Removing Inter-Experimental Variability from Functional Data in Systems Neuroscience

Dominic Gonschorek[*1]
dominic.gonschorek@cin.uni-tuebingen.de

Larissa Höfling[*,1]
larissa.hoefling@uni-tuebingen.de

Klaudia P. Szatko[1]
klaudia.szatko@tuebingen.mpg.de

Katrin Franke[1]
katrin.franke@cin.uni-tuebingen.de

Timm Schubert[1]
timm.schubert@cin.uni-tuebingen.de

Benjamin A. Dunn[2]
benjamin.dunn@ntnu.no

Philipp Berens[1]
philipp.berens@uni-tuebingen.de

David A. Klindt[‡,2]
klindt.david@gmail.com

Thomas Euler[‡,1]
thomas.euler@cin.uni-tuebingen.de

Abstract

Integrating data from multiple experiments is common practice in systems neuroscience but it requires *inter-experimental variability* to be negligible compared to the biological signal of interest. This requirement is rarely fulfilled; systematic changes between experiments can drastically affect the outcome of complex analysis pipelines. Modern machine learning approaches designed to adapt models across multiple data domains offer flexible ways of removing inter-experimental variability where classical statistical methods often fail. While applications of these methods have been mostly limited to single-cell genomics, in this work, we develop a theoretical framework for domain adaptation in systems neuroscience. We implement this in an adversarial optimization scheme that removes inter-experimental variability while preserving the biological signal. We compare our method to previous approaches on a large-scale dataset of two-photon imaging recordings of retinal bipolar cell responses to visual stimuli. This dataset provides a unique benchmark as it contains biological signal from well-defined cell types that is obscured by large inter-experimental variability. In a supervised setting, we compare the generalization performance of cell type classifiers across experiments, which we validate with anatomical cell type distributions from electron microscopy data. In an unsupervised setting, we remove inter-experimental variability from data which can then be fed into arbitrary downstream analyses. In both settings, we find that our method achieves the best trade-off between removing inter-experimental variability and preserving biological signal. Thus, we offer a flexible approach to remove inter-experimental variability and integrate datasets across experiments in systems neuroscience. Code available at `https://github.com/eulerlab/rave`.

*‡Equal contributions, [1] University of Tübingen, [2] Norwegian University of Science and Technology.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

1 Introduction

Systems neuroscientists are often concerned with identifying and characterizing how properties of neurons vary along certain dimensions of interest. Differences in these properties between neurons form the basis for sorting them into discrete categories. Both the advance of large-scale data acquisition techniques in experimental neuroscience as well as the development of more efficient and powerful data analysis methods allow collecting and analyzing datasets of increasing size; and hence the discovery of more subtle variations in neural function between cell types [e.g. 1–4]. However, as data acquisition is often an incremental process, it has become common practice to pool data from multiple experiments. This practice ignores variability in the data stemming from external factors, which include non-biological ones (e.g. sample handling resulting in small differences in tissue quality, or temperature fluctuations affecting the rates of biochemical processes) but potentially also unforeseen biological ones (e.g. subtle genetic variations) [3, 5, 6]. Such variability due to external factors, here referred to as *inter-experimental variability*, can confound and obscure the biological signals of interest. In some cases, the source of inter-experimental variability is known and can be modeled [5], but if this is not the case, a method for removing it from the data is required.

The issue of inter-experimental variability in systems neuroscience is analogous to the problem of *domain shift* in machine learning, where the data distribution changes between training ('source') and test ('target') data, causing an algorithm to fail when deployed on data from an unseen target domain [7–10]. Methods that address this issue have to perform some form of *domain adaptation*, i.e. adapting the algorithm to work both on the training as well as some (usually unseen) test domain [11]. In single-cell genomics, a number of different studies have proposed methods for removing inter-experimental variability (see Section 2), but related works in systems neuroscience are lacking, despite the recognized need for such approaches [3, 5]. Here, we contribute to closing this gap as follows:

- We cast the removal of inter-experimental variability from functional data in systems neuroscience in the theoretical framework of domain adaptation (Figure 1 and Section 3).
- We adapt and evaluate different approaches and demonstrate improved performance of cell type assignment, while preserving the biological signals of interest (Table 1 and Figure 4).
- We demonstrate that our method produces cell type predictions on a new dataset that are best aligned with anatomical data (Table 2 and Figure 5).
- Finally, we showcase in a downstream analysis that the corrected data (Figure 3) clearly exhibits biological effects that were obscured by inter-experimental variability (Figure 6).

2 Related Work

As mentioned before, few studies have proposed specialized solutions to the issue of inter-experimental variability in systems neuroscience. Two studies have approached the problem of temporal alignment of neural responses across experiments. Zhao et al. [5] proposed a solution to deal with the specific effects of temperature fluctuations on the response kinetics of retinal neurons by modeling them explicitly. Williams et al. [12] proposed a more general method for the temporal alignment of data across trials or recording sessions. Other studies have suggested models of neural function that integrate data across experiments. Shah et al. [3] build encoding models to predict the responses of retinal ganglion cells across different experiments [see also 13] and compare it to covariates such as the gender of an animal. Sorochynsky et al. [14] propose a way to measure noise correlations in each recording and integrate those into models of neural populations of a specific cell type. This latter approach is complementary to our method because it allows the study of the structure and function [see also 15] of noise correlations, which we discard as nuisance variability. Somewhat related to the example application in our paper, Jouty et al. [16] suggested a method to perform non-parametric physiological classification of retinal ganglion cells in the mouse retina while trying to find matching clusters of cell types across experiments. Crucially, all of these approaches offer specialized solutions that do not represent general purpose correction methods.

In single-cell genomics, a number of approaches for removing inter-experimental variability from data have been developed [17–24]. Two such methods are *Harmony* [25] and *scGen* [26]. *Harmony* performs iterative clustering using a variant of soft k-means until convergence to align cells from different datasets in a joint embedding. *scGen*, on the other hand, combines a variational autoencoder

adapted for scRNA-seq data with latent space arithmetics to predict gene expression, while removing inter-experimental variability between datasets. In this paper, we compare our approach against these methods as they have been found to perform particularly well in two benchmarking studies [27, 28].

3 Theoretical Framework

The generative process of data denoted by a random variable X with image $\mathcal{X}$ is depicted in Figure 1. The biological signal shared across experiments (e.g. variation due to cell types) is represented by a random variable S ('signal') with image $\mathcal{S}$. We define D ('domain') as a random variable with image $\mathcal{D}$ that represents inter-experimental variability. Now, our objective is to learn a function f that transforms the data into a new random variable $Z := f(X)$ with image $\mathcal{Z}$. Importantly, we distinguish two settings: (*i*) *unsupervised* — where S is unknown and we simply try to retain in Z as much information about the data as possible while removing inter-experimental variability; (*ii*) *supervised* — where S is known and we additionally try to retain in Z as much information about S as possible. These objectives can be formulated in terms of mutual information, giving the unsupervised loss function

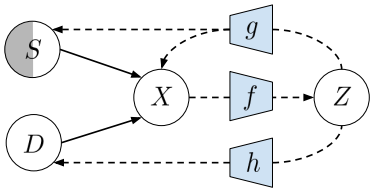

Figure 1: **Problem Setting**. Arrows represent given (solid lines) and modeled (dashed lines) relations. Capital letters denote random variables, small letters transformations (see Section 3). The setting with known S (white circle) is supervised, while unknown S (gray circle) is an unsupervised setting.

$$\mathcal{L} = I(Z;D) - I(Z;X) \tag{1}$$

and, provided knowledge about S, we obtain the supervised loss function

$$\mathcal{L}_+ = \mathcal{L} - I(Z;S). \tag{2}$$

Now, $I(Z;D)$ attains its minimum for $I(f(X);D) = 0$ because of the non-negativity of mutual information. And $I(Z;S)$ attains its maximum for $I(f(X);S) = I(X;S)$ because of the data processing inequality. If f were a bijection, it would follow that $I(f(X);S) = I(X;S)$, but also $I(f(X);D) = I(X;D)$. But by assumption, $I(X;D) > 0$ (otherwise there is no inter-experimental variability and we are done) and so we would have $I(f(X);D) > 0$. Thus, at the minimum of $I(Z;D)$, f cannot be a bijection. Generally, if there is an interaction between recording, signal and domain i.e. $I(X;S;D) \neq 0$, then there will be a *trade-off* between maximizing $I(f(X);S)$ and minimizing $I(f(X);D)$. This trade-off becomes even more apparent in the unsupervised setting where $I(f(X);X)$ and $I(f(X);D)$ are clearly competing.

Mutual information quantifies the dependence between two variables but it is difficult to estimate [29–32]. Instead, we measure dependency through nonlinear regression with an appropriate distance metric d.[2] Usually D is a discrete random variable indicating the experiment of a recording, and so, to estimate $I(Z;D)$, we can perform classification with a classifier function $h : \mathcal{Z} \to \mathcal{D}$, minimizing the standard cross-entropy $d_{\text{CE}}(h(Z), D)$ (Figure 1). Lemma 10 in [34] shows that this gives a variational lower bound to $I(Z;D)$ [see also 35]. In some cases S may also be discrete (e.g. cell types) and we can do the same, in other cases it might be a (high-dimensional) continuous random variable and so, to approximate $I(Z;S)$, we can perform regression, minimizing the mean squared error $d_{\text{MSE}}(g(Z), S)$. Similarly, in the unsupervised setting, to approximate $I(Z;X)$, we minimize $d_{\text{MSE}}(g(Z), X)$. To keep notation simple, in the unsupervised setting we define the mapping $g : \mathcal{Z} \to \mathcal{X}$, and in the supervised setting $g : \mathcal{Z} \to \mathcal{X} \times \mathcal{S}$. Putting this together, in the unsupervised setting our objective is

$$\min \mathcal{L} \longrightarrow \min_{h} \max_{g,f} \lambda\, d(h(f(X)), D) - d(g(f(X)), X) \tag{3}$$

where we have introduced a hyperparameter λ that mitigates the trade-off discussed above. In the supervised setting our objective becomes

$$\min \mathcal{L}_+ \longrightarrow \min_{h} \max_{g,f} \lambda\, d(h(f(X)), D) - d(g(f(X)), (X, S)). \tag{4}$$

[2] If the joint probability density of two random variables is a bivariate normal distribution, then the mutual information is proportional to their linear correlation [33].

In both equations we find a $\min - \max$ optimization, where h is trying to predict D from Z, tightening the lower bound on $I(Z;D)$ [see 34], while f is trying to prevent that by removing information about D from Z, effectively lowering $I(Z;D)$. Practically, this optimization scheme has become a standard adversarial setting in machine learning, for instance, in the training of generative adversarial networks [36] or for discriminative adversarial domain adaptation [37].

4 Methods

4.1 Datasets

To test our model approach, we use two datasets of two-photon imaging recordings [38–40] from the 14 mouse retinal bipolar cell (BC) [41] types' responses to two visual stimuli, a local and full-field chirp stimulus (Figure 3). The axon terminals of BCs stratify at distinct, cell type-specific depths within the second synaptic layer of the retina, the inner plexiform layer (IPL). The functional BC data were obtained by imaging the glutamate output at their axon terminals using the genetically encoded glutamate-sensing fluorescent reporter iGluSnFR [42]. In our study, we refer to these two datasets as *A* [2] and *B* [5] (for further preprocessing see Appendix).

In [2], an anatomy-guided functional clustering approach to group the BCs into the 14 functional types was applied to dataset *A*, thus providing functional reference cell type labels, which do not exist for dataset *B*. However, even if both datasets recorded the same cell types, they suffer from inter-experimental variability making it difficult to match and, for example, to use dataset *A* to predict the cell type labels for dataset *B*. We discuss potential sources of inter-experimental variability in the Appendix. For preprocessing, both BC datasets are high-pass filtered above 0.1 Hz (to remove the trends of decreasing fluorescence signal over time) and resampled to 30 Hz. Each cell's response is normalized to zero mean and standard deviation one. In addition, to ensure high quality responses, only cells with a sufficient response quality are used (for details about quality criterion see [2]).

4.2 Models

All methods transform the data, either into a low-dimensional embedding $z \in \mathcal{Z}$ or directly into a reconstruction $\hat{x} \in \mathcal{X}$ from which inter-experimental variability has been removed to a varying degree. Usually we have $\dim(\mathcal{Z}) \ll \dim(\mathcal{X})$ and so, for different downstream evaluations, we map between these representations: (*i*) with least squares reconstructions ($\mathcal{Z} \to \mathcal{X}$), or (*ii*) principle component projections ($\mathcal{X} \to \mathcal{Z}$) (see Appendix).

4.2.1 Unsupervised Model

We parameterize the functions f, g and h (Figure 1) with neural networks. In the unsupervised model, the function $g : \mathcal{Z} \to \mathcal{X}$ provides a reconstruction $\hat{x} := g(z),\ \hat{x} \in \mathcal{X}$ of the data. With the concurrent task (eq. 3) of minimizing the predictability of the domain D, this reconstruction should only contain parts of the original data that are indiscernible across experiments. Since the purpose of our method is to Remove, Adversarially, Variability from datasets collected in different Experiments, we term our model `RAVE`.

4.2.2 Supervised Model

In the supervised setting, we have partial knowledge about the biological signal S. The function $g : \mathcal{Z} \to \mathcal{X} \times \mathcal{S}$ now returns a reconstruction as well as a prediction of that signal $(\hat{x}, \hat{s}) := g(z),\ \hat{s} \in \mathcal{S}$. When optimizing equation (4), this additional task is equivalent to discriminative adversarial domain adaptation [37]. In the particular data that we work with, we have two datasets $\mathcal{D} := \{A, B\}$, but the biological signal S consists of cell type labels which are only available in the first dataset A. Thus, more accurately, this presents a *semi-supervised* scenario where one wishes to classify a newly recorded dataset according to some existing classification scheme. We term this extended version of our model `RAVE+`.

4.2.3 Training and Optimization Details

All of our models are implemented and optimized in PyTorch [43]. For both `RAVE` and `RAVE+`, we use the same model architecture, they only differ in the objective function. We randomly split the data

into training, validation and test set and train all models with empirical risk minimization. Model weights are trained with stochastic gradient descent using one instance of the Adam optimizer [44] for the outer minimization of f and g in equations (3) and (4); and then a second instance of Adam for the inner minimization of h in those equations. We optimize hyperparameters through random search [45] on the validation set and report performances on the test set which is only used for final evaluation. In the random search, we test different learning rates for both optimizers, and also different training schedules. We additionally search over depth, width and drop-out rate for each of the neural networks (f, g, h), as well as the trade-off parameter λ introduced in equation (3). Finally, we explore training the inner optimization (h, estimating $I(Z; D)$) more often than the outer optimization, which proved more stable and effective in early experiments.

4.2.4 Comparison Models

We test three different methods for comparison with our model. Our simplest comparison model (*Linear*) is a linear model that projects out the contrast between the dataset indicator variables (see Appendix). This has an analytic solution and no hyperparameters, serving as a baseline to get an estimate of the correction quality achieved by a standard method in classical statistics. The other two methods (*scGen* and *Harmony*) are run in an unsupervised learning mode without cell type information. Even though *scGen* could be utilized to run in a supervised mode with cell type information, this is not specified in a semi-supervised setting with only partial cell type labels available.

4.3 Performance Evaluation

For evaluating the correction performed by the various methods, we analyze their output with respect to dataset-mixing (achieved by removing inter-experimental variability) and preservation of signal information.

4.3.1 Dataset-Mixing

The Rand index [46] measures similarity between two clusterings; the adjusted Rand index (ARI) is the Rand index adjusted for chance level (see Appendix) which was recently used by Tran et al. [28] to assess the quality of dataset-mixing in genomics. It takes as input the true and the predicted labels for a set of samples. We define $\text{ARI}_{dom}(z) := \text{ARI}(d, \hat{d}_z)$ with d the original domain labels (of the test set) and $\hat{d}_z$ the domain labels predicted by a classifier trained on z. On the raw data ($z = x$), we expect ARI_{dom} to be high due to inter-experimental variability. After successful correction (with z the output of a model), we expect ARI_{dom} to be low indicating good dataset-mixing.

In addition, we compute the accuracy (Acc_{dom}) of a domain classifier with the objective to predict the domain labels based on the input data. For low dataset-mixing, we expect a high Acc_{dom} as it should be trivial for the classifier to differentiate the datasets. However, after removing inter-experimental variability, Acc_{dom} is supposed to be close to chance level ($\sim 64\%$, cf. Table 1), which would indicate successful dataset-mixing. For the domain classifier, we use a random forest classifier with cross-validated hyperparameters for each model (see Appendix). This is crucial, because a powerful encoder f might hide (through multiple nonlinear transformations) domain information from a simple classifier, but still recover that information in an equally powerful decoder g. Conversely, we observe that overly expressive random forest classifiers, tend to overfit on the training set, thus underestimating the preserved domain or type information on the test set.

4.3.2 Preservation of Signal Information

In the unsupervised setting, to assess the amount of information preserved about the original data x during the process of removing inter-experimental variability, we evaluate the rank correlation $\text{Corr}(x, \hat{x})$ between input x and reconstruction $\hat{x}$. In the (semi-) supervised setting, we have reference cell type labels s_A for dataset A. To estimate how much of this information is preserved, we predict cell type labels $\hat{s}_A$ from z_A with random forest classifiers like above (see Figure 2; Appendix for further details).

If a method succeeds at preserving signal information in z after removing inter-experimental variability, then we expect the classifier to have a high accuracy (Acc_{type}). Deteriorating classification

performance between predicting $\hat{s}_A$ from raw data x_A versus predicting it from the model output z_A would indicate signal loss.

Additionally, we would like to evaluate how well cell types can be distinguished and how biologically plausible they are for the unlabeled dataset *B*. To this end, we apply the classifiers to predict cell type labels $\hat{s}_B$ from z_B. One direct comparison is between the distributions over cell types in $\hat{s}_B$ and as expected from electron-microscopy (EM) data [47, 48] (Figure 5A). However, we can also evaluate the accuracy of these predictions by making use of BC axonal stratification profiles obtained from the same EM data. From those data, we know where in the IPL a BC type stratifies its axon terminals. Thus, we can compare the distribution over IPL depth for the predicted cell types ($\hat{s}_B$) with the distributions expected from EM data. We quantify the difference between the expected and predicted distributions by calculating the Jensen-Shannon distance. We define the depth score (DS) as the mean Jensen-Shannon distance between those two distributions (Table 2). Additionally, we evaluate the robustness of cell type labels $\hat{s}_B$ by fitting the classifier ten times with different seeds and calculating the average ARI between different runs, giving the ARI_{type} score (Table 1).

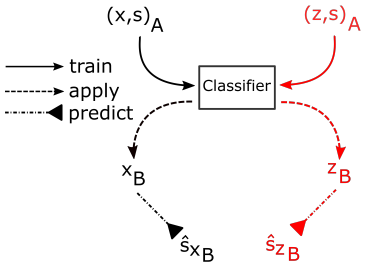

Figure 2: **Workflow.** Evaluating the preservation of signal information: A classifier gets trained on the labeled dataset *A* (either $(x, s)_A$ or $(z, s)_A$) and applied to dataset *B* to predict labels (either $\hat{s}_{x_B}$ or $\hat{s}_{z_B}$). The predicted labels are then used for further evaluation.

5 Results

5.1 Simulation Experiments

First of all, we validated that our model performs as expected on simulated ground truth data. To do this, we generated bipolar cell responses for all 14 cell types based on the published bipolar cell model in Schröder et al. [49]. To simulate different individual neurons, we added small perturbations to the model weights for each cell type until we matched the intra-cell-type variability observed in the real data. Thus, we generated $N = 1000$ distinct neurons for each of the 14 BC types. Approximating the differences of the two datasets in the paper, we presented the model with the slightly altered versions of the stimulus from the actual experiments (see Appendix B). We additionally added white noise to match realistic signal to noise ratios, as estimated from repeated stimulations of the real neurons. This resulted in two datasets ('A' and 'B') with similar intra- and inter-experimental variability as observed in the real data, but with known ground truth cell type labels.

The results are discussed here and, additionally, they are presented in Appendix Fig. 9. We first confirmed that the classifiers are indeed perfectly able to separate these two artificial datasets based on their systematic differences (domain accuracy on raw simulated data: 1.0). However, cell type classifiers trained on dataset A fail completely on dataset B indicating severe inter-experimental variability and a failure to transfer models across datasets (type accuracy on dataset A: 0.98, type accuracy on dataset B: 0.16). In contrast, after correction with RAVE+, the performance of a classifier trained to distinguish the two datasets drops from 1.0 to 0.66 (chance level 0.5), indicating strong removal of inter-experimental variability. Importantly, we find that a classifier trained on the output of RAVE+ on dataset A does now generalize to dataset B and recovers the ground truth cell labels nearly perfectly (type accuracy 0.99). This constitutes an important validation of our model.

5.2 Unsupervised Removal of Inter-Experimental Variability

All methods tested (*Linear*, *Harmony*, *scGen*, `RAVE` and `RAVE+`) succeed at retaining a significant amount of information about x in $\hat{x}$, reflected by high correlations between x and $\hat{x}$ (Table 1). $\text{Corr}(x, \hat{x})$ reaches similar levels for data from both datasets, suggesting that both datasets are modified to find a midway representation. This impression is confirmed when visualizing x_A, x_B and $\hat{x}_A$ and $\hat{x}_B$ next to each other (Figure 3). Moreover, we note that a side-effect of the alignment is a more general denoising that `RAVE` tends to perform along with removing inter-experimental variability. We recognize that this a desirable feature that we will study further in future research.

Model	$\text{Corr}_A \uparrow$	$\text{Corr}_B \uparrow$	$\text{Acc}_{dom} \downarrow$	$\text{Acc}_{type} \uparrow$	$\text{ARI}_{dom} \downarrow$	$\text{ARI}_{type} \uparrow$
Raw	100	100	99.8 (0.1)	77.4 (0.9)	99.3 (0.4)	37.3 (3.3)
Linear	**99.0** (0.5)	**97.0** (1.7)	99.5 (0.2)	83.4 (0.8)	98.1 (0.7)	7.6 (1.7)
Harmony	72.0 (10.7)	72.0 (13.8)	94.2 (0.4)	82.5 (0.5)	78.0 (1.6)	31.4 (2.3)
scGen	78.0 (9.8)	80.0 (10.5)	99.6 (0.1)	**84.7** (0.8)	98.7 (0.3)	14.3 (2.6)
RAVE	60.0 (12.8)	58.0 (17.3)	77.5 (0.5)	69.5 (0.4)	28.9 (1.2)	81.2 (2.7)
RAVE+	59.0 (14.8)	58.0 (19.1)	**65.9** (0.9)	78.6 (0.8)	**10.0** (1.2)	**83.7** (2.3)

Table 1: **Model Comparison.** All entries in percentage. Mean and standard deviation metric scores across 10 random seeds. Bold font in each row indicates best score. Corr_A (Corr_B) is the correlation of corrected data from dataset *A* (*B*) with its raw data. Acc_{dom} (Acc_{type}) is the accuracy of the domain (cell type) classifier. For ARI_{dom} and ARI_{type} see Section 4.3.

We show mean traces for exemplary cell types from dataset *A*, and mean traces of cells from dataset *B* whose cell type labels we predict twice, first based on x (left pathway in Figure 2) and then again based on $\hat{x}$ (right pathway in Figure 2, but on $\hat{x}_{\texttt{RAVE}}$ instead of z). As expected, inter-experimental variability obscuring the common signal s behind x_A and x_B causes the cell type assignment to fail; the similarity between responses of cells assigned to the same cell type, but coming from the different datasets is low (Figure 3, BC type 5t). Repeating the classification pipeline based on x with the same classifier architecture and different seeds yields highly variable cell type predictions for dataset *B* (Table 1, ARI_{type}) despite high prediction accuracy on dataset *A* (Table 1, Acc_{type}). This demonstrates a failure in transferring to dataset *B*, and not the classification itself. These results on the raw data x affected by inter-experimental variability were expected; however, the same pattern - low ARI_{type} (dataset *B*) and high Acc_{type} (dataset *A*) - is observed for *Harmony*, *scGen* and the *Linear* model. This suggests that these methods fail at removing inter-experimental variability. The high domain accuracy achieved by a classifier trained on the outputs of these models confirms this conclusion. RAVE, on the other hand, succeeds at significantly lowering domain accuracy Acc_{dom}, while at the same time maintaining high scores for ARI_{type} and Acc_{type}.

5.3 Supervised Removal of Inter-Experimental Variability

RAVE+ extends RAVE to the (semi-) supervised setting where (partial) signal information is present. RAVE+ excels at removing inter-experimental variability (Table 1, Acc_{dom} and ARI_{dom}) and at the same time retaining signal information (Table 1, Acc_{type} and ARI_{type}). A low dimensional t-SNE

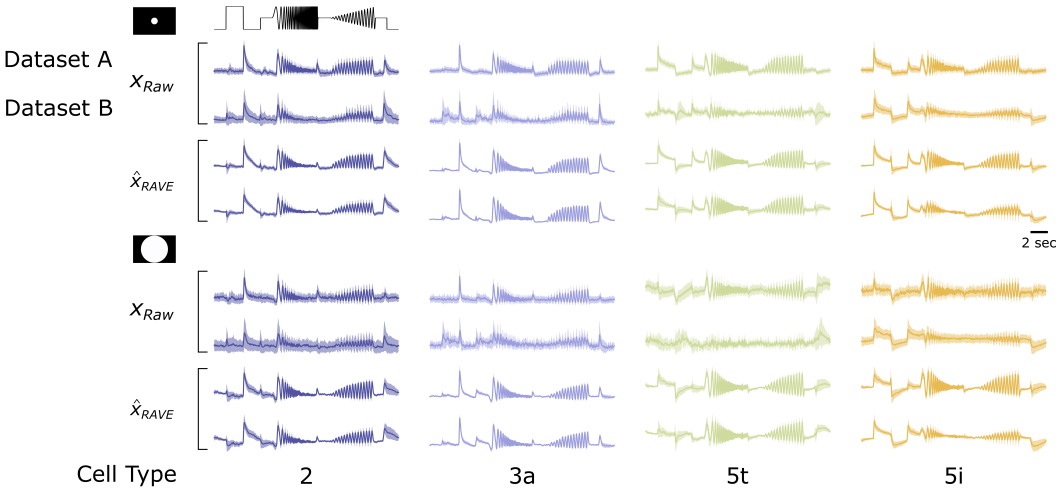

Figure 3: **Exemplary Cell Type Responses from both Datasets to the Chirp Stimuli.** Four bipolar cell type responses of the types 2, 3a, 5t and 5i to the local (top panel) and full-field (bottom panel) chirp of raw data x_{Raw} (two top upper panels) and reconstructed data $\hat{x}_{\texttt{RAVE}}$ (two bottom lower panels) by RAVE for both datasets *A* and *B*. Each column shows the mean responses of one cell type (standard deviation shaded).

[50] embedding (Figure 4) shows that cells from datasets *A* and *B* are mapped onto the same cell type "islands". The distribution of types across IPL depth predicted by a classifier trained on $z_{\texttt{RAVE+}}$ matches the expected anatomical distributions better than for all other methods (Figure 5 and Table 2). This provides a valuable validation of the estimate $\hat{s}_B$ learned by RAVE+ in the absence of ground truth knowledge of s_B.

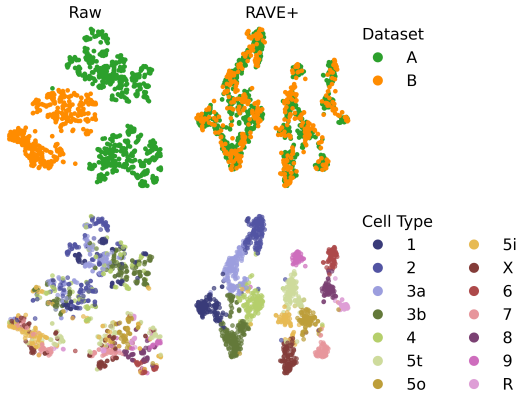

Figure 4: **Dataset Embeddings**. t-SNE embeddings of the test set of raw (left column) and corrected output data by RAVE+ (right column). Embedded cells are color-coded by dataset (top row) and cell type (bottom row). Cell type labels for the raw data of dataset *B* (bottom left) were predicted using a cell type classifier trained on the raw data of dataset *A* (Figure 2, left pathway).

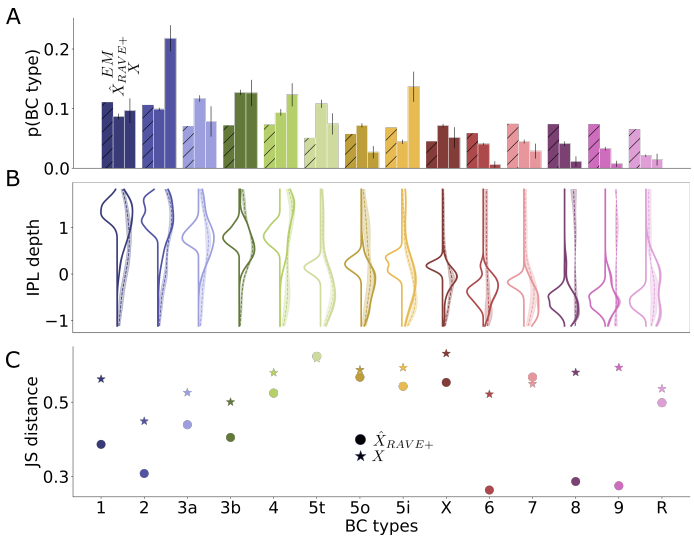

Figure 5: **Distribution Across BC Types and IPL Depths.** We compare the expected and predicted distribution of BCs from dataset *B* across the 14 types and across IPL depth. **(A)** Probability that a BC belongs to a certain type as estimated from EM data; as estimated from BC type labels predicted on $\hat{x}_{\texttt{RAVE+}}$; and as estimated from BC type labels predicted on x. Error bars indicate SD across 10 seeds of the classifier. (**B**) Distributions per cell type over IPL depth for EM data (distribution shown to the left), RAVE+ output (solid line to the right) and raw data (dashed line to the right). Shaded area around the distributions shown to the right indicate SD across 10 seeds of the classifier. (**C**) JS distances corresponding to the distributions in B).

5.4 Downstream Analyses on Reconstructed Traces

As in our unsupervised setting, it is common that no particular signal information is available and that one wants to remove inter-experimental variability from the data to perform further downstream

BC Type	1	2	3a	3b	4	5t	5o	5i	X	6	7	8	9	R	∅	all
Raw	**34**	52	**38**	42	56	63	62	54	60	52	**51**	38	58	**37**	50	31
Linear	58	40	53	51	57	62	57	56	62	55	57	58	58	53	56	34
Harmony	42	32	42	47	54	59	58	56	61	37	59	**26**	58	38	48	23
scGen	51	38	48	48	56	59	**56**	59	63	55	57	56	56	42	53	31
RAVE	41	32	**38**	**40**	57	**58**	62	55	66	55	55	38	50	50	50	23
RAVE+	38	**30**	43	**40**	**52**	62	**56**	**54**	**55**	**26**	56	28	**27**	49	**44**	**17**

Table 2: **Depth Score Comparison.** All entries in percentage, lower is better. Bold font in each row indicates best score. Depth Score - Jensen-Shannon (JS) distance between predicted types and EM depth distribution: $JS(p_{EM}(depth|type = t), p_{model}(depth|type = t))$. Last column ("all"): $JS(p_{EM}(type), p_{model}(type))$.

analyses. We show that a previously demonstrated biological effect, obscured by inter-experimental variability in x, emerges when performing the same analyses on the reconstructed traces $\hat{x}$ obtained from RAVE. Full-field visual stimulation has been shown to decorrelate responses from different BC types compared to local stimulation due to inhibitory feedback from amacrine cells (see Figure 3A, B in [2]). We expect this fundamental feature to be present in dataset *B*, but cannot fully reproduce it if we assign cells of dataset *B* to cell types based on the raw data (Figure 2, left pathway; and Figure 6A). However, using the reconstructed traces $\hat{x}_{\texttt{RAVE}}$, the expected feature is unmasked (Figure 2, right pathway, but on $\hat{x}_{\texttt{RAVE}}$ instead of z; and Figure 6B). Here, the mean responses to the local chirp are more correlated across cell types than the full-field responses (Figure 6B, left panel). This can also be seen when comparing the mean correlations between local and full-field chirp responses for each cell type with all other cell types, both of the same and the opposite response polarity (On and Off polarity) (Figure 6B, right panel).

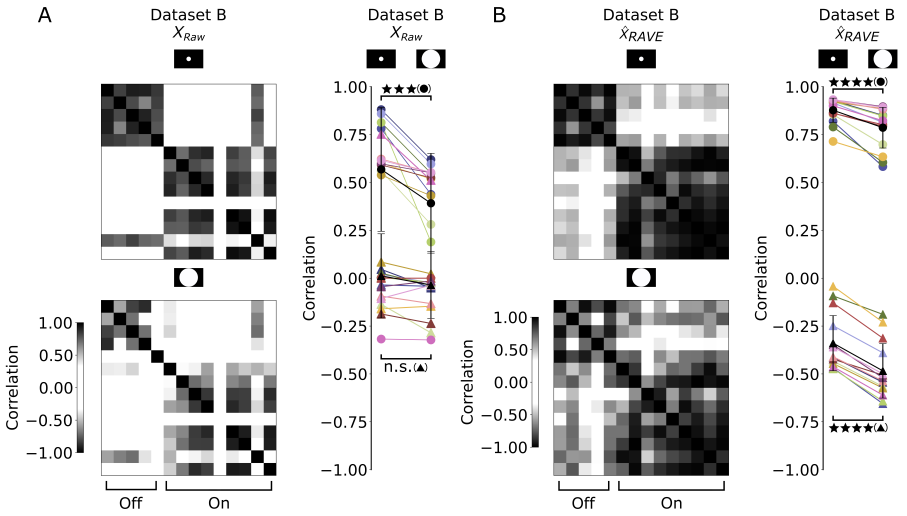

Figure 6: **Removing Inter-Experimental Variability Reveals Biological Feature.** (**A**) Correlation matrices show the correlations between mean responses per cell type to local (top) and full-field (bottom) chirp of raw data x from dataset *B*. The right panel represents the mean correlation for each cell type mean response with all other types of the same (circle) and opposite (triangle) response polarity between local and full-field chirp shown for raw responses. (x: mean correlation same polarity: p_{local} = 0.57 and $p_{full\text{-}field}$ = 0.39, P < 0.005; opposite polarity: p_{local} = 0.01 and $p_{full\text{-}field}$ = -0.04, n.s.; n = 14, non-parametric paired Wilcoxon signed-rank test).(**B**) Same analysis as **A**, but with the reconstructed responses obtained from RAVE. ($\hat{x}_{\texttt{RAVE}}$: mean correlation same polarity: p_{local} = 0.88 and $p_{full\text{-}field}$ = 0.79, P < 0.0005; opposite polarity: p_{local} = -0.34 and $p_{full\text{-}field}$ = -0.48, P < 0.0005; n = 14, non-parametric paired Wilcoxon signed-rank test).

6 Limitations

Our method is limited to datasets where neurons were presented with the same stimulus. For other kinds of data, such as neural recordings from free behavioral paradigms where each trial will be different, it will be difficult to 'align' neural responses in a meaningful way. One solution to this could be to learn a shared embedding space [see 51], from which domain effects are removed, but distinct encoders f_i and decoders g_i for different trials i. In another setting, where different stimuli are presented between experiments, one might resort to an approach like Shah et al. [3]. Nevertheless, we do acknowledge that the data in our applications consists of *ex vivo* retinal recordings which have little to no attentional effects or task-dependent noise correlations like they would be present in *in vivo* cortical data. We are optimistic that our framework of adversarially removing inter-experimental variability is still a promising approach in those settings, under the constraint that a much more severe trade-off may need to be made between retaining signal and removing domain shifts.

7 Discussion

We present a framework to remove inter-experimental variability from functional recordings in systems neuroscience. To the best of our knowledge, this is the first application of domain adaptation methods to this kind of data. Using our unsupervised (`RAVE`) and (semi-) supervised (`RAVE+`) approaches, we demonstrate that we are able to remove inter-experimental variability while retaining signal information, which allows us to robustly predict cell type labels for a new dataset. We validate those predictions using an anatomy-based comparison to existing EM data.

Furthermore, our unsupervised approach `RAVE` is able to remove inter-experimental variability without cell type information. By using the corrected dataset B, we unmask biological effects, obscured by inter-experimental variability, that have been previously described for dataset A. Thus, by allowing the integration and alignment of functional recordings across experiments, we show that biological effects in the data become more pronounced when using our model approaches. Inter-experimental variability is ubiquitous and we hope that this method will become a helpful resource to many experimenters as we make the code toolbox publicly available.

We believe that our method can also make a contribution to systems neuroscience research in the context of the 3Rs (Replacement, Reduction and Refinement) for animal ethics: By enabling detection of more subtle biological signals after removal of inter-experimental variability, fewer animals may be needed to test a specific hypothesis. Lastly, we acknowledge that the removal of inter-experimental variability from any kind of data (thus not only within systems neuroscience) can be useful in various applications. Virtually any analysis that aggregates data across experiments can be confounded by inter-experimental variability. Consequently, we cannot exclude the possibility that some military application will find value in this approach. Although unlikely, we cannot fully anticipate such developments. Therefore we condemn, without any exceptions, the use of `RAVE(+)` for any warlike applications or other nefarious purposes.

8 Acknowledgments and Disclosure of Funding

This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) project number 335549539/GRK2381 and the CRC 1233 "Robust Vision" (grant number 276693517). Moreover, this work was partially supported by a Research Council of Norway FRIPRO grant (90532703). PB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645 and the Tübingen AI Center (FKZ: 01IS18039A). He was supported through a Heisenberg Professorship by the DFG (BE5601/8-1).

References

[1] Tom Baden, Philipp Berens, Katrin Franke, Miroslav Román Rosón, Matthias Bethge, and Thomas Euler. The functional diversity of retinal ganglion cells in the mouse. *Nature*, 529(7586):345–350, 2016.

[2] Katrin Franke, Philipp Berens, Timm Schubert, Matthias Bethge, Thomas Euler, and Tom Baden. Inhibition decorrelates visual feature representations in the inner retina. *Nature*, 542(7642):439–444, 2017.

[3] Nishal Shah, Nora Brackbill, Ryan Samarakoon, Colleen Rhoades, Alexandra Kling, Alexander Sher, Alan Litke, Yoram Singer, Jonathon Shlens, and EJ Chichilnisky. Individual variability of neural computations in the primate retina. *bioRxiv*, pages 1–22, 2021.

[4] Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al. A deep learning framework for neuroscience. *Nature neuroscience*, 22(11):1761–1770, 2019.

[5] Zhijian Zhao, David A. Klindt, André Maia Chagas, Klaudia P. Szatko, Luke Rogerson, Dario A. Protti, Christian Behrens, Deniz Dalkara, Timm Schubert, Matthias Bethge, Katrin Franke, Philipp Berens, Alexander S. Ecker, and Thomas Euler. The temporal structure of the inner retina at a single glance. *Scientific Reports*, 10(1):1–17, 2020.

[6] Stephanie C Hicks, F William Townes, Mingxiang Teng, and Rafael A Irizarry. Missing data and technical variability in single-cell rna-sequencing experiments. *Biostatistics*, 19(4):562–578, 2018.

[7] Jing Jiang. A literature survey on domain adaptation of statistical classifiers. *URL: http://sifaka. cs. uiuc. edu/jiang4/domainadaptation/survey*, 3:1–12, 2008.

[8] John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman. Learning bounds for domain adaptation. 2008.

[9] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

[10] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

[11] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

[12] Alex H Williams, Ben Poole, Niru Maheswaranathan, Ashesh K Dhawale, Tucker Fisher, Christopher D Wilson, David H Brann, Eric M Trautmann, Stephen Ryu, Roman Shusterman, et al. Discovering precise temporal patterns in large-scale neural recordings through robust and interpretable time warping. *Neuron*, 105(2):246–259, 2020.

[13] Fabian Sinz, Alexander S Ecker, Paul Fahey, Edgar Walker, Erick Cobos, Emmanouil Froudarakis, Dimitri Yatsenko, Zachary Pitkow, Jacob Reimer, and Andreas Tolias. Stimulus domain transfer in recurrent models for large scale cortical population prediction on video. *Advances in Neural Information Processing Systems*, 31:7199–7210, 2018.

[14] Oleksandr Sorochynskyi, Stéphane Deny, Olivier Marre, and Ulisse Ferrari. Predicting synchronous firing of large neural populations from sequential recordings. *PLoS computational biology*, 17(1):e1008501, 2021.

[15] Alexander S Ecker, Philipp Berens, R James Cotton, Manivannan Subramaniyan, George H Denfield, Cathryn R Cadwell, Stelios M Smirnakis, Matthias Bethge, and Andreas S Tolias. State dependence of noise correlations in macaque primary visual cortex. *Neuron*, 82(1):235–248, 2014.

[16] Jonathan Jouty, Gerrit Hilgen, Evelyne Sernagor, and Matthias H Hennig. Non-parametric physiological classification of retinal ganglion cells in the mouse retina. *Frontiers in Cellular Neuroscience*, 12:481, 2018.

[17] Gabriele Beate Schweikert, Christian Widmer, Bernhard Schölkopf, and Gunnar Rätsch. An empirical analysis of domain adaptation algorithms for genomic sequence analysis. In *NIPS*, volume 8, pages 1433–1440, 2008.

[18] Jonathan Bryan Dayton. Adversarial deep neural networks effectively remove nonlinear batch effects from gene-expression data. 2019.

[19] Tongxin Wang, Travis S Johnson, Wei Shao, Zixiao Lu, Bryan R Helm, Jie Zhang, and Kun Huang. Bermuda: a novel deep transfer learning method for single-cell rna sequencing batch correction reveals hidden high-resolution cellular subtypes. *Genome biology*, 20(1):1–15, 2019.

[20] Jingshu Wang, Divyansh Agarwal, Mo Huang, Gang Hu, Zilu Zhou, Chengzhong Ye, and Nancy R Zhang. Data denoising with transfer learning in single-cell transcriptomics. *Nature methods*, 16(9):875–878, 2019.

[21] Dongfang Wang, Siyu Hou, Lei Zhang, Xiliang Wang, Baolin Liu, and Zemin Zhang. imap: integration of multiple single-cell datasets by adversarial paired transfer networks. *Genome biology*, 22(1):1–24, 2021.

[22] Xiangjie Li, Kui Wang, Yafei Lyu, Huize Pan, Jingxiao Zhang, Dwight Stambolian, Katalin Susztak, Muredach P Reilly, Gang Hu, and Mingyao Li. Deep learning enables accurate clustering with batch effect removal in single-cell rna-seq analysis. *Nature communications*, 11(1):1–14, 2020.

[23] Eugene Lin, Sudipto Mukherjee, and Sreeram Kannan. A deep adversarial variational autoencoder model for dimensionality reduction in single-cell rna sequencing analysis. *BMC bioinformatics*, 21(1):1–11, 2020.

[24] Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):1–14, 2019.

[25] Ilya Korsunsky, Nghia Millard, Jean Fan, Kamil Slowikowski, Fan Zhang, Kevin Wei, Yuriy Baglaenko, Michael Brenner, Po-ru Loh, and Soumya Raychaudhuri. Fast, sensitive and accurate integration of single-cell data with harmony. *Nature methods*, 16(12):1289–1296, 2019.

[26] Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. scgen predicts single-cell perturbation responses. *Nature methods*, 16(8):715–721, 2019.

[27] Malte D Luecken, Maren Buttner, Kridsadakorn Chaichoompu, Anna Danese, Marta Interlandi, Michaela F Müller, Daniel C Strobl, Luke Zappia, Martin Dugas, Maria Colomé-Tatché, et al. Benchmarking atlas-level data integration in single-cell genomics. *BioRxiv*, 2020.

[28] Hoa Thi Nhu Tran, Kok Siong Ang, Marion Chevrier, Xiaomeng Zhang, Nicole Yee Shin Lee, Michelle Goh, and Jinmiao Chen. A benchmark of batch-effect correction methods for single-cell rna sequencing data. *Genome biology*, 21(1):1–32, 2020.

[29] Rudy Moddemeijer. On estimation of entropy and mutual information of continuous distributions. *Signal processing*, 16(3):233–248, 1989.

[30] Liam Paninski. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003.

[31] Janett Walters-Williams and Yan Li. Estimation of mutual information: A survey. In *International Conference on Rough Sets and Knowledge Technology*, pages 389–396. Springer, 2009.

[32] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR, 2019.

[33] Izrail Moiseevich Gel'fand and Akiva Moiseevich Yaglom. Computation of the amount of information about a stochastic function contained in another such function. *Uspekhi Matematicheskikh Nauk*, 12(1):3–52, 1957.

[34] Jörn-Henrik Jacobsen, Jens Behrmann, Richard Zemel, and Matthias Bethge. Excessive invariance causes adversarial vulnerability. *arXiv preprint arXiv:1811.00401*, 2018.

[35] David Barber and Felix V Agakov. Information maximization in noisy channels: A variational approach. *Advances in Neural Information Processing Systems*, 16:201–208, 2003.

[36] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[37] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[38] Winfried Denk, James H Strickler, and Watt W Webb. Two-photon laser scanning fluorescence microscopy. *Science*, 248(4951):73–76, 1990.

[39] Winfried Denk and Peter B Detwiler. Optical recording of light-evoked calcium signals in the functionally intact retina. *Proceedings of the National Academy of Sciences*, 96(12):7035–7040, 1999.

[40] Thomas Euler, Susanne E Hausselt, David J Margolis, Tobias Breuninger, Xavier Castell, Peter B Detwiler, and Winfried Denk. Eyecup scope—optical recordings of light stimulus-evoked fluorescence signals in the retina. *Pflügers Archiv-European Journal of Physiology*, 457(6):1393–1414, 2009.

[41] Thomas Euler, Silke Haverkamp, Timm Schubert, and Tom Baden. Retinal bipolar cells: elementary building blocks of vision. *Nature Reviews Neuroscience*, 15(8):507–519, 2014.

[42] Jonathan S. Marvin, Bart G. Borghuis, Lin Tian, Joseph Cichon, Mark T. Harnett, Jasper Akerboom, Andrew Gordus, Sabine L. Renninger, Tsai Wen Chen, Cornelia I. Bargmann, Michael B. Orger, Eric R. Schreiter, Jonathan B. Demb, Wen Biao Gan, S. Andrew Hires, and Loren L. Looger. An optimized fluorescent probe for visualizing glutamate neurotransmission. *Nature Methods*, 10(2):162–170, 2013.

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[45] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

[46] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

[47] Jinseop S. Kim, Matthew J. Greene, Aleksandar Zlateski, Kisuk Lee, Mark Richardson, Srinivas C. Turaga, Michael Purcaro, Matthew Balkam, Amy Robinson, Bardia F. Behabadi, Michael Campos, Winfried Denk, and H. Sebastian Seung. Space-time wiring specificity supports direction selectivity in the retina. *Nature*, 509(7500):331–336, 2014.

[48] Matthew J. Greene, Jinseop S. Kim, and H. Sebastian Seung. Analogous Convergence of Sustained and Transient Inputs in Parallel On and Off Pathways for Retinal Motion Computation. *Cell Reports*, 14(8):1892–1900, 2016.

[49] Cornelius Schröder, David A. Klindt, Sarah Strauss, Katrin Franke, Matthias Bethge, Thomas Euler, and Philipp Berens. System identification with biophysical constraints: A circuit model of the inner retina. *bioRxiv*, 2020.

[50] Geoffrey Hinton and Sam T Roweis. Stochastic neighbor embedding. In *NIPS*, volume 15, pages 833–840. Citeseer, 2002.

[51] Rohan Gala, Nathan Gouwens, Zizhen Yao, Agata Budzillo, Osnat Penn, Bosiljka Tasic, Gabe Murphy, Hongkui Zeng, and Uygar Sümbül. A coupled autoencoder approach for multi-modal analysis of cell types. *arXiv preprint arXiv:1911.05663*, 2019.

[52] Klaudia P Szatko, Maria M Korympidou, Yanli Ran, Philipp Berens, Deniz Dalkara, Timm Schubert, Thomas Euler, and Katrin Franke. Neural circuits in the mouse retina support color vision in the upper visual field. *Nature communications*, 11(1):1–14, 2020.

[53] DI Vaney. 'coronate' amacrine cells in the rabbit retina have the 'starburst'dendritic morphology. *Proceedings of the Royal society of London. Series B. Biological sciences*, 220(1221):501–508, 1984.

[54] Katrin Franke, André Maia Chagas, Zhijian Zhao, Maxime J.Y. Zimmermann, Philipp Bartel, Yongrong Qiu, Klaudia P. Szatko, Tom Baden, and Thomas Euler. An arbitrary-spectrum spatial visual stimulator for vision research. *eLife*, 8:1–28, 2019.

[55] F Alexander Wolf, Philipp Angerer, and Fabian J Theis. Scanpy: large-scale single-cell gene expression data analysis. *Genome biology*, 19(1):1–5, 2018.

[56] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[57] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[58] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[59] Pavlin G Poličar, Martin Stražar, and Blaž Zupan. opentsne: a modular python library for t-sne dimensionality reduction and embedding. *BioRxiv*, page 731877, 2019.

Appendix

A Model details

A.1 Linear mappings between z and x

Usually, we have data $x \in \mathbb{R}^{N \times D_1}$ and latent representation $z \in \mathbb{R}^{N \times D_2}$ with N the number of neurons, D_1 the dimensionality of the data, D_2 the dimensionality of the latent space and, usually, $D_1 \gg D_2$. In cases where a method m does only produce some latent representation z_m, we fit a reconstruction $\hat{x}_m = W z_m$ with a least squares projection $W = (z_m^T z_m)^{-1} z_m^T x$. In cases where a method m does only produce some reconstruction $\hat{x}_m$, we produce a simple latent representation z_m by extracting the first D_2 columns of the left singular vectors U from the singular value decomposition $x = USV^T$. Both of these projections are fitted on the training data, then fixed and also used on the validation and test data.

B Data

We used three datasets, where the first two (dataset *A* [2] n=8417 cells; *B* [52] n=4600) are two-photon recordings of mouse retinal bipolar cell (BC) responses to the chirp stimuli (local and full-field, see [2] for details). Both datasets were used for model fitting and removal of inter-experimental variability. For the validation of cell type predictions made by the different models, we used the third dataset, which comprises EM data of axonal stratification profiles as probability distribution of each BC type [47, 48].

The inter-experimental variability between the two functional datasets may originate from, at least, the three following differences between the datasets: (*i*) dataset *A* recorded BCs mostly at certain IPL depths ('ChAT-bands', which are landmarks within the IPL [53]) using tangential scans parallel to the retinal layers, whereas dataset *B* used axial scans employing an electrically tunable lens to record from BCs across the entire IPL simultaneously [5], resulting in different sampling distributions; (*ii*) the chirp stimulus used in dataset *B* differs slightly as the sinusoidal intensity modulation of the increasing frequency is marginally slower; (*iii*) dataset *A* did not employ a gamma correction of the display device to linearize its intensity curve, resulting in slightly different stimulus contrasts [54].

C Training Results

The outcome of the random search can be seen in Figure 7, showing metrics on the validation set for both models. To select the best `RAVE` model, we picked the point in the top right corner (center plot, first row, Figure 7). This was the model with the highest $I(Z;X)$, i.e. correlation, and the lowest $I(Z;D)$, i.e. domain classification accuracy. To select the best `RAVE+` model, we picked the (`RAVE+`) point in the top right corner of the $3D$ space spanned by $\{I(Z;X), I(Z;S), -I(Z;D)\}$, i.e. the model with the best reconstruction and cell type prediction accuracy but with the lowest domain prediction accuracy.

Moreover, Figure 7 also demonstrates the trade-off between maximizing $I(Z;X)$ and $I(Z;S)$ and minimizing $I(Z;D)$. In the top row on the left, one can see that models with high $I(Z;X)$ also tend to have a high $I(Z;S)$, indicating that these two tasks can be performed well at the same time (this is what we mean by 'synergy' in the title; naturally, we cannot make a causal statement here). In the top row middle, one can see for models that achieve a high $I(Z;X)$ (some hyperparameter configurations in the random search simple lead to bad models), that there is a negative slope with respect to $I(Z;D)$, indicating that there is a trade-off between optimizing these two objectives. The same can be seen in the top row on the right with respect to $I(Z;S)$ and $I(Z;D)$. The bottom row of Figure 7 zooms in on the high performing models (see axes limits) and indicates the rank correlations. As stated above, we find a positive correlation between $I(Z;X)$ and $I(Z;S)$ (i.e. no conflict), but a negative correlation between $I(Z;X)$ and $I(Z;D)$, and between $I(Z;S)$ and $I(Z;D)$ (i.e. a trade-off).

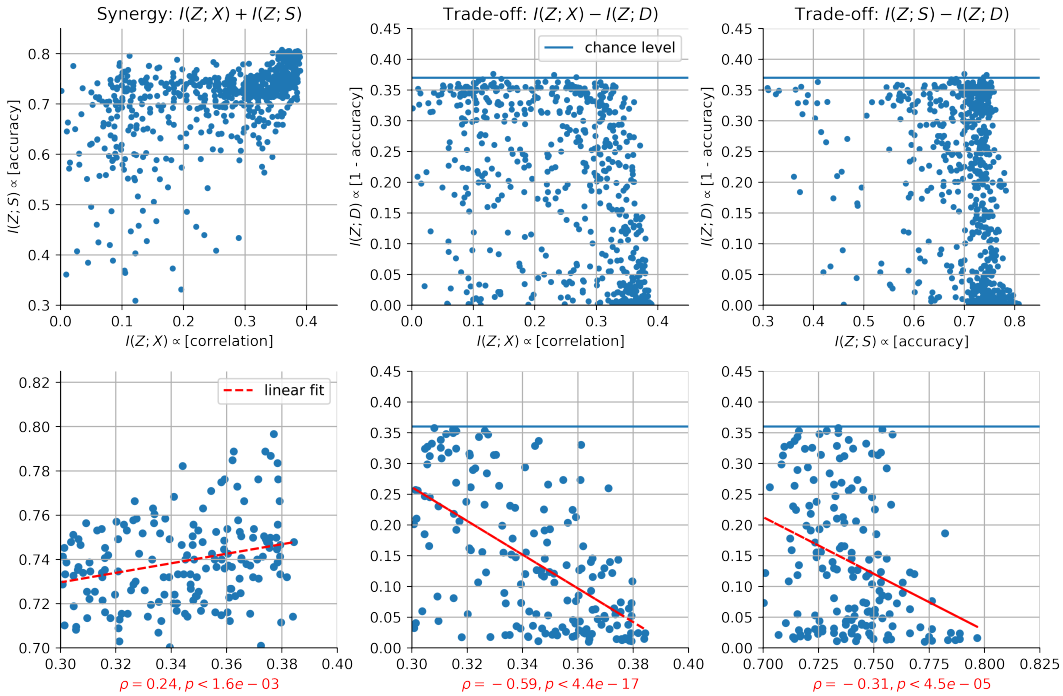

Figure 7: **Random Search Result**. Optimizing models with different hyperparameters shows how the terms in the objective function interact. The top row shows all models, the bottom row only filtered (high-performing) models within the indicated axes ranges. The red lines in the bottom plot indicate linear fits and the red axis labels show the rank correlation coefficients ρ and p values.

D Details for Comparison Models

D.1 Linear Model

Let our full dataset $x \in \mathbb{R}^{(N+M)\times D}$ consists of the concatenated datasets $x_A \in \mathbb{R}^{N\times D}$ and $x_B \in \mathbb{R}^{M\times D}$, i.e. $x = (x_A, x_B)^T$ For the linear model, we chose a design matrix $\beta \in \mathbb{R}^{(N+M)\times 2}$ of the form

$$\beta = \begin{bmatrix} 1 & -\frac{1}{N} \\ \vdots & \vdots \\ 1 & -\frac{1}{N} \\ 1 & \frac{1}{M} \\ \vdots & \vdots \\ 1 & \frac{1}{M} \end{bmatrix} \tag{5}$$

where the first column gives the constant component and the second column (the first N entries equal to $-\frac{1}{N}$ and the second M entries equal to $\frac{1}{M}$) encodes a contrast for the difference between the datasets. The matrix is orthogonal, thus avoiding a singular design. To produce a version of the data with domain effects removed, we fit this to the data with least squares $\gamma = \min_\gamma \|A\gamma - X\|_2^2$, $\gamma \in \mathbb{R}^{2\times D}$ and project out the second component like

$$\hat{x}_{Linear} = x - x_{(:,2)}\gamma_{(2,:)} \tag{6}$$

to obtain the linearly domain-corrected data.

D.2 Harmony

For *Harmony*, we used *Harmonypy* (version 0.05) (https://github.com/slowkow/harmonypy), which is the adapted *Harmony* [25] version for the Python environment. As input, we provided a PCA

embedding of the raw data (preprocessed). Here, we used the same number of principle components (PCs) as used for RAVE. Since *Harmonypy* returns corrected PCs, we performed further evaluation on these PCs (cf. Appendix Section A.1). To find the best model(s), we performed a random search over hyperparameters. We chose the best model with Acc_{dom} close to or at chance level, while having high Acc_{type} on predicted cell type labels. Furthermore, we used the exact same dataset splits as we did for RAVE and RAVE+.

D.3 scGen

We used *scGen* [26] (version 2.0.0) within the *Scanpy* [55] (version 1.7.2) working environment. As input to *scGen*, we used the raw responses with dataset source information (either dataset *A* or *B*) using the AnnData [55] object format (version 0.7.6). To run *scGen*, we used the following functions as described in the documentation (https://scgen.readthedocs.io/en/latest/tutorials/scgen_batch_removal.html): $setup\_anndata$ to setup the AnnData object for *scGen*, $SCGEN$ to setup the model, $train$ to train the model and $batch\_removal$ to remove inter-experimental variability.

As *scGen* returns corrected input data, we performed PCA on the output data, which were used for further evaluation (cf. Appendix Section A.1). Here, we used the same number of principle components (PCs) as used for RAVE. To find the best model, we performed a random search over hyperparameters. Just like *Harmony*, we chose the best model that had Acc_{dom} close to or at chance level, while having high Acc_{type} on predicted cell type labels.

D.4 Results of Dataset-Mixing by Harmony and scGen

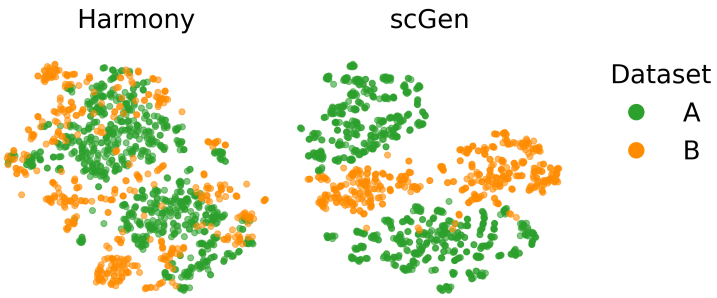

Figure 8: **Dataset Embeddings**. t-SNE embeddings of corrected data by *Harmony* (left) and *scGen* (right). Embedded cells are colored by dataset.

The low dimensional t-SNE embeddings [50, 56] (Figure 8), performed after the application of the two comparison methods (*Harmony* and *scGen*), show that cells from datasets *A* and *B* are not properly mixed; hence they are not removing inter-experimental variability sufficiently (see main paper, Table 1).

E Simulation experiments

In Figure 9, we present the results of the simulation experiments discussed in the main text. More specifically, we show example simulated cell responses for both stimuli (i.e., datasets 'A' and 'B') in Fig. 9A. Then in Fig. 9B, we demonstrate with a t-SNE embedding that the two datasets show clear inter-experimental variability. However, after correction with RAVE+, we can see in Fig. 9C that the two datasets have become aligned, and that the different cell types form clearly separated "islands". And lastly, in Fig. 9D, we see that the depth distributions of the RAVE+ corrected data are much better aligned with the ground-truth EM distributions than those of the raw data. This last steps further supports our validation procedure for RAVE+ on real data, based on EM IPL depth profiles.

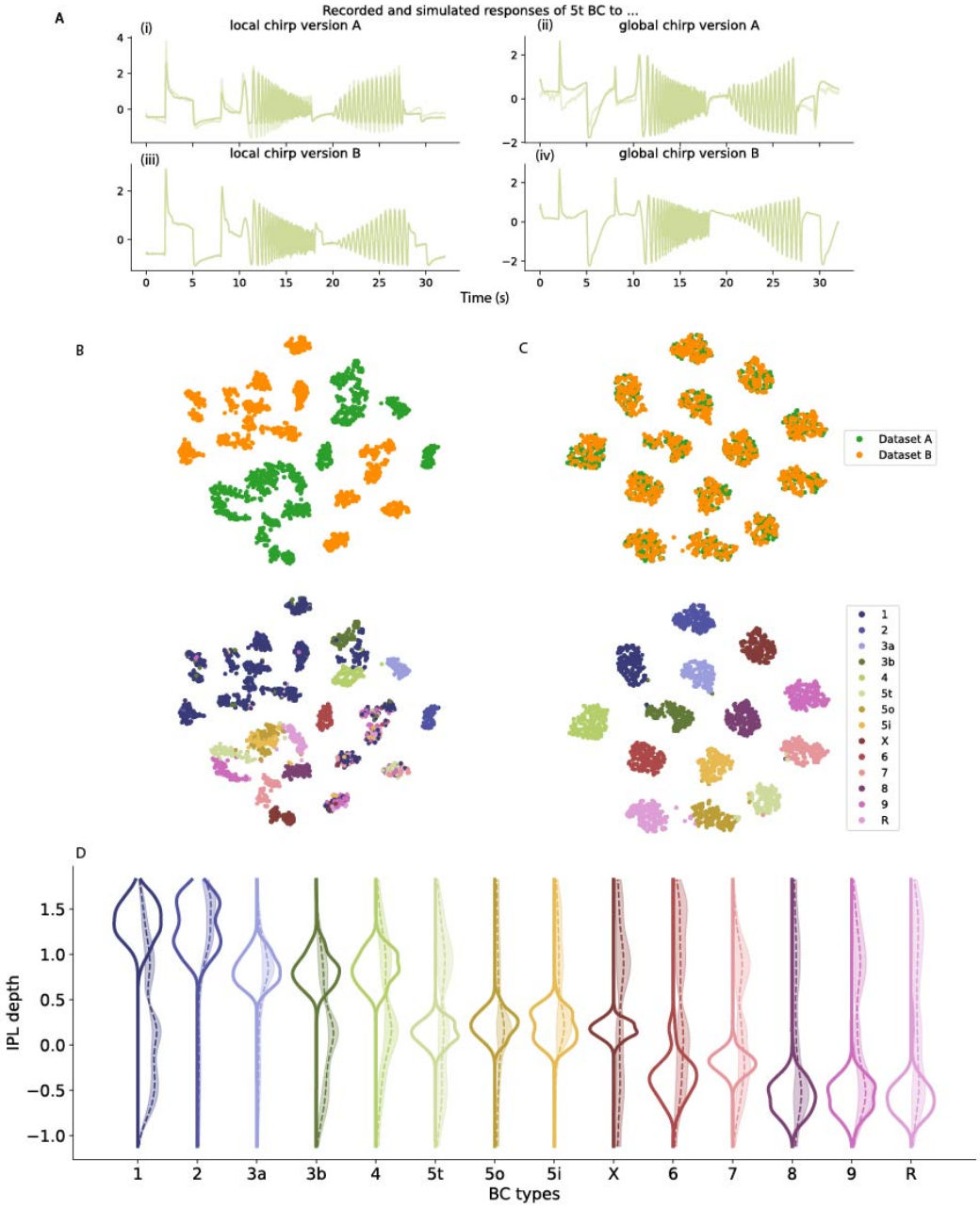

Figure 9: **RAVE+ results on simulated BC responses A**: Simulated (bold) and recorded (light) BC responses of example type 5t in response to (i) the local chirp version A (i.e. the stimulus played in dataset *A*); (ii) the global chirp version A; (iii) the local chirp version B (i.e. the stimulus played in dataset *B*); (iv) the global chirp version B. Note that for chirp version B, we do not have ground truth type labels for recorded responses. **B**: tSNE embedding of raw simulated test set data, colored according to dataset ground truth labels (top) and according to type labels predicted by a classifier trained on raw simulated responses of dataset *A* (bottom). The classifier fails for dataset *B*. **C**: same as **B**, but for `RAVE+` output. A classifier trained on `RAVE+` output for dataset A achieves accuracies of 1 for dataset *A* and 0.99 for dataset *B*. **D**: Distributions per cell type over IPL depth for EM data (distribution shown to the left), `RAVE+` output (solid line to the right) and raw data (dashed line to the right). Shaded area around the distributions shown to the right indicate SD across 10 seeds of the classifier. We sampled IPL depth values for the simulated data according to the type specific distributions known from EM data.

F Details for Performance Evaluation

F.1 Dataset-Mixing

To evaluate dataset-mixing, we used the `scikit-learn` [57] (version 0.24.1) implementation of the adjusted Rand Index (ARI) (cf. [28]).

F.2 Domain and Cell Type Classifier

In order to evaluate the model correction, we employ a domain and cell type classifier by using a random forest classifier (RFC) [58] from `scikit-learn` with cross-validated hyperparameters for each model. The RFC gets fitted on a subset of dataset *A* and validated on a held-out validation set. We performed the cross-validated grid search on the following hyperparameters: $n\_estimators$ (5, 10, 20, 30), max_depth (5, 10, 15, 20, `None`), ccp_alpha (0, 0.001, 0.01) and $max\_samples$ (0.5, 0.7, 0.9, 1). The grid search was performed using 10 random seeds to avoid overfitting (see main paper, section 4.3.1) and the best scoring RFC (highest Acc_{type}; lowest Acc_{dom} on validation set, respectively) was selected to predict cell types or domain labels on the test set of the corrected data.

F.3 Visualization of Dataset Embedding

We used the t-SNE algorithm [50] to visualize the cells in a low dimensional space [56]. For this purpose, we chose the `openTSNE` [59] implementation (version 0.6.0) in Python and ran it with default parameters and fixed seed.