therefore *n* equals 1,000 is selected for further work.

Among the aggregation techniques, for all observed *n* values, implementation of Gaussian and NL decays is associated with higher performance. Because NL decays are easier to calculate they are selected for further analysis.

Among all the initially fitted methods, KNN and CatBoost demonstrate the highest R^2 results coupled with the smallest R^2 standard deviation across folds (see Appendix) and therefore are selected for further fine-tuning. Another well-performing algorithm, XgBoost, is not stable across folds (see Appendix) and therefore not selected for tuning.

ML Model		CatBoost	KNN	XgBoost	MLP
Initial	mean	0.5620	0.5268	0.5206	0.4053
	std	0.1873	0.2175	0.1958	0.0038
	median	0.6034	0.5824	0.5648	0.4070
Tuned	mean	0.5622	0.5512	-	0.5434
	std	0.1871	0.1969	-	0.0078
	median	0.6036	0.5971	-	0.5423

Table II: R^2 *(initial and tuned) of top-performing PyCaret methods and MLP. Only results obtained for data generated with NL aggregation on n = 1,000 neighbourhood are shown.*

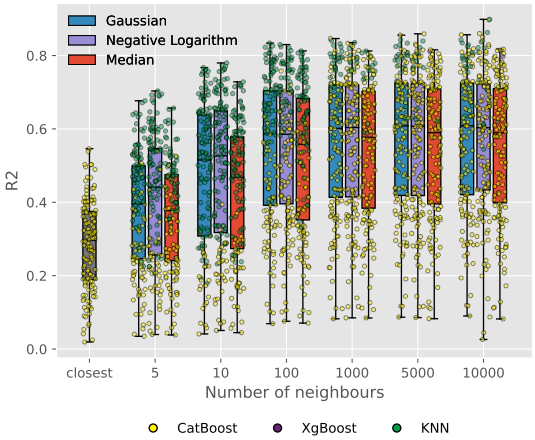

Figure 1: *Analysis of the number of neighbouring nuclei (X-axis) and aggregation approach. The performance for each of the three implemented aggregation techniques is shown as a boxplot. Performance is estimated using the highest R^2 from initially fitted PyCaret models. Individual scores are scattered across the corresponding boxplot, dot colour represents the best scoring algorithm.*

B. Tuning

Tuning CatBoost models does not demonstrate drastic improvement (Table II) and is time-consuming. Thus, the initial CAtBoost models remain in the analysis pipeline. For KNN larger improvement is observed during the tuning (average R^2 growth 0.0243 ± 0.0237). The final comparison of tuned models reveals that the vast majority of lipids have a larger R^2 when predicted using initial CatBoost. When KNN outperforms CatBoost the difference is quite small (0.0034 ± 0.0027). With the PyCaret package, only standard linear and tree-based techniques can be used. It is interesting to observe how Deep Learning techniques, like MLP, would approach this particular task. Such model would not be easily interpretable, but its results can improve the credibility of the results obtained with PyCaret models. After hyperparameters tuning MLP performance becomes comparable to one for CatBoost and KNN, R^2 values of MLP and CatBoost correlate (Figure 4). CatBoost provides importance for each analysed feature, thus to facilitate further model interpretation and results aggregation, initial-CatBoost models are used for further analysis.

The quality of the final models (one for each lipid) is assessed using the test dataset (30% of the original data). The average R^2 is 0.5669 ± 0.1866, however, the distribution of R^2 values is skewed to the right, thus the median R^2 (0.6042) is a more relevant measurement in this case. We notice during the analysis on both training and testing data subsets that CatBoost performance quality correlates with the standard deviation of the lipid abundances (Figure 4) and does not correlate with the total number of NA measurements for the lipid (see Appendix for more details).

C. CatBoost models interpretation

Analysed brain lipids can be stratified based on the lipid type (more granular) and lipid category (less granular). As can be seen in Figure 2, the distribution of R^2 among different types and categories of lipids can be varied. Certain lipid types (e.g. HexCer and SM) have a higher overall prediction quality. Stratification based on lipids category demonstrates that prediction quality is higher for Sphingolipids in comparison to Glycerophospholipids.

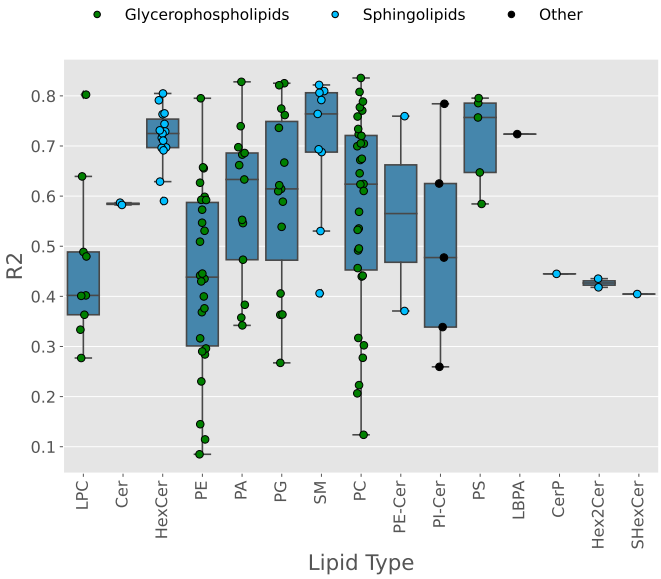

Figure 2: *Distribution of R^2 values obtained after applying CatBoost models on test data. Boxplots represent lipids stratified per lipid type (specified in the X axis), and scattered individual measurements are coloured per lipid category.*

Two different strategies (co-occurrence of the important features based on Jaccard distance and correlation of the features' importance profiles) are implemented to understand whether certain genes are more important for the overall prediction quality across different lipids. Although most of the lipids do not share a large proportion of the important genes, both methods reported one group of lipids (33 for the first strategy and 42 for the second one) for which the same genes set have a high importance. These groups of lipids have a large overlap (30 lipids) and both are enriched with Sphingolipids (see Appendix for more details).

The analysis of residuals obtained for the test subset of the data demonstrates the split of brain lipids into two subgroups (Figure 3). One enriched with Sphingolipids and another enriched with Glycerophospholipids, each associated with certain brain regions where the quality of the prediction is higher (lower weighted average R^2 residual).

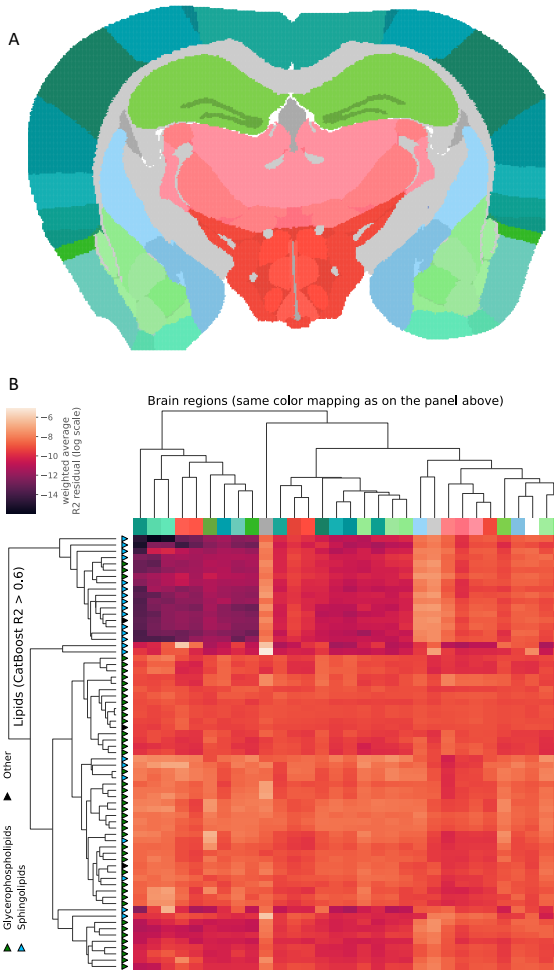

Figure 3: *Impact of the brain region on the prediction quality. Only lipids with Test CatBoost $R^2 > 0.6$ were analysed. Brain pixels are coloured by the brain region (A). Heatmap representing the average balanced R^2 residual per brain region per lipid in logarithmic scale (B).*

IV. Discussion

In this study, we try to reveal the connection between brain lipids distribution and the expression level of brain genes as the first step in explaining brain lipids regulation.

Since the gene expression data is measured only for 500 out of thousands of brain genes, we expect that for a certain proportion of lipids, the accurate prediction would not be feasible due to the simple absence of genes regulating their production. However, for more than half of the analysed lipids, we can achieve a strong predictive performance with

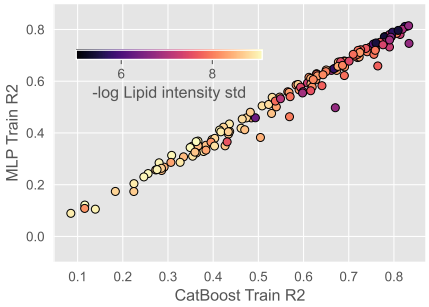

Figure 4: *R^2 values obtained for Initial CatBoost and tuned MLP models. Each dot represents a separate lipid, dot colour reflects lipids abundance std*

$R^2 > 0.6$. Performance varies based on the lipid type and category. The analysis of the R^2 residuals demonstrates that analysed lipids can be split into groups based on the brain region with the most successful abundance prediction. These groups correlate with the observed lipids categories (Glycerophosoho- and Sphingolipids). We also observe a group of lipids (enriched with Sphingolipids) which might be co-regulate since they share a large proportion of genes important for their prediction quality.

A critical aspect of this work is the development of the averaging techniques to aggregate the values of the selected gene nuclei. This study demonstrates that the growth of the number of nuclei considered is improving the prediction quality. This study shows that the proper aggregation technique can balance the influence of proximal and distant nuclei (particularly by penalising distant nuclei to achieve a weighted average reflective of spatial relevance) and can drastically improve the prediction quality. The possible further improvement of this algorithm would be the adaptation of the optimal number of nuclei specifically for each gene as different genes might influence different areas around its expression locus.

An extensive evaluation of a wide range of ML models using PyCaret allows us to identify those with optimal performance. Though CatBoostRegressor has demonstrated outstanding performance during the initial fit, it almost did not react on the tuning attempts. This is observed when tuning all or only certain parameters with the custom grid. This might be caused by a small number of iterations (1,000) used for the tuning process.

We observe that the performance of tuned MLP and initial CatBoost models correlate, which supports the conclusions regarding different predictive power of analysed genes towards analysed lipids. However, MLP provides a slightly lower range of R^2 than the Initial CatBoost models. This is in agreement with previous studies demonstrating that for tabular data analysis, CatBoost is preferable in comparison to MLP [13], [14].

V. Ethics

The results obtained in this study should be interpreted carefully as they might be heavily biased by the lineage, age, and even daily lifestyle of the particular *Mus musculus* used in this experiment. Here we demonstrate just an association between certain lipids and genes, allowing us to predict the distribution of lipids using transcriptomics data. This does not necessarily mean causation and thus model animals should not undergo any experiments involving modulation of observed important genes without additional evidence proving the causation of gene-lipid association.

Furthermore, one might want to extrapolate results obtained when working with *Mus musculus* brains on *Homo Sapiens*. Although *Mus musculus* is a model organism for many human disorders, conclusions obtained using mouse models should not be strong evidence in any human-related brain dysfunction discussions and experiments planning.

References

[1] J. H. Yoon, Y. Seo, Y. S. Jo, S. Lee, E. Cho, A. Cazenave-Gassiot, Y.-S. Shin, M. H. Moon, H. J. An, M. R. Wenk, and et al., "Brain lipidomics: From functional landscape to clinical significance," *Science Advances*, vol. 8, no. 37, 2022.

[2] Y. Peng, P. Gao, L. Shi, L. Chen, J. Liu, and J. Long, "Central and peripheral metabolic defects contribute to the pathogenesis of alzheimer's disease: Targeting mitochondria for diagnosis and prevention," *Antioxidants amp;amp; Redox Signaling*, vol. 32, no. 16, p. 1188–1236, 2020.

[3] "Python data analysis library," https://pandas.pydata.org/.

[4] "scipy.spatial.ckdtree - scipy v1.11.4 reference guide," https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.cKDTree.html, 2023, accessed: [2023].

[5] "Pycaret regression module documentation," https://pycaret.readthedocs.io/en/stable/api/regression.html#pycaret.regression.create_model, 2023, accessed: [2023].

[6] "Pycaret regression module - tune_model function documentation," https://pycaret.readthedocs.io/en/stable/api/regression.html#pycaret.regression.tune_model, 2023, accessed: [2023].

[7] "Keras api," https://keras.io/.

[8] "Tensorflow library," https://www.tensorflow.org/?hl=it.

[9] "Glorotuniform initialization, tensorflow documentation," https://www.tensorflow.org/api_docs/python/tf/keras/initializers/GlorotUniform.

[10] "Jaccard index," https://en.wikipedia.org/wiki/Jaccard_index.

[11] "scipy.spatial.distance.pdist - scipy v1.11.4 reference guide," https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.pdist.html, 2023, accessed: [2023].

[12] "Allen brain atlas: Mouse brain," https://mouse.brain-map.org/static/atlas, 2003.

[13] Y. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 18 932–18 943. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/file/9d86d83f925f2149e9edb0ac3b49229c-Paper.pdf

[14] V. McElfresh, Khandagale, F. Prasad C, G. Hegde, Ramakrishnan, and White, "When do neural nets outperform boosted treeson tabular data?" 2023. [Online]. Available: https://arxiv.org/pdf/2305.02997.pdf

VI. Code and data availability

Data and code used to perform this analysis are placed on the GitHub repository

VII. Appendix

For more information, it is possible to look at the Appendix.