

¹ RESEARCH

² **Modeling the cell-type specific murine connectome**

³ **Samson Koelle^{1,2}, Jennifer Whitesell¹, Karla Hirokawa¹, Hongkui Zeng¹, Marina Meila², Julie Harris¹, Stefan Mihalas¹**

⁴ ¹Allen Institute for Brain Science, Seattle, WA, USA

⁵ ²Department of Statistics, University of Washington, Seattle, WA, USA

⁶ **Keywords:** [a series of capitalized words, separated with commas]

ABSTRACT

⁷ The Allen Brain Connectivity Atlas consists of thousands of labelling experiments targeting
⁸ interrogating diverse structures and classes of projecting neurons. This paper describes the
⁹ conversion of these experiments into class-specific connectivity matrices representing the connection
¹⁰ between source and target structures. We introduce and validate a novel statistical model for creation
¹¹ of connectivity matrices that combines spatial and categorical smoothing to share information
¹² between similar neuron classes. We then investigate patterns in the resultant connectivities, and show
¹³ that our connectivities display expected cell-type specific structures.

AUTHOR SUMMARY

INTRODUCTION

¹⁴ The animal nervous system enables an extraordinary range of natural behaviors, and has inspired
¹⁵ much of modern artificial intelligence. Neural connectivities - axon-dendrite connections from one
¹⁶ region to another - form the architecture underlying this capability. These connectivities vary by
¹⁷ neuron type, as well as axonic source and dendritic target structure. Thus, characterization of the

18 relationship between neuron type and source and target structure is an important step to
19 understanding the nervous system.

20 Viral tracing experiments - in which a viral vector expressing GFP is transduced into neural cells
21 through stereotaxic injection - are a useful tool for understanding these connections on the mesoscale
22 (???). The GFP protein moves from axon to dendrite through the process of anterograde projection, so
23 neurons 'downstream' of the injection site will also fluoresce. Two-photon tomography imaging can
24 then determine the location and strength of the fluorescent signals in two-dimensional slices. These
25 locations can then be mapped back into three-dimensional space, and the signal is partitioned into
26 the transduced source and merely transfected target regions.

27 The conversion of such experiment-specific signals into an overall estimate of the connectivity
28 strength of two regions is accomplished by a statistical model. ? and ? describe two such methods.
29 Intuitively, both of these models provide some improvement over simply averaging the projection
30 signals of injections in a given region. is another. These models are evaluated based off of their ability
31 to predict held-out experiments in leave-one-out cross validation. A model that performs well in such
32 validation experiments is then assumed to generate the most accurate connectivity.

33 Both ? and ? develop models for mostly wild-type mice using a standardized vector over all
34 experiments. However, recent work (?) has extended these datasets to include viral tracing
35 experiments inducing cell-type specific fluorescence. This is accomplished by injecting vectors with
36 Cre-recombinase triggered GFP promoters into transgenic mice with cell-type specific
37 Cre-recombinase expression Thus, the this paper extends the methodology of ? and ? to deal with the
38 diverse set of cre-lines described in ?.

39 This extension relies on a to our knowledge novel estimator that takes into account both the spatial
40 position of the labelled source, as well as the categorical cre-label. This model outperforms the model
41 of ?, even for wild-type experiments.

42 The resulting cell-type specific connectivity matrices form a multi-way *neural connection tensor* of
43 information about neural structure. We do not attempt an exhaustive analysis of this data, but do
44 demonstrate several basic phenomena. First, we verify several cell-type specific patterns found
45 elsewhere in the literature. Second, we discover cell-type specific signals in the neural connection

⁴⁶ tensor. Finally, we decompose the overall (wild-type) connectivity matrix into factors representing
⁴⁷ archetypal connective patterns.

METHODS

48 We create cell-type specific connectivity matrices using a model trained on murine viral-tracing
49 experiments. This model predicts projection patterns of different neuron classes at different locations
50 within the brain that are more accurate than simple averages over nearby experiments in
51 cross-validation. This section describes the data used to generate the model, the model itself, the
52 evaluation of the model, and the use of the model in creation of the connectivity matrices. We then
53 give exploratory analyses of the resulting connectivities that illustrate their key features. Additional
54 information on our methods is given in Supplemental Section .

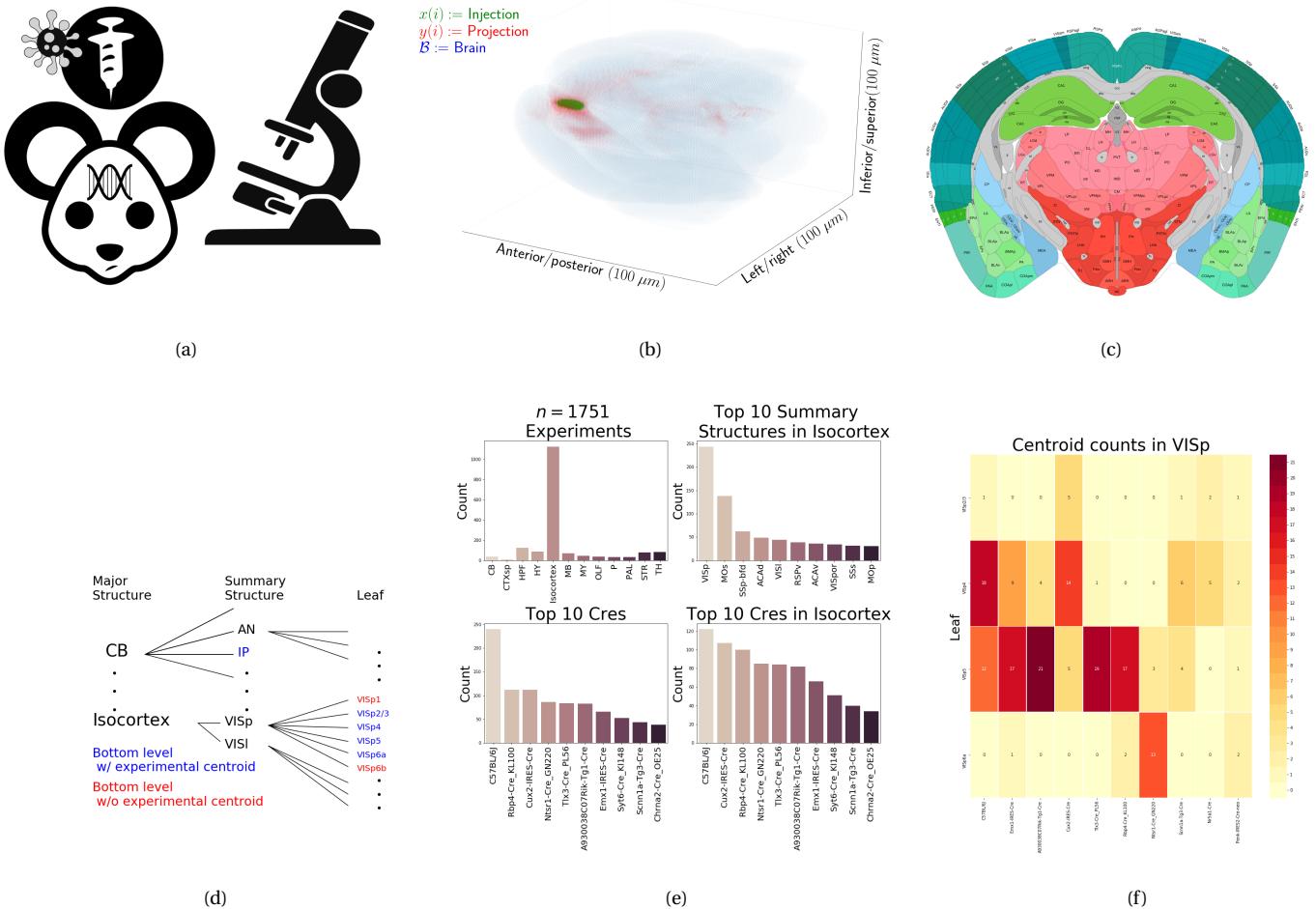


Figure 1: Experimental setting. a) Within the brain (blue), injection (green) and projection (red) areas are determined via histological analysis and alignment to the Allen Common Coordinate Framework (CCF). b) An example of the segmentation of projection and injection for a single experiment. c) Example of structural segmentation within a horizontal plane. d) Explanation of nested structural ontology highlighting lowest-level and data-relevant structures. e) Abundances of crelines and structural injections. f) Cooccurrence of layer-specific centroids and creline within V1Sp

55 **Mice**

56 (SK's comment:**Experiments involving mice were approved by the Institutional Animal Care and**
57 **Use Committees of the Allen Institute for Brain Science in accordance with NIH guidelines.**)

58 **Data**

59 Our dataset \mathcal{D} consists of $n = 1751$ experiments from the Allen Mouse Connectivity Atlas. Figures 1a
60 summarizes the multistage experimental process used to generate this data. In each experiment, a
61 GFP-labelled transgene cassette with a potentially cre-specific promoter is injected into a particular
62 location in a cre-driver mouse. This causes fluorescence that depends on the localization of
63 cre-recombinase expression within the mouse. While frequently this localization corresponds to a
64 specific cell-type, it can also correspond to a combination of cell-types. For example, in wild-type
65 mice injected with non-cre specific promoters, fluorescence is observed in all areas projected to from
66 the injection site, regardless of cell-type. Thus, we use the term *neuron class* to describe the neurons
67 targeted by a specific combination of transgene and mouse-line. This is the notion of cell-type
68 specificity that we model.

69 After injection, the resultant fluorescent signal is imaged, and aligned into the Allen Common
70 Coordinate Framework (CCF) v3, a three-dimensional idealized model of the brain that is consistent
71 between animals. This imaging and alignment procedure (described in detail in (??)) records
72 fluorescent intensity discretized at the $100 \mu\text{m}$ voxel level. The image is histologically segmented
73 into *injection* and *projection* areas corresponding to areas of transduction and
74 transduction/transfection, respectively. An example for a single experiment is given in Figure 1b.

75 Our goal is the estimation of structural connectivity from one structure to another. Thus, a visual
76 depiction of this structural regionalization for a slice of the brain is given in Figure 1c. For different
77 areas of the brain, the Allen Atlas contains different depths of discretization. We denote these levels as
78 Major Structures, Summary Structures, and Leafs. As indicated in Figure 1d, the dataset used to
79 generate the connectivity model reported in this paper contains certain combinations of structure
80 and neuron class (S, V) frequently, and others not at all. A summary of the most frequently assayed
81 neuron classes and structures is given in Figures 1e and 1f. Since users of the connectivity matrices
82 may be interested in particular combinations, or interested in the amount of data used to generate a

⁸³ particular connectivity estimate, we exhaustively present this information about all experiments in
⁸⁴ Appendix .

⁸⁵ We can preprocess this publically available data in several ways. A detailed mathematical
⁸⁶ description of these is given Appendix .

⁸⁷ **Connectivity**

At an essential level, cell-class specific neural connectivity is a function $f : \mathcal{V} \times \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}^+$ giving the directed connection of a particular neuron class from a one position in the brain to another. However, what we actually create is, given a set of source regions $\mathcal{S} = \{S\}$, target regions $\mathcal{T} = \{T\}$, and neuron classes V ,

$$\text{connectivity strength } \mathcal{C} \in \mathcal{V} \times \mathcal{S} \times \mathcal{T} \times \mathbb{R}_{\geq 0} \text{ with } \mathcal{C}(V, S, T) = \sum_{s \in S} \sum_{t \in T} f(v, s, t),$$

$$\text{normalized connectivity strength } \mathcal{C}^S \in \mathcal{V} \times \mathcal{S} \times \mathcal{T} \times \mathbb{R}_{\geq 0} \text{ with } \mathcal{C}^S(V, S, T) = \frac{1}{|S|} \mathcal{C}(V, S, T),$$

$$\text{normalized projection density } \mathcal{C}^D \in \mathcal{V} \times \mathcal{S} \times \mathcal{T} \times \mathbb{R}_{\geq 0} \text{ with } \mathcal{C}^D(V, S, T) = \frac{1}{|S||T|} \mathcal{C}(V, S, T).$$

- ⁸⁸ These represent the strength of the connection from source to target regions for each class. Since the
⁸⁹ normalized strength and density are computable from the strength via a fixed normalization, our
⁹⁰ main statistical goal is to estimate $\mathcal{C}(V, S, T)$. We call this estimator $\hat{\mathcal{C}}$.

Construction of such an estimator raises the questions of what data to use for estimating which connectivity, how to featurize the dataset, what statistical estimator to use, and how to reconstruct the connectivity using the chosen estimator. Mathematically, we represent these considerations as

$$\hat{\mathcal{C}}(V, S, T) = e^*(\hat{e}(e_*(\mathcal{J}(\mathcal{D}))). \quad (1)$$

- ⁹¹ This makes explicit the data featurization e_* , statistical estimator \hat{e} , and any potential subsequent
⁹² transformation e^* such as averaging over the source region, as well as the fact that different data \mathcal{D}
⁹³ may be used to estimate different connectivities. Table 1 reviews estimators used for this data-type.
⁹⁴ Additional information is given in Appendix .

Model	e^*	\hat{e}	e_*	Training Data
(?)	$\hat{e}(S)$	NNLS(X,Y)	$X = r(x(I)), Y = r(y(I))$	$I = I_M$
(?)	$\sum_{s \in S} \hat{e}(s)$	NW(X,Y)	$X = c(x(I)), Y = r(y(I))$	$I = I_M$
Cre-NW	$\sum_{s \in S} \hat{e}(s)$	NW(X,Y)	$X = c(x(I)), Y = n(r(y(I)))$	$I = I_S \cap I_V$
Expected Loss (EL)	$\sum_{s \in S} \hat{e}(s)$	EL _S (X, Y, V)	$X = c(x(I)), Y = n(r(y(I))), V = v(I)$	$I = I_S$

Table 1: Estimation of \mathcal{C} using connectivity data. The regionalization, estimation, and featurization steps are denoted by e^* , \hat{e} , and e_* , respectively. The training data used to fit the model is given by I . We generically denote the set of experiments used to train a particular model as I , and experiments from particular major brain divisions, summary structures, and leafs as I_M , I_U , and I_L , respectively.

95 Our contributions - the Cre-NW and Expected Loss (EL) models - have several differences from the
 96 previous methods. In contrast to the ? non-negative least squares and ? Nadaraya-Watson estimators
 97 that take into account s and t , but not v , our new estimators specifically account for neural class. The
 98 Cre-NW estimator only uses experiments from a particular neural class to predict connectivity for that
 99 class, while the EL estimator shares information between classes.

100 ***Model evaluation***

We select optimum functions from within and between our estimator classes using empirical risk minimization. Equation 1 includes a deterministic step e^* included without input by the data. The performance of $\hat{\mathcal{C}}$ is therefore determined by performance of the model $\hat{f}(v, s, t) = \hat{e}(e_*(\mathcal{J}(\mathcal{D})))$. We can then evaluate $\hat{f}(v, s, t)$ using leave-one-out cross validation, in which the accuracy of the model is assessed by its ability to predict experiments excluded from the training data. In order to compare between methods, we necessarily restrict to the smallest set of evaluation experiments suggested by any of our models. Since the number of parameters fit is quite low relative to the size of the evaluation set, we do not make use of a formal validation-test split. We use l_2 -loss and weighted l_2 -loss to evaluate these predictions.

$$\begin{aligned} l_2\text{-loss } l(\hat{f}) &= \frac{1}{|I_M|} \sum_{i \in I_M} \|r(y(i)) - \hat{f}(c(i))\|_2^2. \\ \text{weighted } l_2\text{-loss } l(\hat{f}) &= \frac{1}{|\{S, V\}|} \sum_{s, v \in \{S, V\}} \frac{1}{|I_{s, v}|} \sum_{i \in I_{s, v}} l(r(y(i)), \hat{f}(\mathbb{D} \setminus i)). \end{aligned}$$

101 As a final modeling step, we establish a lower limit of detection. This is covered in Appendix

102 **Connectivity analyses**

103 We show neuronal processes underlying our estimated connectome using a variety of types of
104 undersupervised learning. Clustering projection patterns by class and source structure. This shows
105 that cell-class has a dominating effect on projection in certain regions. We flatten the connectivity
106 tensor $\mathcal{C} \in \mathbb{R}^{c \times s \times t}$ to $\mathcal{C}_b \in \mathbb{R}^{cs \times t}$ and cluster the cs sources by their t -dimensional projections.

107 Second, we extend the characterization of ? on structural differences in short-range projections.
108 These are primarily assumed to be due to diffusion, and the diffusion-rate helps to characterize the
109 basic structural anatomy. Third, since the overall wild-type connectome results from the combination
110 of underlying cell-classes, we apply non-negative matrix factorization (NMF) to decompose the
111 observed long-range connectivity into *connectivity archetypes* that linearly combine to reproduce the
112 observed connectivity. These methods identify structures with both known and plausible biological
113 meaning, and simplistically exemplify useful posthoc analyses for data of this type. Technical details
114 of these approaches are given in Appendix.

RESULTS

¹¹⁵ Our results include evaluation of model fit, the cre-specific connectivity matrices themselves, and
¹¹⁶ retrospective analyses of these matrices for patterns related to cre-type and source and target regions.

¹¹⁷ ***Model evaluation***

¹¹⁸ Table contains the sizes of these evaluation sets in each major structure. This information may be
¹¹⁹ cross-referenced visually with the figures in Our two-stage model generally performs better than the
¹²⁰ cre-line specific NW estimator.

Structure	Estimator Smoothing Target # Eval exps	EL	NW	Average	NW	NW-wt
		SS	Cre-SS	Cre-SS	SS	M
		SS	SS	SS	SS	SS
CB	10	0.044	0.081	0.081	0.058	0.439
CTXsp	2	0.497	0.497	0.497	0.497	0.000
HPF	79	0.122	0.140	0.143	0.155	0.471
HY	41	0.241	0.266	0.269	0.244	1.019
Isocortex	838	0.173	0.195	0.202	0.234	0.404
MB	23	0.151	0.151	0.166	0.139	0.759
MY	7	0.186	0.233	0.233	0.184	0.452
OLF	17	0.069	0.095	0.100	0.073	0.110
P	8	0.236	0.239	0.239	0.264	0.984
PAL	11	0.190	0.198	0.198	0.260	1.401
STR	45	0.084	0.088	0.089	0.097	0.265
TH	29	0.351	0.678	0.678	0.365	1.088

Table 2: Weighted losses with summary structure targets.

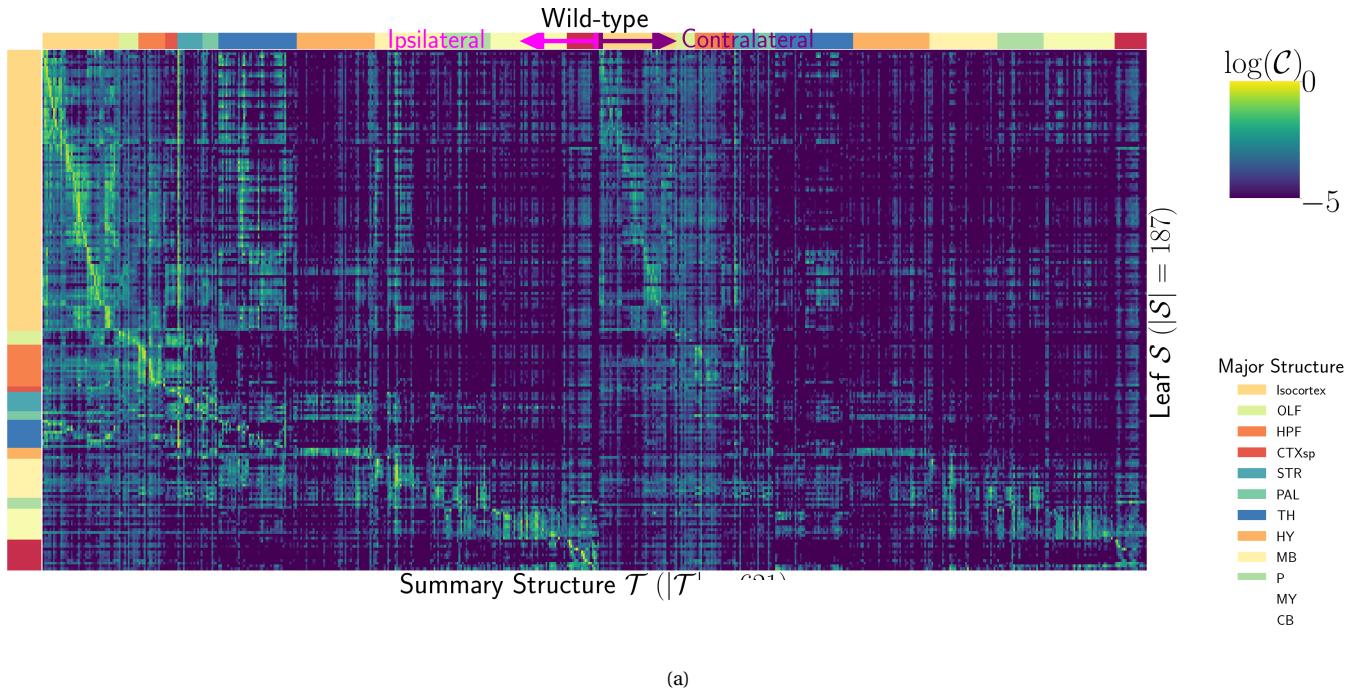
¹²¹ Additional information on model evaluation, including class and structure specific performance, is
¹²² given in Appendix

123 **Connectivities**

124 Our main result is the estimation of matrices $\hat{\mathcal{C}}_v$ representing connections of source structures to
125 target structures for particular cre-lines v . We exhibit several characteristics of interest, and confirm
126 the detection of several well-established connectivities within our tensor. Many additional interesting
127 biological processes are visible within this matrix - more than we can report in this paper - and it is
128 our expectation that these will be identified by users of our results. The connectivity tensor and code
129 to reproduce it are available at

130 https://github.com/AllenInstitute/mouse_connectivity_models/tree/2020.

131 *Overall connectivity* The connectivity matrix for wild-type connectivities from leaf sources to
132 summary structure targets is illustrated in Figure ???. The clear intraareal connectivities mirror
133 previous estimates in ? and ? and descriptive depictions of individual experiments in ?. Compared
134 with ?, our more discretized source smoothing and greater number of experiments leads to a
135 significantly more discretized connectivity matrix. This is generally expected - for example, different
136 cortical layers have more substantially different connectivities.



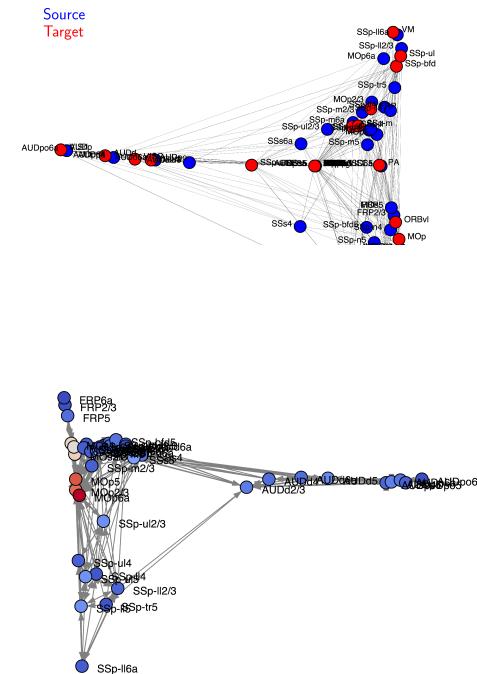
(a)

	# Ipsilateral Leaf Targets	Top Entropy	Bottom Sparsity	Bottom Entropy	Top Sparsity
Isocortex	51	CP	BAC	BAC	ENT1
OLF	11	TMv	III	III	NaN
HPF	15	IG	EPv	PA	NaN
CTXsp	7	TT	FC	APr	TT
STR	14	RPA	ISN	PVR	TU
PAL	9	PG	ACVII	GR	MG
TH	44	NOD	DN	SSp-ll	SCm
HY	44	CLA	SH	LSc	DG
MB	39	NDB	SubG	SGN	SUB
P	26	MT	Acs5	SOC	NDB
MY	43	RT	NaN	OV	EPd
CB	18	ECT	AOB	MOB	GU

(b)

	# Ipsilateral Leaf Targets	Top Entropy	Bottom Sparsity	Bottom Entropy	Top Sparsity
Isocortex	51	CP	BAC	BAC	ENT1
OLF	11	TMv	III	III	NaN
HPF	15	IG	EPv	PA	NaN
CTXsp	7	TT	FC	APr	TT
STR	14	RPA	ISN	PYR	TU
PAL	9	PG	ACVII	GR	MG
TH	44	NOD	DN	SSp-ll	SCm
HY	44	CLA	SH	LSc	DG
MB	39	NDB	SubG	SGN	SUB
P	26	MT	Acs5	SOC	NDB
MY	43	RT	NaN	OV	EPd
CB	18	ECT	AOB	MOB	GU

(d)

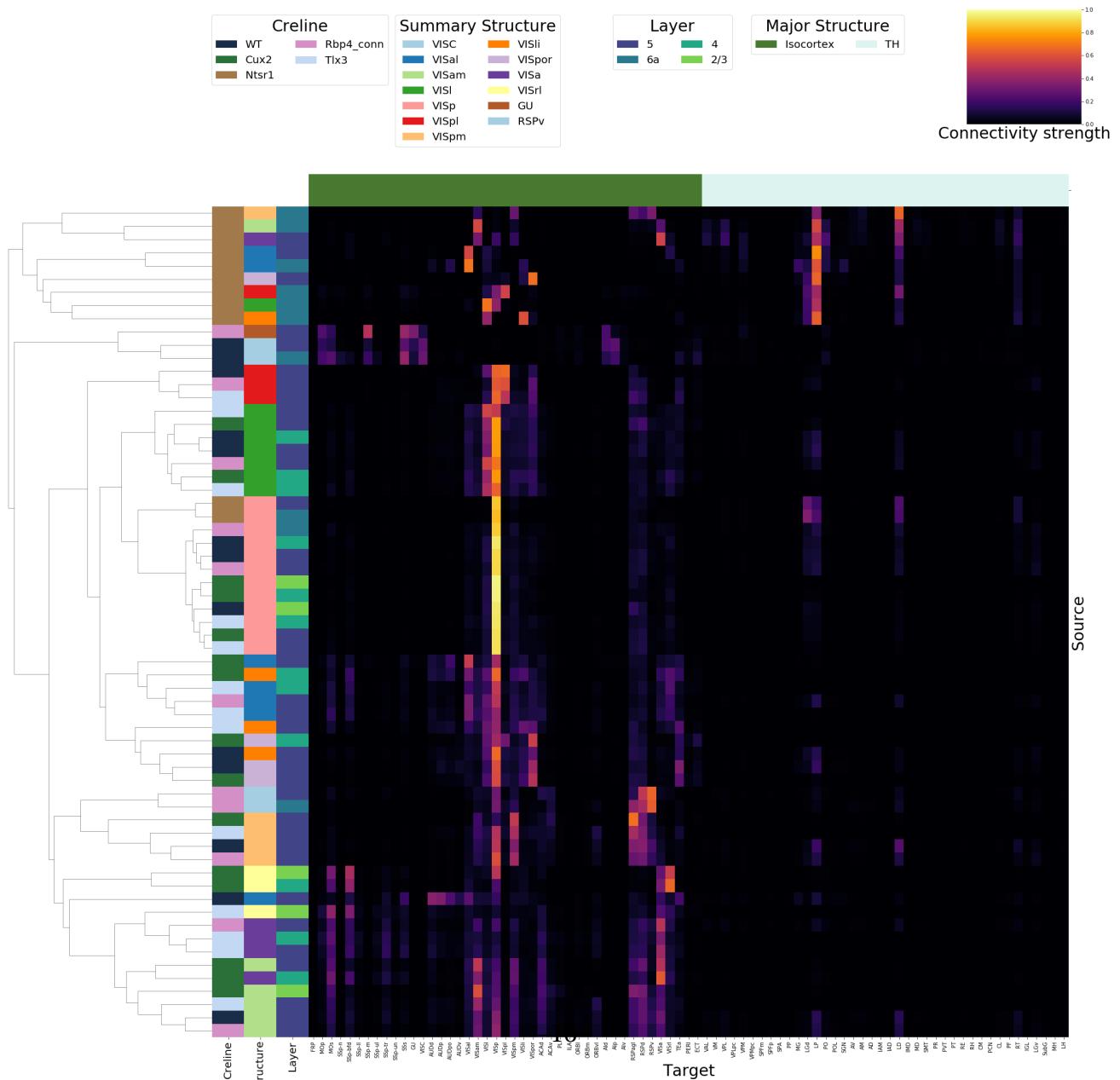


(e)

¹³⁷ *Class-specific connectivities* The cell-type specific connectivities that we provide also conform to
¹³⁸ well-known behaviors. Examples from the visual processing and motor control regions of the cortex
¹³⁹ are given in Figure ?? for both wild type and several cre-lines. Rbp4-Cre and Ntsr1-Cre target layers 5
¹⁴⁰ and 6, respectively. As in ?, layer 5 projects to anterior basolateral amygdala (BLA) and capsular
¹⁴¹ central amygdala (CEA), while layer 6 does not.



(f)



142 **Connectivity Analyses**

143 The connectivity matrix represents a collection of relatively few biological processes. For example,
144 certain cell-types and layers have a characteristic connectivity pattern, and structures tend to connect
145 most strongly to the most proximal areas. We elucidate these patterns through two types of analyses.
146 First, we demonstrate cell-type specific connectivity patterns by hierarchical clustering of
147 connectivities from multiple cre-lines, and showing that cre-line is a key factor driving the observed
148 behavior. Then, we perform a different unsupervised analysis - non-negative matrix factorization - of
149 distal wild-type connectivities, to estimate underlying overall connectivity patterns.

150 Figure ?? shows a collection of connectivity strengths generated using cre-specific models for
151 wild-type, Cux2, Ntsr1, Rbp4, and Tlx3 cre-lines from visual signal processing leafs in the cortex to
152 cortical and thalamic nucleii. Heirarchical clustering is applied to sort the different source/cre
153 combinations by the similarity of their connectivities to summary-structure targets. This analysis
154 shows that Ntsr1 cre-lines tend to target thalamic nucleii, in particular LP and LD ?. However, with
155 this exception, for the other plotted cre-lines, connectivity tends to cluster by source structure. That
156 the tendency for structures to connect to themselves is quite strong emphasizes the special nature of
157 the Ntsr1-Thalamic connection in this analysis.

158 The overall wild-type connectivity strength matrix also displays an underlying modellable
159 structure. As discussed in ?, one of the most basic processes underlying the observed connectivity is
160 the tendency of each source region to predominantly project to proximal regions. The heatmap in ??a)
161 shows intraregion distances clearly contains an overall pattern reminiscent of the connectivity matrix
162 in ?. This relationship is plotted in ?? b), showing that there exists substantial variability that would
163 be impossible to model with low-error in a univariate model, even using the diffusion model
164 suggested in ?. These connections are biologically meaningful, but also unsurprising, and their
165 relative strength biases learned latent coordinate representations away from long-range structures.
166 For this reason, we establish a $1500\mu m$ 'distal' threshold within which to exclude connections for our
167 analysis. We then apply non-negative matrix factorization (NMF) to decompose the remaining
168 censored matrix into a relatively small number of distinct projection signals, and apply an

169 unsupervised cross-validation method to select the optimum number of signals ([SK's](#)

170 [comment:Percent error... show reconstruction? log scale?](#)).

DISCUSSION

¹⁷¹ Flattening \mathcal{C} prior to unsupervised analysis is not necessarily recommended, but provides an easy
¹⁷² solution for this problem.

¹⁷³ With respect to the model, a Wasserstein-based measure of injection similarity per structure would
¹⁷⁴ combine both the physical simplicity of the centroid model while also incorporating structural
¹⁷⁵ knowledge.

¹⁷⁶ The Nadaraya-Watson weighting procedure introduced here is, to our knowledge, novel. In
¹⁷⁷ particular, our method of utilizing the expected loss to weight points differs from the minimization
¹⁷⁸ task of fitting data to weighted sums of neighbors (?). We make a key assumption: that the additional
¹⁷⁹ statistical accuracy of including more samples makes up for the fact that their expected accuracy is
¹⁸⁰ lower. Note that this assumption can be easily violated, if, for example, the data is distributed on a
¹⁸¹ circle without error, and only nearest neighbors are most predictive.

¹⁸² Model averaging based off of cross-validation has been implemented in ?, but we note that our
¹⁸³ approach makes use of a non-parametric estimator, rather than an optimization method for selecting
¹⁸⁴ the weights. ([SK's comment:CITE METHOD THAT SELECTS WEIGHTS IN KERNEL \(has catchy](#)
¹⁸⁵ [name](#))

ACKNOWLEDGMENTS

¹⁸⁶ The Funder and award ID information you input at submission will be introduced by the publisher
¹⁸⁷ under a Funding Information head during production. Please use this space for any additional
¹⁸⁸ acknowledgements and verbiage required by your funders.

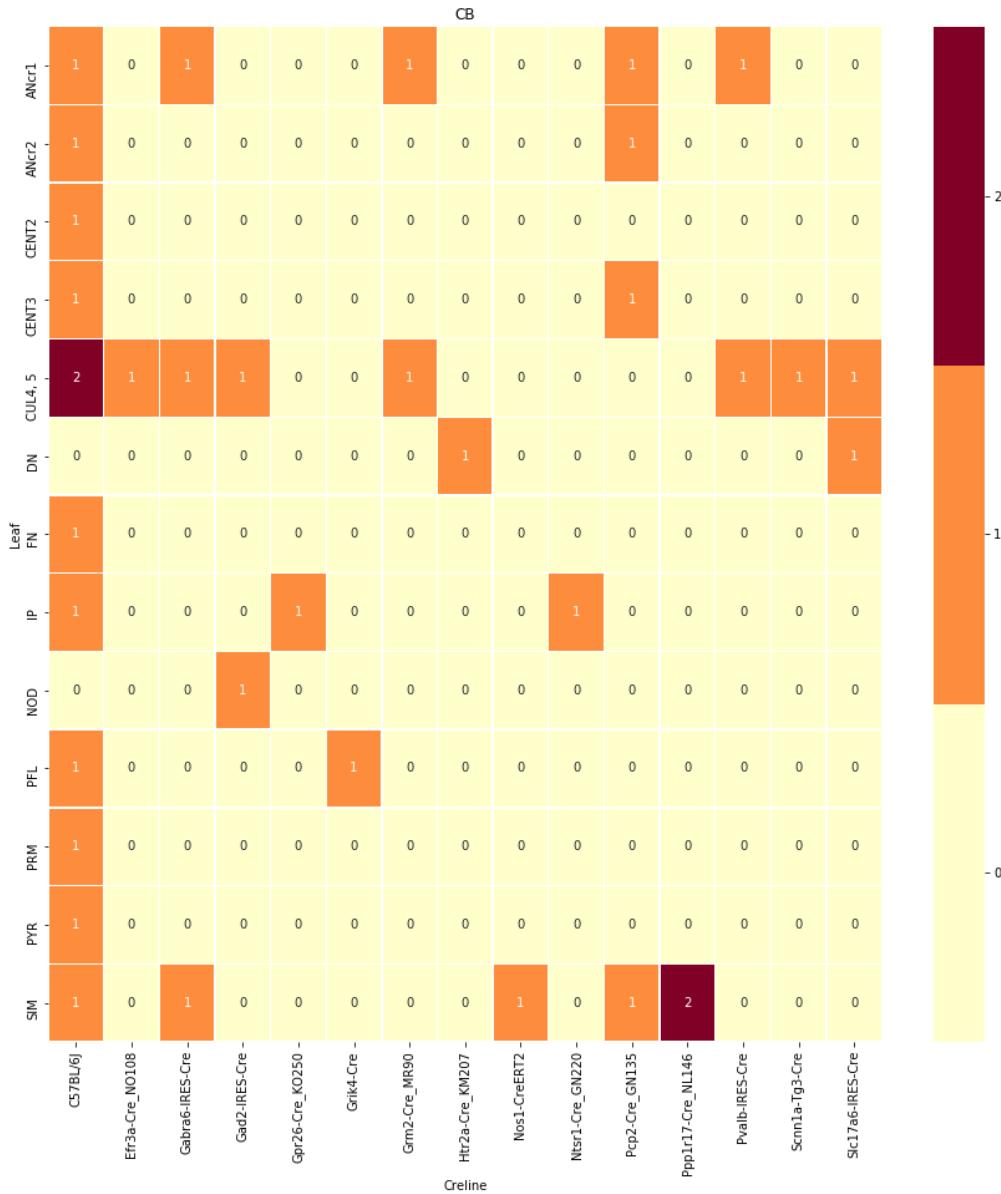
SUPPORTING INFORMATION

SUPPLEMENTAL INFORMATION

¹⁸⁹ **Data**

¹⁹⁰ This section describes the set of leaf and cre-line experimental combinations.

centroid densityoct12.png



centroid densityoct12.png

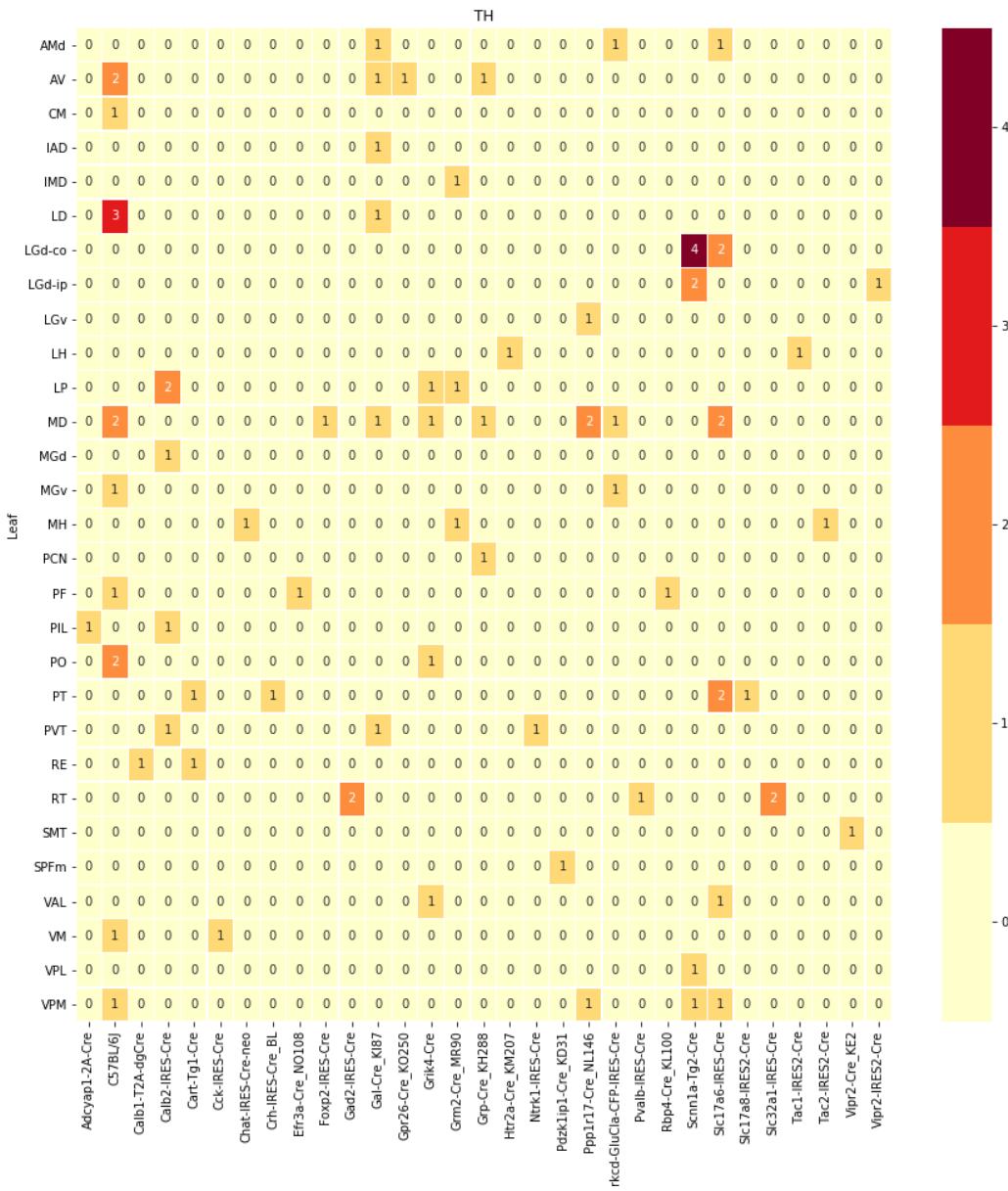


Figure 3: Caption

centroid densityoct12.png

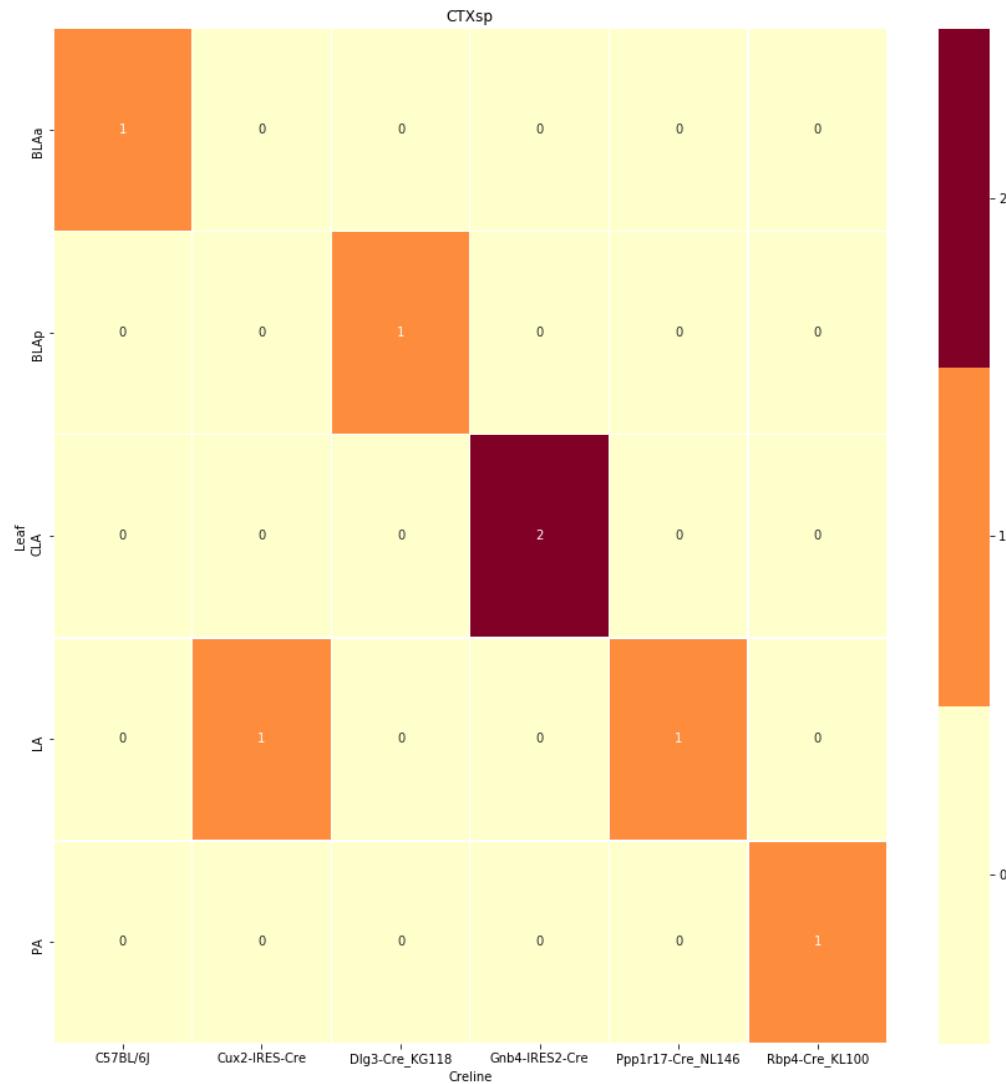
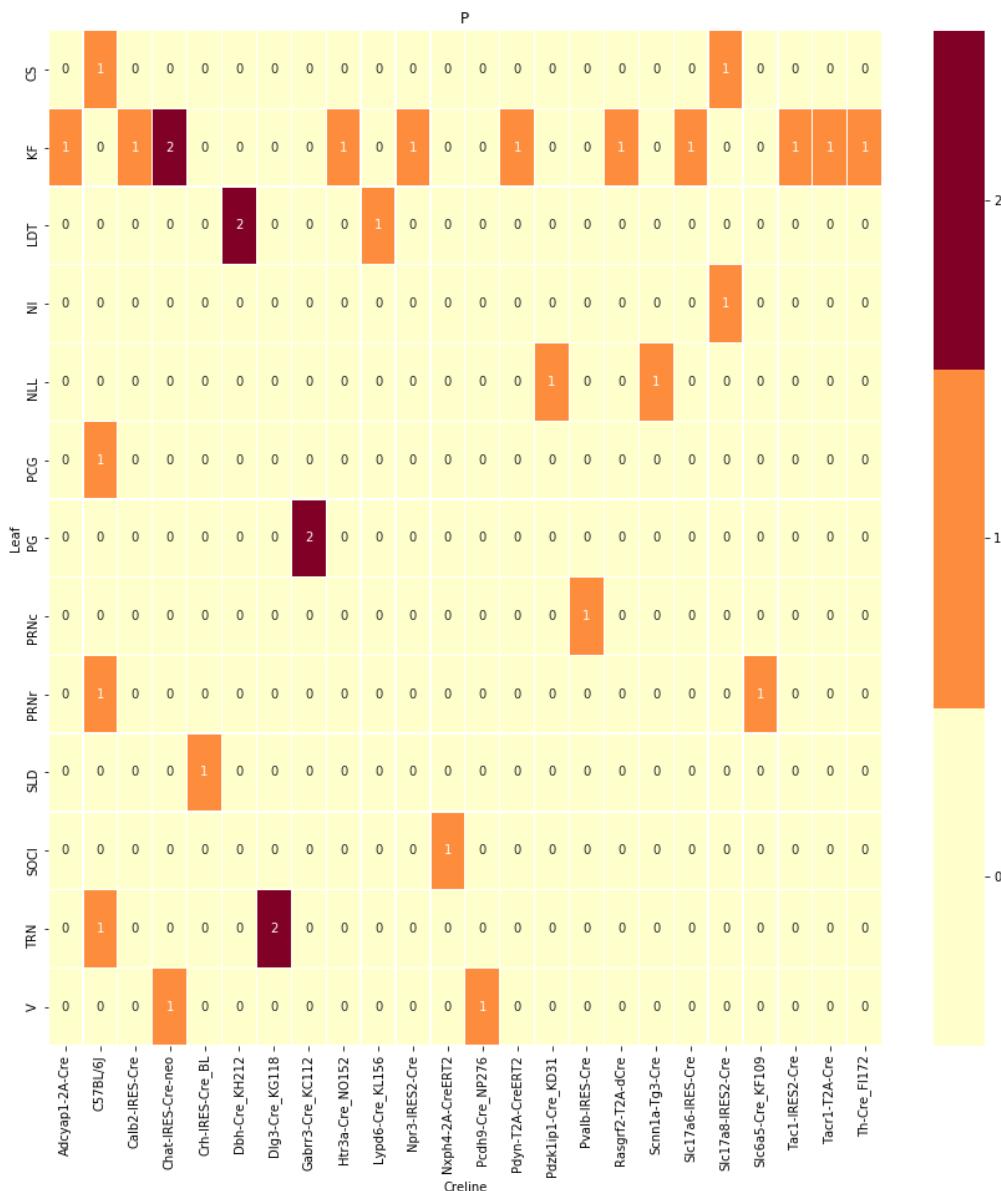
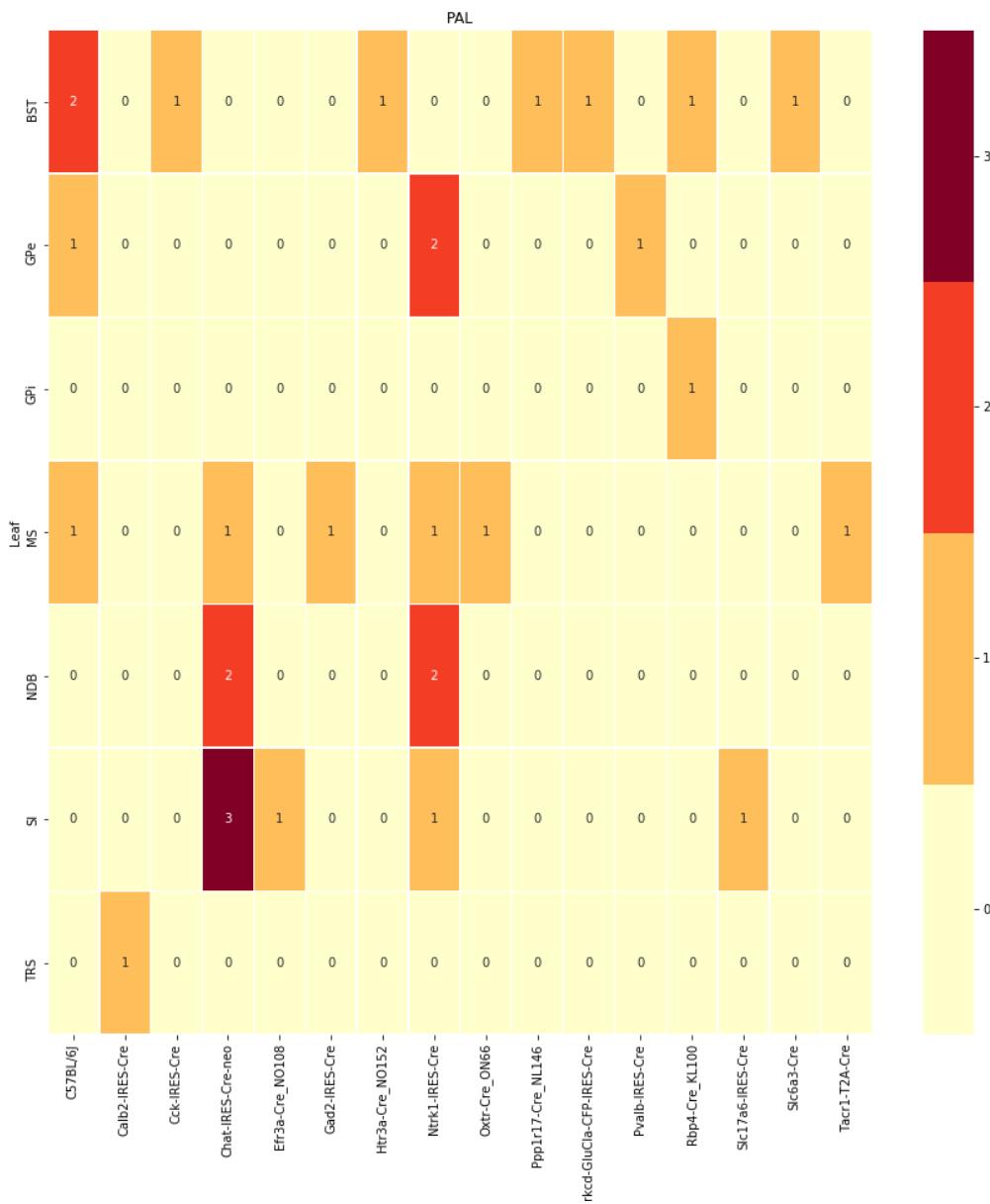


Figure 4: Caption

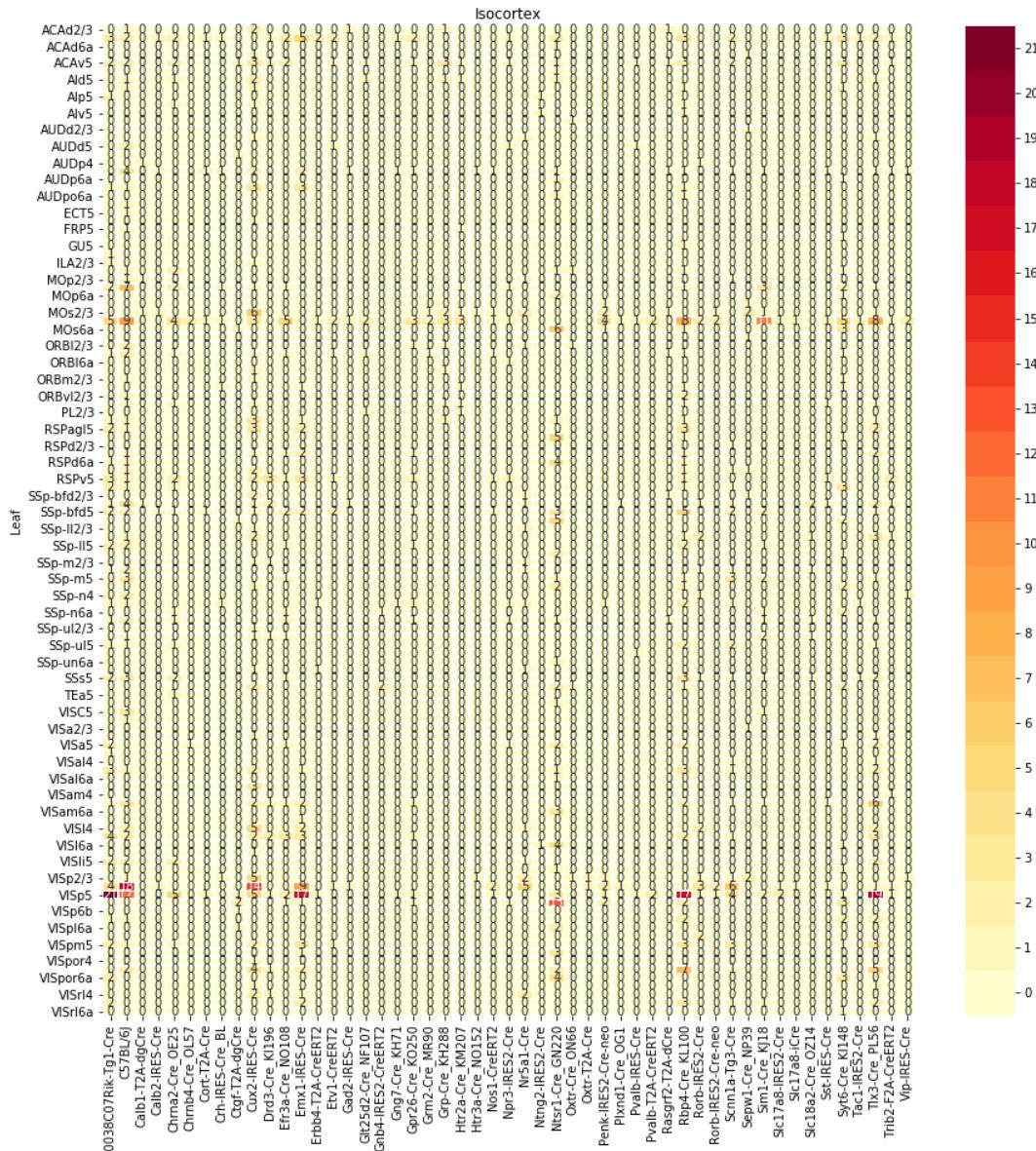
centroid densityoct12.png



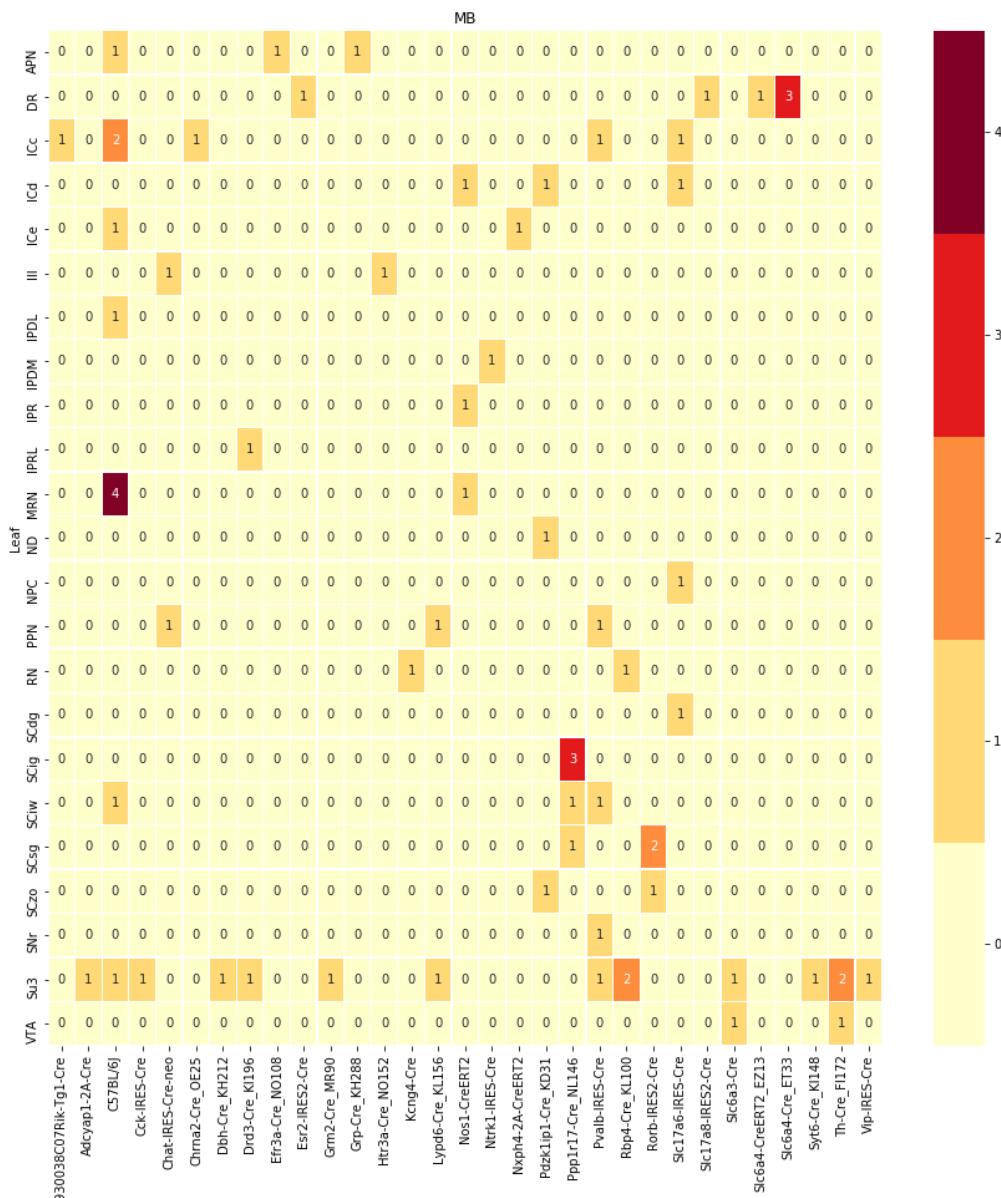
centroid densityoct12.png



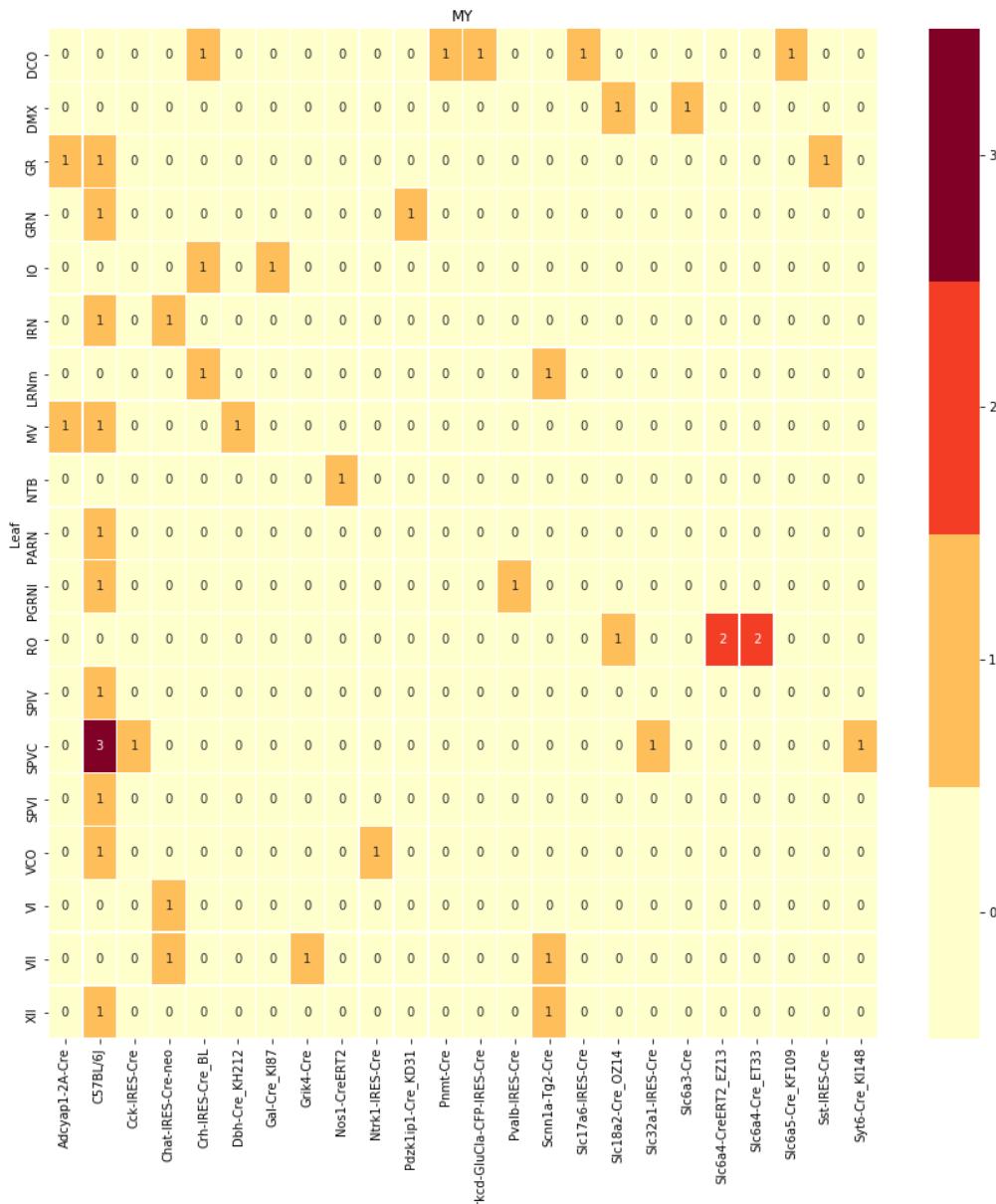
centroid densityoct12.png



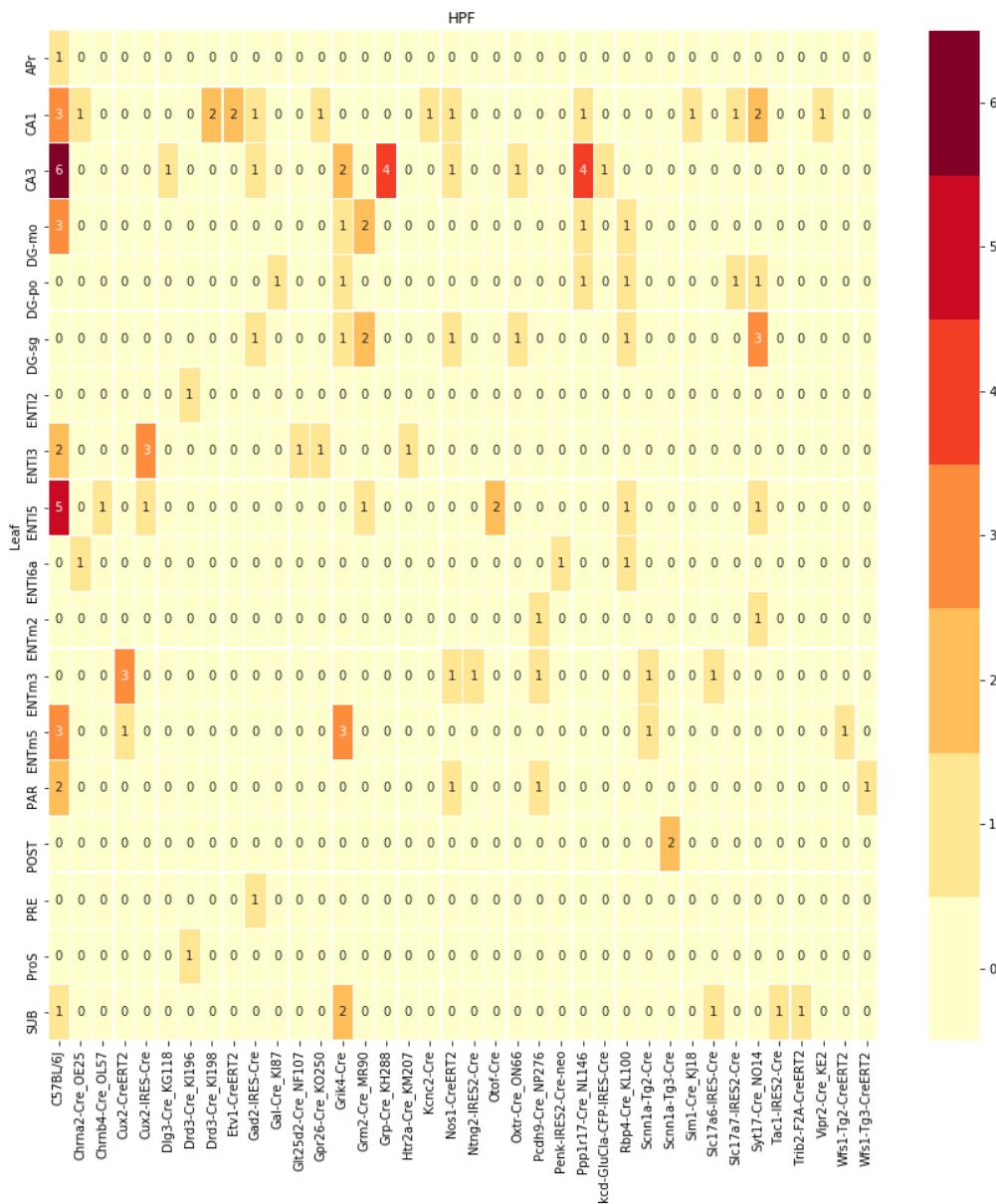
centroid densityoct12.png



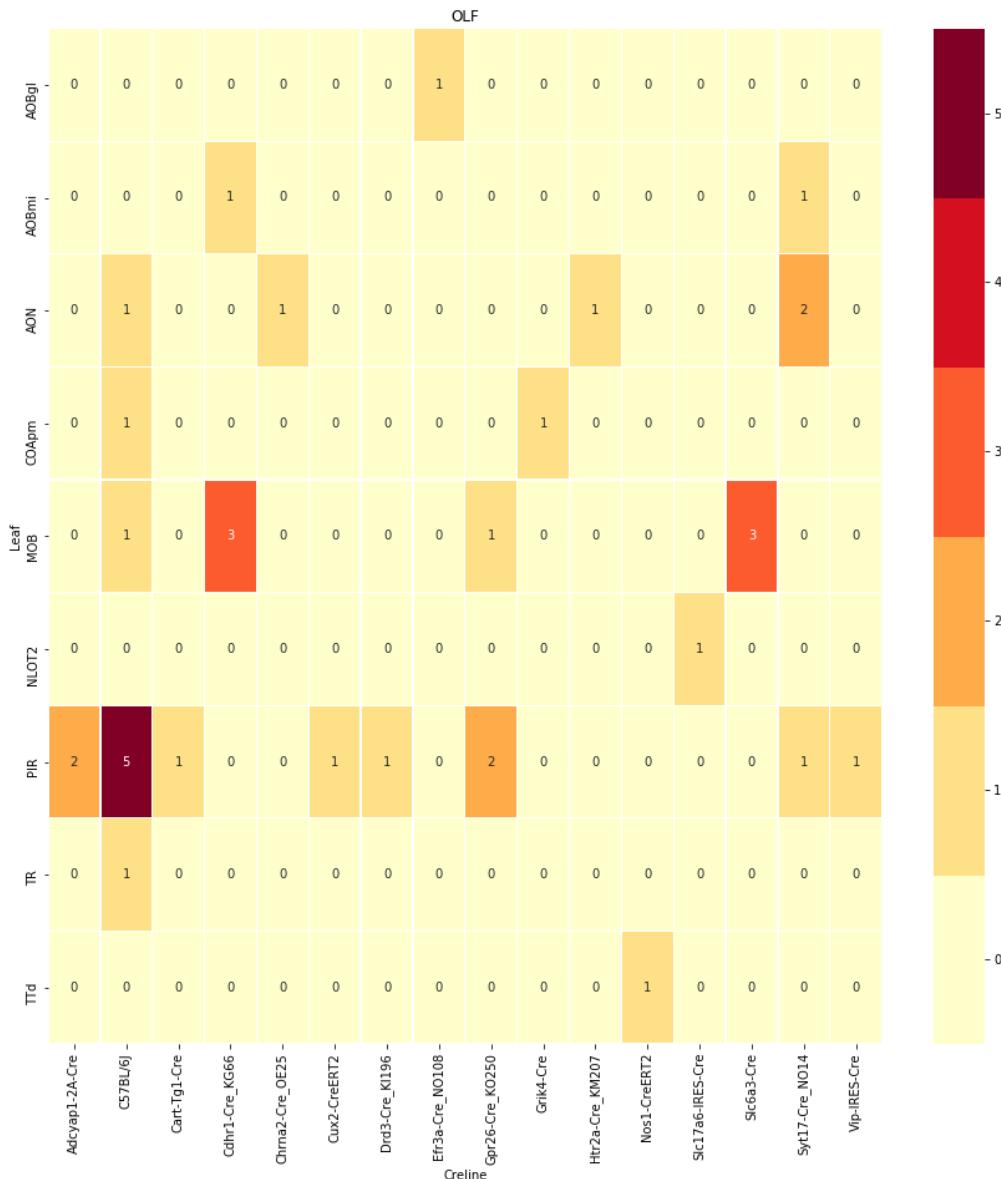
centroid densityoct12.png



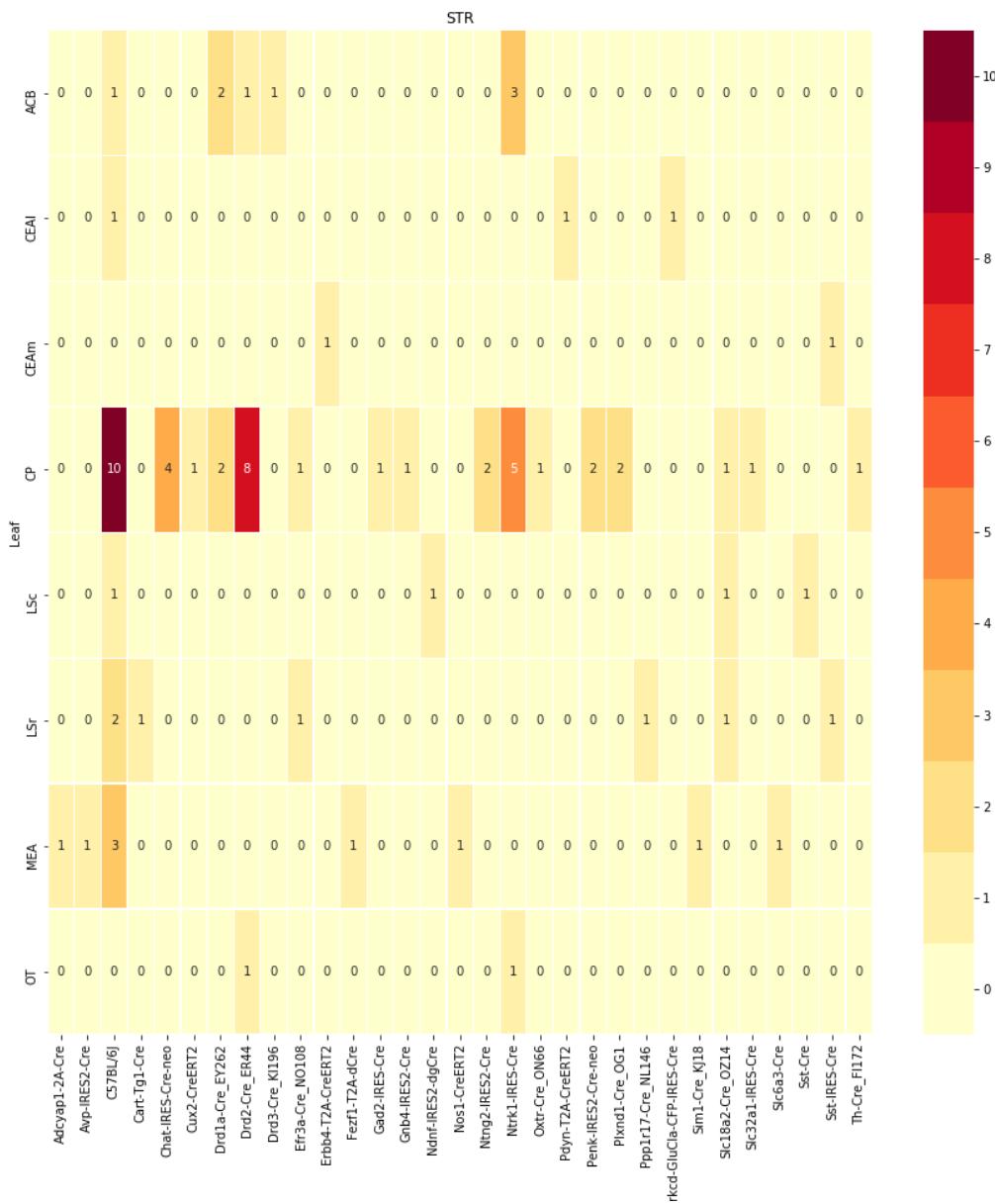
centroid densityoct12.png



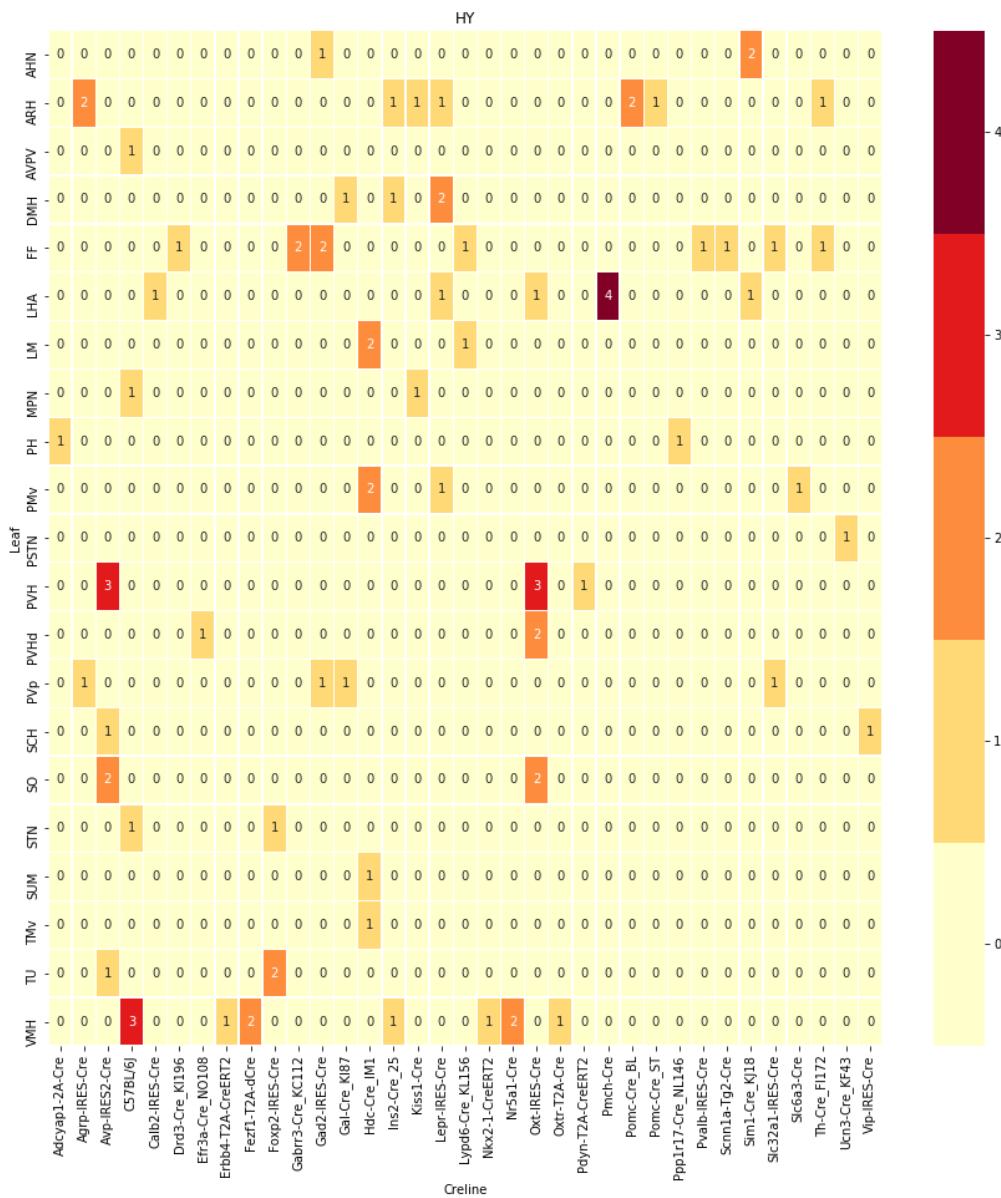
centroid densityoct12.png



centroid densityoct12.png



centroid densityoct12.png



191 **Structure information**

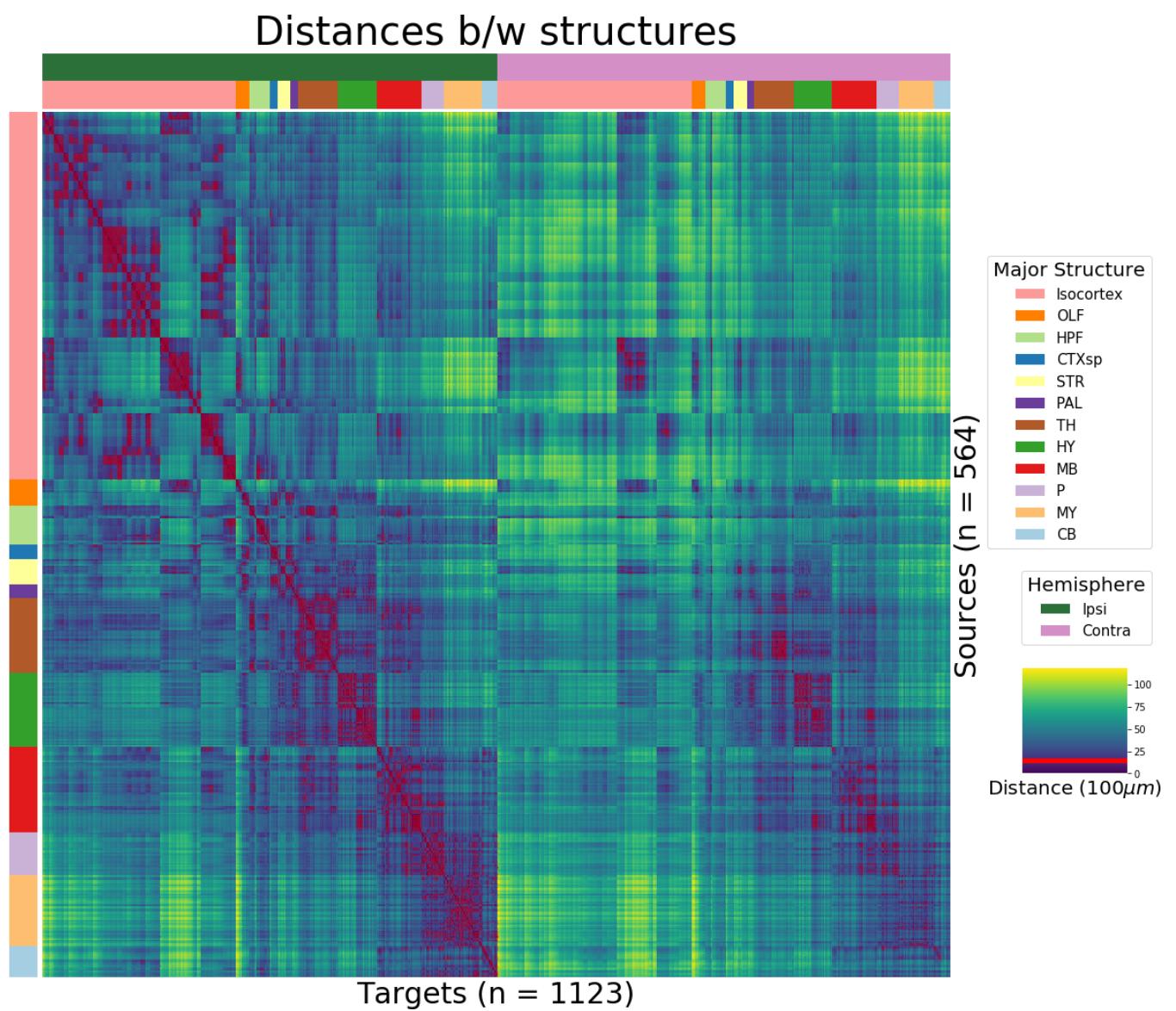


Figure 5: Distance between structures. Short-range connections are masked in red

192 ***Model evaluation***

	Total	Cre-Summary	Cre-Summary, Leaf	Cre-Leaf
0	36	10	9	4
1	7	2	2	2
2	122	79	79	62
3	85	41	41	41
4	1128	838	829	732
5	68	23	23	18
6	46	7	7	7
7	35	17	17	17
8	33	8	8	8
9	30	11	11	11
10	78	45	45	45
11	83	29	29	29

Table 3: Number of experiments available to evaluate models in leave-one-out cross validation. Models that rely on a finer granularity of modeling have less data available to validate with.

SUPPLEMENTAL METHODS

¹⁹³ This section consists of additional information on preprocessing of the neural connectivity data,
¹⁹⁴ estimation of connectivity, and matrix factorization.

¹⁹⁵ **Data preprocessing**

¹⁹⁶ Several data preprocessing steps take place prior to evaluations of the connectivity matrices.
¹⁹⁷ Injections and projections were downloaded using the Allen SDK. These were originally annotated
¹⁹⁸ manually. The injection and projection vectors are Hadamard multiplied by a data quality matrix. We
¹⁹⁹ also have a map $A : \mathbb{R} \rightarrow \mathbb{R}^{|S|}$ where S is the number of structures that takes the average value for voxels
²⁰⁰ in that structure

PREPROCESS 1 Input Injection $x(i)$, Projection $y(i)$, Injection centroid $c(i) \in \mathbb{R}^3$, injection fraction

$F(i)$, data mask $M(i)$

Data-quality censor $y_M(i) = M(i) \odot y(i)$

Average structures $y_S(i) = Ay_M(i)$

Normalize $\tilde{y}(i) = \frac{y_S(i)}{\|y_S(i)\|}$

Output $\tilde{y}(i)$

²⁰¹ The data-quality censor is established by ([SK's comment:fill](#)) The injection fraction accounts for the
²⁰² relatively coarse graining of the voxel grid compared with the histological analysis used to establish
²⁰³ the injection region. In particular, certain voxels are only partially contained within the injection
²⁰⁴ region.

²⁰⁵ One basic but significant methodological change from [?](#) is the normalization of projection vectors.

²⁰⁶ The loss function in [?](#) is

$$\frac{\|y - \hat{y}\|}{\|y\| \|\hat{y}\|}$$

²⁰⁷ **Estimators**

²⁰⁸ Our estimators span a range of training and featurization methods. One commonality is that they
²⁰⁹ model a connectivity vector $f(\mathcal{D}, v, s) \in \mathbb{R}^T$, and so we may write

$$f(v, s, t) = f(v, t)[t].$$

²¹⁰ Thus, for the remainder of this section, we will discuss only $f(s, v)$.

²¹¹ *Centroid-based Nadaraya-Watson* In the Nadaraya-Watson approach of ?, the injection is considered
²¹² only through its centroid, while the projection is considered regionalized. That is,

$$f_*(\mathcal{D}_i) = \{c(x_i), r(y_i)\}.$$

²¹³ Since the injection is considered only by its centroid, this model only generates predictions for
²¹⁴ particular locations c , and the prediction for a structure s is given by integrating over locations within
²¹⁵ the structure

$$f^*(\hat{f}(f_*(\mathcal{D}))(v, s) = \sum_{c \in s} \hat{f}(f_*(\mathcal{D}))(v, c),$$

²¹⁶ This \hat{f} is the Nadaraya-Watson estimator

$$\hat{f}_{NW}(c(x_{1:n}), r(y_{1:n}))(c, v) := \sum_{i \in I} \frac{\omega_{c(x_i)c}}{\sum_{i \in I} \omega_{c(x_i),c}} r(y_i)$$

²¹⁷ where $\omega_{c(x_i)c} = \exp(-\gamma d(c, c(x_i))^2)$ and d is the Euclidean distance between centroid $c(x_i)$ and voxel c .

²¹⁸ Several facets of the estimator are visible here. A smaller γ corresponds to a greater amount of
²¹⁹ smoothing, and index set $I \subseteq \{1 : n\}$ indicates which experiments to use to generate the prediction.
²²⁰ Fitting γ via empirical risk minimization therefore bridges between 1-nearest neighbor prediction and
²²¹ averaging of all experiments in I . In ?, I consisted of experiments sharing the same brain division.
²²² Restricting of index set to only include experiments with the same neuron class gives the
²²³ class-specific model.

224 *The expected-loss estimator* The response induced by each of the cre-lines is effected by both the
 225 injection location and the targeted cell types. Cre-lines that target similar cell types are therefore
 226 expected to induce similar projections, and including similar cre-lines in our estimator thus increases
 227 the effective sample size. In order to leverage this fact in a data-driven way, we introduce an estimator
 228 that assigns a predictive weight to each training point that depends both on its centroid-distance and
 229 cre-line. This weight is determined by the expected prediction error of each of the two feature types,
 230 as determined by cross-validation. These weights are then utilized in a Nadaraya-Watson estimator in
 231 a final prediction step.

232 We formalize cre-line behavior as the average regionalized projection of a cre-line in a given leaf.
 233 This vectorization of categorical information is known as target encoding. We define a cre-distance in
 234 a leaf to be the distance between the target-encoded projections of two cre-lines. The relative
 235 predictive accuracy of cre-distance and centroid distance is determined by fitting a surface of
 236 projection distance as a function of cre-distance and centroid distance.

237 In mathematical terms, our full feature set consists of the centroid coordinates and the
 238 target-encoded means of the combinations of virus type and injection-centroid structure. That is,

$$f_*(\mathcal{D}_i) = \{c(x_i), \bar{r}(y_{I_v}), r(y_i)\}.$$

239 f^* is defined as in (2). The expected loss estimator is then

$$\hat{f}_{EL}(c, c(x_i), v, r(y_{I_v})) = \sum_{i \in I} \frac{\nu(c(x_i), c, v_i, v)}{\sum_{i \in I} \nu(c(x_i), c, v_i, v)} r(y_i)$$

240 where

$$\nu_i = \exp(-\gamma g(d(c, c(x_i))^2, d(\bar{r}(v), \bar{r}(v_i))^2))$$

241 Note that g must be a concave, non-decreasing function of its arguments with $g(0, 0) = 0$, then g
 242 defines a metric on the product of the metric spaces defined by experiment centroid and
 243 target-encoded cre-line, and \hat{f}_{EL} is a Nadaraya-Watson estimator. A derivation of this fact is given in
 244 Appendix , and we therefore use shape-constrained B-splines to estimate g .

245 This contrasts with the model is ?, where $\hat{f}(c)$ does not depend on v , and ?, where connectivity was
 246 directly estimated by \hat{f} a function of S without an integral. Estimating $\hat{f}(v, c)$ shares the advantage of

²⁴⁷ fine-scale spatial resolution with $\mathbf{?}$, but in addition enables us to model a particular virus-type v , and,
²⁴⁸ as we will see, make use of experimental data in our estimator.

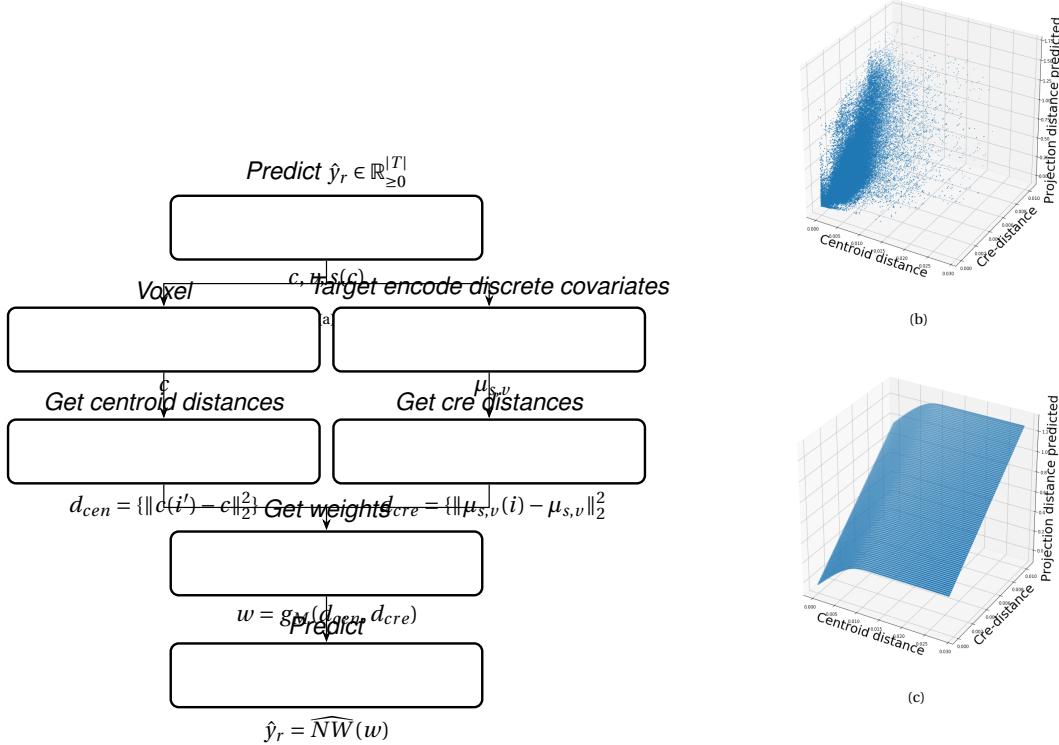


Figure 6: The Expected-Loss estimator

249 JUSTIFICATION OF SHAPE CONSTRAINT The shape-constrained expected-loss estimator introduced
 250 in this paper is, to our knowledge, novel. It should be considered an alternative method to the classic
 251 weighted kernel method. While we do not attempt a detailed theoretical study of this estimator, we do
 252 establish the need for the shape constraint in our spline estimator. Though this fact is probably well
 253 known, we prove a (slightly stronger) version here for completeness.

254 Given a collection of metric spaces X_1, \dots, X_n with metrics d_1, \dots, d_n (e.g. $d_{centroid}, d_{cre}$), and a
 255 function $f : (X_1 \times X_1) \times \dots \times (X_n \times X_n) = g(d_1(X_1 \times X_1), \dots, d_n(X_n \times X_n))$, then f is a metric iff g is
 256 concave, non-decreasing and $g(d) = 0 \iff d = 0$.

257 We first show g satisfying the above properties implies that f is a metric.

- 258 ▪ The first property of a metric is that $f(x, x') = 0 \iff x = x'$. The left implication:
 259 $x = x' \implies f(x_1, x'_1, \dots, x_n, x'_n) = g(0, \dots, 0)$, since d are metrics. Then, since $g(0) = 0$, we have that
 260 $f(x, x') = 0$. The right implication: $f(x, x') = 0 \implies d = 0 \implies x = x'$ since d are metrics.
 261 ▪ The second property of a metric is that $f(x, x') = f(x', x)$. This follows immediately from the
 262 symmetry of the d_i , i.e. $f(x, x') = f(x_1, x'_1, \dots, x_n, x'_n) = g(d_1(x_1, x'_1), \dots, d_n(x_n, x'_n)) =$
 263 $g(d_1(x'_1, x_1), \dots, d_n(x'_n, x_n)) = f(x'_1, x_1, \dots, x'_n, x_n) = f(x', x)$.
 264 ▪ The third property of a metric is the triangle inequality: $f(x, x') \leq f(x, x^*) + f(x^*, x')$. To show this
 265 is satisfied for such a g , we first note that $f(x, x') = g(d(x, x')) \leq g(d(x, x^*) + d(x^*, x'))$ since g is
 266 non-decreasing and by the triangle inequality of d . Then, since g is concave,
 267 $g(d(x, x^*) + d(x^*, x')) \leq g(d(x, x^*)) + g(d(x^*, x')) = f(x, x^*) + f(x^*, x')$.

268 We then show that f being a metric implies that g satisfies the above properties.

- 269 ▪ The first property is that $g(d) = 0 \iff d = 0$. We first show the right implication: $g(d) = 0$, and
 270 $g(d) = f(x, x')$, so $x = x'$ (since f is a metric), so $d = 0$. We then show the left implication:
 271 $d = 0 \implies x = x'$, since d is a metric, so $f(x, x') = 0$, since f is a metric, and thus $g(d) = 0$.
 272 ▪ The second property is that g is non-decreasing. We proceed by contradiction. Suppose g is
 273 decreasing in argument d_1 in some region $[l, u]$ with $0 < l < u$. Then
 274 $g(d_1(0, l), 0) \geq g(d_1(0, 0), 0) + g(d_1(0, u), 0) = g(d_1(0, u), 0)$, which violates the triangle inequality on
 275 f . Thus, decreasing g means that f is not a metric, so f a metric implies non-decreasing g .
 276 ▪ The final property is that g is concave. We proceed by contradiction. Suppose g is strictly convex.
 277 Then there exist vectors d, d' such that $g(d + d') < g(d) + g(d')$. Assume that d and d' only are
 278 non-zero in the first position, and $d = d(0, x), d' = d(0, x')$. Then, $f(0, x) + f(0, x') < f(0, x + x')$,
 279 which violates the triangle inequality on f . Therefore, g must be concave.

280 *Establishing a lower detection limit* The lower detection limit of our approach is a complicated
 281 consequence of our experimental and analytical protocols. For example, the Nadaraya-Watson
 282 estimator is likely to generate many small false positive connections, since the projection of even a
 283 single experiment within the source region to a target will cause a non-zero connectivity in the
 284 Nadaraya-Watson weighted average. On the other hand, the complexities of the experimental

285 protocol itself and the image analysis and alignment can also cause spurious signals. Therefore, it is of
 286 interest to establish a lower-detection threshold below which we have very little power-to-predict, and
 287 set estimated connectivities below this threshold to zero. This should make our estimated
 288 connectivities more accurate, especially in the biologically-important sense of sparsity.

289 We establish this limit with respect to the sum of Type 1 and Type 2 errors

$$\tau = \sum 1_{f(s,t,c)=0} 1_{\hat{f}(s,t,c)>0} + 1_{f(s,t,c)>0} 1_{\hat{f}(s,t,c)=0}.$$

290 ***Decomposing the connectivity matrix***

291 We utilize non-negative matrix factorization (NMF) to analyze the principal signals in our
 292 connectivity matrix. Here, we review this approach as applied to decomposition of the distal elements
 293 of the estimated connectivity matrix $\hat{\mathcal{C}}$ to identify q connectivity archetypes. Aside from the NMF
 294 program itself, the key elements are selection of the number of archetypes q and stabilization of the
 295 tendency of NMF to give random results over different initialization.

296 *Non-negative matrix factorization* Given a matrix $X \in \mathbb{R}_{\geq 0}^{a \times b}$ and a desired latent space dimension q , the
 297 non-negative matrix factorization is

$$NMF(X, q) = \arg \min_{W \in \mathbb{R}_{\geq 0}^{a \times q}, H \in \mathbb{R}_{\geq 0}^{q \times b}} \| (X - WH) \|_2^2.$$

298 NMF creates a useful decomposition since X is in the positive orthant, and PCA cannot apply.
 299 There is no orthogonality without sparsity.

300 We note the existence of NMF with alternative norms for certain marginal distributions, but leave
 301 utilization of this approach for future work (?). We can also apply a mask $1_M \in \mathbb{R}^{S \times T}$ of ones and zeros
 302 and solve

$$\arg \min_{W \in \mathbb{R}_{\geq 0}, H \in \mathbb{R}_{\geq 0}} \| 1_M \odot ((\hat{\mathcal{C}} - WH)) \|_2^2$$

303 For us, such a mask serves for two purposes. First, it enables computation of the NMF objective while
 304 excluding self and nearby connections. These connections are both strong and linearly independent,
 305 and so would dominate the *NMF* reconstruction error. Long range connections are more biologically

306 interesting or cell-type dependent. Second, it enables cross-validation based selection of the number
 307 of retained components.

308 *Cross-validating NMF* Perhaps surprisingly, cross-validation techniques may also be applied to
 309 unsupervised learning problems. These techniques are somewhat standard, but not entirely
 310 well-known, so we review them here, in particular as they apply to the NMF problem. A NMF model is
 311 first fit on a reduced data set, and an evaluation set is held out. After random masking of the
 312 evaluation set, the loss of the learned model is then evaluated on the basis of successful
 313 reconstruction of the held-out values. This procedure is performed repeatedly, with different held out
 314 regions and random mask at different dimensionalities l , to determine to point past which additional
 315 hidden units provide no reconstructive value.

That is, given a matrix $X \in \mathbb{R}^{S \times T}$ we can decompose $X \sim d(e(X))$ where $e(X)$ is some map that encodes X in a learned representation, and d is the decoding reconstruction map. In our case, d is simply left multiplication by W , and e is the solution of a regularized non-negative least squares optimization problem

$$H := e_W(X) = \arg \min_{\beta} \|X - W\beta\|_2^2.$$

316 The form of this solution particularly motivates our cross-validation estimator.

Recall that in supervised learning, the learned model is $Y \sim f(X)$. Standard cross-validation removes elements of X , fits f , and then uses the f learned from part of the data to predict Y . A good f will have low error on the training data, and also low error on the test data, indicating that it has not overfit. Although there is no assumed dichotomy between X and Y in unsupervised learning, for techniques like autoencoders, the above paradigm still applies, i.e., one can still hold out values of X . We can then estimate

$$\arg \min_{d,e} \hat{E}(l(X, d_{XC}(e_{XC}(X)))) = \sum_{r=1}^R l(X_r, d_{XC_r}(e_{XC_r}(X_r)))$$

over R random samples of rows of X . However, in our setting, since computing $e(X)$ on the test rows amounts to fitting a non-negative least squares w.r.t. W , so the negative effects of an overfit model can simply be optimized away from. Thus, the standard solution is to generate uniformly random masks

$1_{M(p)} \in \mathbb{R}^{S \times T}$ where

$$1_{M(p)}(s, t) \sim \text{Bernoulli}(p).$$

Our cross-validation error is then

$$\epsilon_q = \frac{1}{R} \sum_{r=1}^R (\|1_{M(p)_r^C} \odot X - \hat{d}_q(\hat{e}_q(1_{M(p)_r^C} \odot X))\|_2^2$$

where

$$\hat{d}_q, \hat{e}_q = \widehat{\text{NMF}}(1_{M(p)_r} \odot X, q).$$

The optimum number of components is then

$$\hat{q} = \arg \min_q \epsilon_q.$$

317 *Stabilizing NMF* The NMF program is non-convex, and, empirically, individual replicates will not
 318 converge to the same optima. One solution therefore is to run multiple replicates of the NMF
 319 algorithm, cluster the resulting vectors. This approach raises the questions of how many clusters to
 320 use, and how to deal with stochasticity in the clustering algorithm itself. We address this issue through
 321 the notion of clustering stability (?).

The clustering stability approach is to generate L replicas of k-cluster partitions $\{C_{kl} : l \in 1 \dots L\}$ and then compute the average dissimilarity between clusterings

$$\xi_k = \frac{2}{L(L-1)} \sum_{l=1}^L \sum_{l'=1}^l d(C_{kl}, C_{kl'}).$$

Then, the optimum number of clusters is

$$\hat{k} = \arg \min_k \xi_k.$$

322 A review of this approach is found in ?. Intuitively, archetype vectors that cluster together frequently
 323 over clustering replicates indicate the presence of a stable clustering. For d , we utilize the adjusted
 324 Rand Index - a simple dissimilarity measure between clusterings. Note that we expect to select slightly
 325 more than the q components suggested by cross-validation, since archetype vectors which appear in
 326 one NMF replicate generally should appear in others. We then select the q clusters with the most

327 archetype vectors - the most stable NMF results - and take the median of each cluster to create a
328 sparse representative archetype.

SUPPLEMENTAL EXPERIMENTS

329 *Establishing a lower limit of detection*

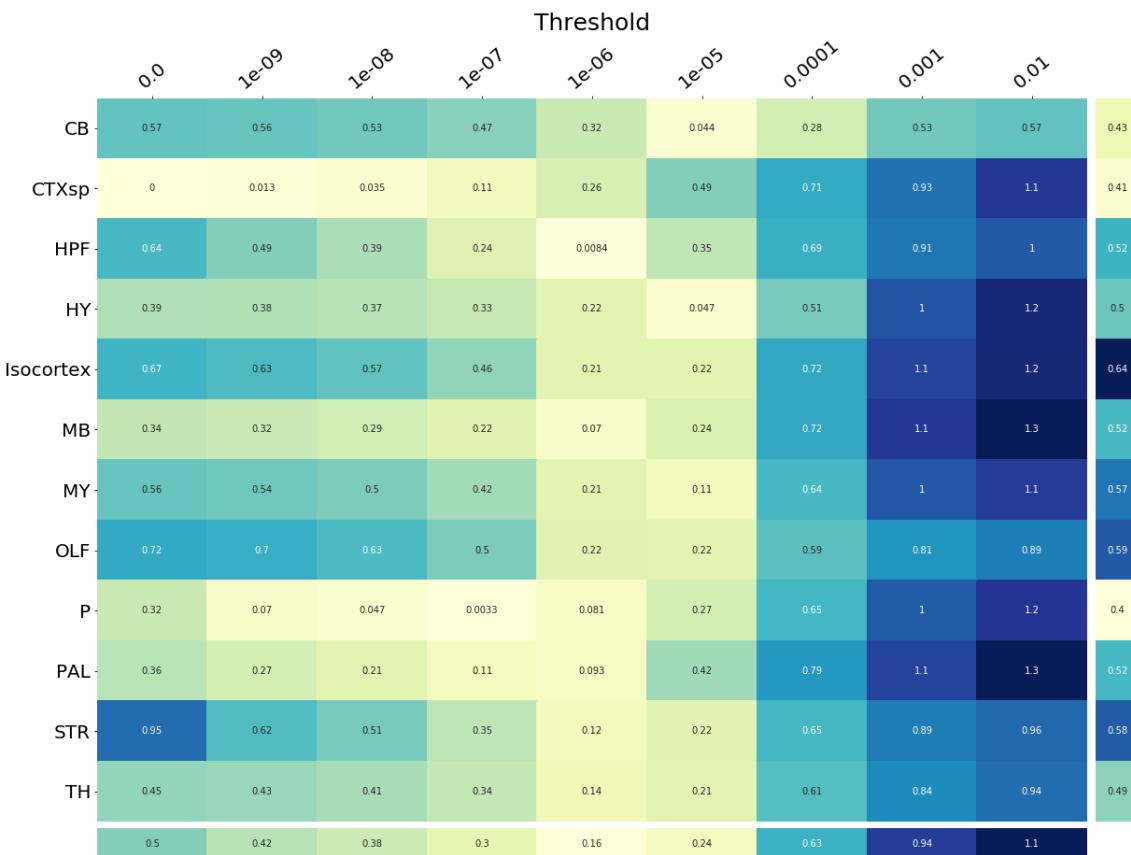


Figure 7: τ at different limits of detection.

330 *Loss subsets*

331 The

332 *Matrix Factorization*

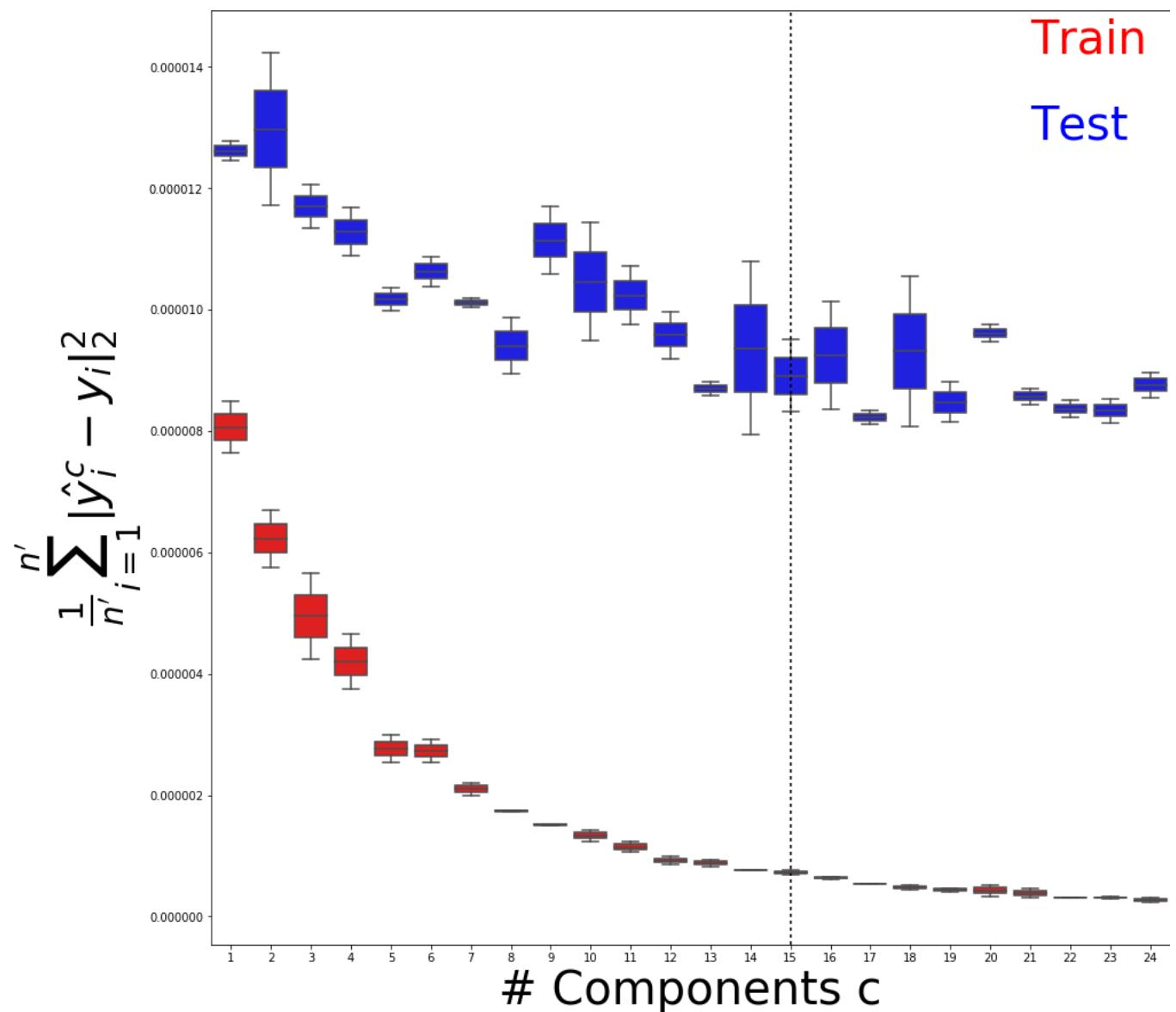


Figure 8: Train and test error across 2 (SK's comment:[increase](#)) replicates using NMF decomposition.

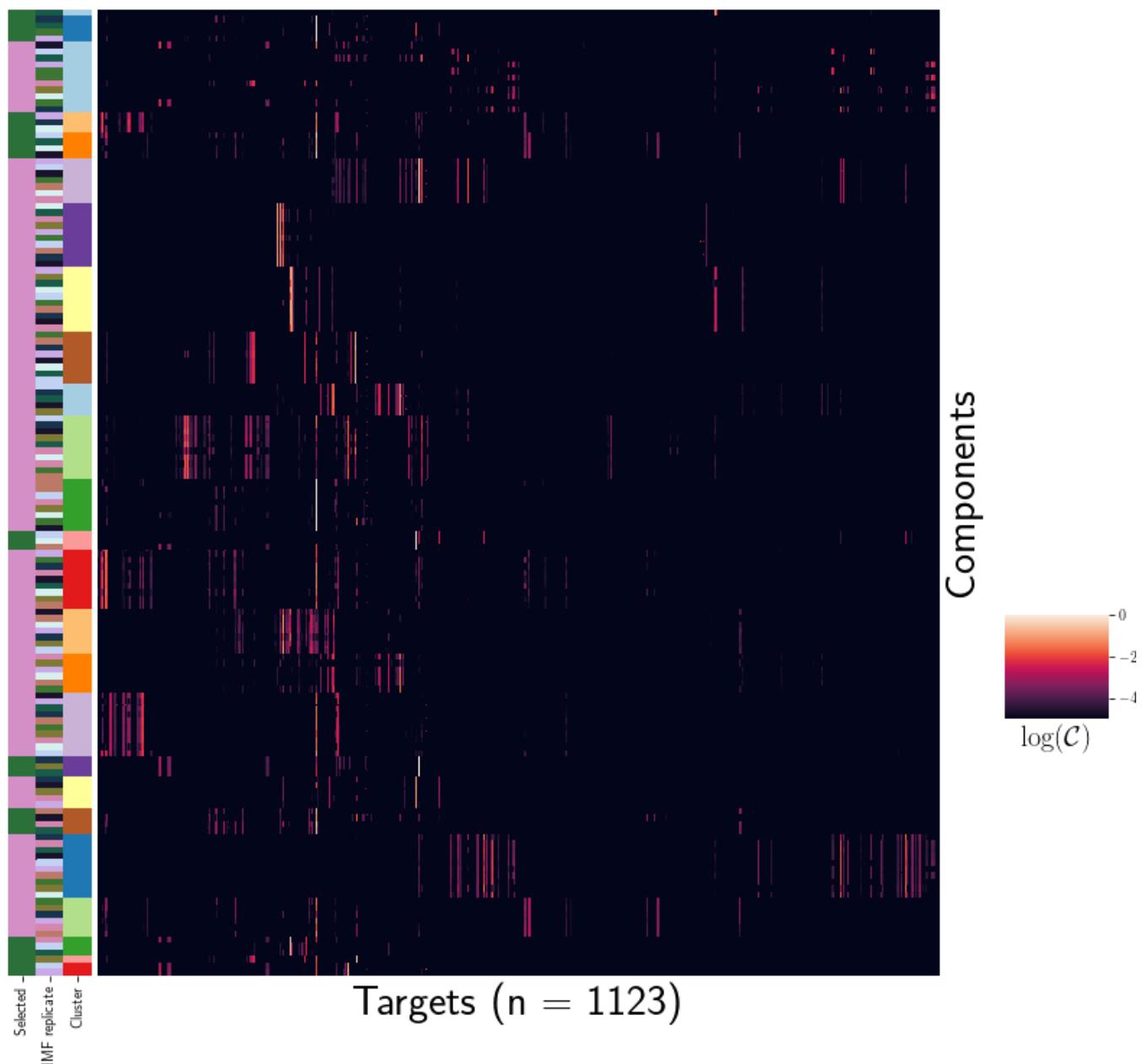


Figure 9: Stability of NMF results across replicates.

COMPETING INTERESTS

- ³³³ This is an optional section. If you declared a conflict of interest when you submitted your manuscript,
³³⁴ please use this space to provide details about this conflict.

TECHNICAL TERMS

³³⁵ All NETN article types require Technical Terms.

³³⁶ Identify approximately 10 key terms that are mentioned in your article and whose usage and
³³⁷ definition may not be familiar across the broad readership of the journal. Provide brief (20-word or
³³⁸ less) definitions for each term, avoiding in these definitions the use of jargon, or highly technical or
³³⁹ specialized language. When the article is typeset, the Technical Terms will appear in the margins at or
³⁴⁰ near their first mention in the text.

³⁴¹ In your manuscript, bold the first occurrence of each **Technical Term** and then provide a list of the
³⁴² terms and their definitions at the end of the manuscript after the references.

³⁴³ **Technical Term** a key term that is mentioned in an NETN article and whose usage and definition
³⁴⁴ may not be familiar across the broad readership of the journal.