Biological Cartography: Building and Benchmarking Representations of Life

Safiye Celik* **Jan-Christian Hütter†** **Sandra Melo Carlos†** **Nathan H Lazar***

Rahul Mohan† **Conor Tillinghast*** **Tommaso Biancalani†** **Marta Fay***

Berton A Earnshaw* **Imran S Haque***

Abstract

The continued scaling of genetic perturbation technologies combined with high-dimensional assays (microscopy and RNA-sequencing) has enabled genome-scale reverse-genetics experiments that go beyond single-endpoint measurements of growth or lethality. Datasets emerging from these experiments can be combined to construct "maps of biology", in which perturbation readouts are placed in unified, relatable embedding spaces to capture known biological relationships and discover new ones. Construction of maps involves many technical choices in both experimental and computational protocols, motivating the design of benchmark procedures by which to evaluate map quality in a systematic, unbiased manner.

In this work, we propose a framework for the steps involved in map building and demonstrate key classes of benchmarks to assess the quality of a map. We describe univariate benchmarks assessing perturbation quality and multivariate benchmarks assessing recovery of known biological relationships from large-scale public data sources. We demonstrate the application and interpretation of these benchmarks through example maps of scRNA-seq and phenomic imaging data.

1 Introduction

Advances in genomic technologies and high-throughput screening capabilities have enabled building maps of biology through unbiased, large-scale profiling of genetic perturbations. These maps have massive potential to uncover novel biology and accelerate drug discovery processes. Recent work [1, 2] has used CRISPR interference (CRISPRi) or CRISPR-mediated gene knockouts to build genome-wide perturbation maps using single cell RNA-seq [1] or cellular imaging [2] as readouts. Here, we propose a systematic framework for constructing and evaluating such maps by suggesting a shared vocabulary and benchmarking criteria, which we expect will lead to more comparable analyses of future maps of biology. We study the two cases above as examples, and report metrics for several design choices. Note that in this work we do not identify the parameters of the best map, nor do we survey all choices one may make in the course of building a map of biology.

2 Map building pipeline

Throughout this paper, we call the smallest experimental entity that is measured in a map context a "perturbation unit". This can be a single cell (e.g., Perturb-seq [1, 3]) or a well with hundreds of cells

*Recursion, first.lastname@recursion.com

†Genentech, huetter.janchristian-klaus/melo-carlos.sandra/mohan.rahul/biancalani.tommaso@gene.com

LMRL (Learning Meaningful Representations of Life) Workshop at the 36th Conference on Neural Information Processing Systems (NeurIPS 2022).

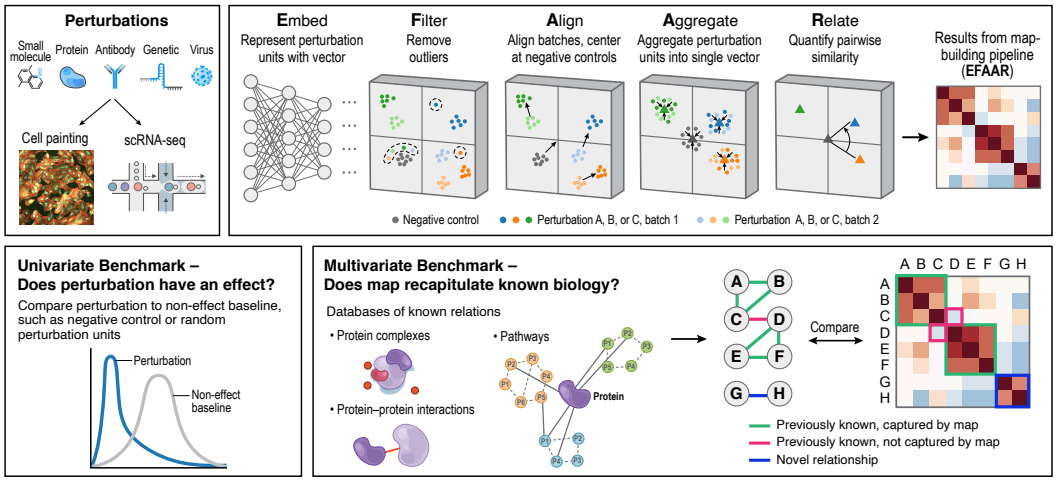

Figure 1: Graphical abstract of biological cartography: map building and benchmarking

in a certain experimental condition (e.g., phenomics/cell painting [4, 5]). Each unit is associated with assay output data, which may be structured or unstructured (e.g. transcript counts or a multi-channel cell image). Building a map (which relates perturbations to one another in a meaningful way) from these raw assay data, requires a number of post-experimental processing steps. We divide these transformations into five categories as below and name it the EFAAR pipeline. EFAAR steps may take place in a different order, multiple times (e.g., perturbation units may be filtered pre- and post-embedding), or potentially in a single end-to-end process.

- **E**mbedding assay data from each perturbation unit to generate a vector representation
- **F**iltering perturbation units that do not pass quality criteria
- **A**ligning different batches of perturbation units
- **A**ggregating units representing each perturbation (e.g., a gene)
- **R**elating different perturbations to each other (e.g., identifying gene relationships)

Embedding perturbation units This step is aimed at creating a vector representation of the experimental screening results. Intermediate layers of neural networks are commonly used to generate embeddings for unstructured data (e.g., cell images). Linear dimensionality reduction methods like k-means or principal component analysis (PCA) are common for structured data such as transcriptomic profiles, however non-linear dimensionality reduction techniques based on neural networks have been found to be effective as well [6, 7].

Filtering perturbation units or perturbations In any experimental screening process, some perturbation units will not satisfy pre-defined quality criteria and need to be filtered out. This filtering can occur before or after embeddings are generated, or before the relationships are generated. Examples include wells with too high or too low pixel intensity in a cellular imaging screen or perturbation units that are not distinguishable from the controls in terms of their readout or embeddings.

Aligning batches A *batch effect* is a systematic effect shared by all observations obtained under similar experimental conditions (e.g., microscopy acquisition artifacts, donor batch, incubation times) that potentially confound the interpretation of desired biological signal from the readouts. A baseline approach for aligning perturbation units is to use control units in each batch to center and scale features in each set. Another linear method aligning not only the first order statistics but also the covariance structures is TVN (typical variation normalization) [8]. Non-linear methods based on nearest neighbor matching [9, 10] or on conditional variational autoencoders, have been particularly successful for the alignment of single cell transcriptomic data [6, 7, 11].

Aggregating perturbation units There are typically multiple technical or biological replicates representing each perturbation in a given map, e.g. the same perturbation may be applied to dozens of wells or hundreds of cells. Aggregation of these replicates is critical for a robust final representation of a perturbation. Coordinate-wise mean and median aggregation are commonly used. More advanced methods like the Tukey median [12] may reduce the impact of outliers on the final representation, while increasing computational complexity.

Relating perturbations Identifying relationships between biological entities (e.g., gene-gene interactions arising from protein complexes or signaling pathways) is an important use case for maps built based on genetic perturbations. Computing distances (e.g., Euclidean or cosine) between aggregated perturbation representations is commonly used as a proxy for relationships, where smaller distance means a stronger relationship. These distances, in turn, can also be used to visualize the global structure of perturbations through further dimensionality reduction techniques such as uniform manifold approximation (UMAP) [13] or minimum-distortion embedding (MDE) [14].

3 Map benchmarking pipeline

Benchmarking can be done to evaluate the ability of an EFAAR pipeline to recover signal on individual perturbations (utilizing the perturbation replicates after alignment) or on its ability to recover relationships (utilizing the relationships between aggregate representations). We call these univariate and multivariate benchmarks, respectively, and describe results on two orthogonal datasets. Replogle et al. [1] perturb approximately 10,000 expressed genes in K562 cells using CRISPRi and measure single-cell RNA-seq readout to generate a transcriptomic map while the Recursion data contain a proprietary collection of imaging data in which CRISPR knockout technology was used to target approximately 17,000 genes in primary HUVEC cells [2].

3.1 EFAAR pipeline choices

3.1.1 Transcriptomic maps

We downloaded pre-filtered single-cell gene expression for K562 cells from `gwps.wi.mit.edu`. We used either the top 100 principal components from PCA or 128 latent dimensions from scVI (single-cell variational inference) [7], a conditional variational auto-encoder providing both embedding and alignment. Below are the EFAAR steps specifying choices of the different pipelines we used to build the transcriptomic maps we benchmarked.

- **Align & Embed**: ***(Choice 1)*** Compute the mean and standard deviation of all non-targeting controls per batch and use those to z-score all cells in the same batch and apply PCA and retain top 100 principal components. ***(Choice 2)*** Obtain a vector representation through scVI using a network that has two hidden layers with 256 nodes and 128 latent dimensions.
- **Align**: Compute the mean over all non-targeting controls in the PCA space and subtract this mean vector from all cells.
- **Aggregate**: Compute the mean vector across cells for each perturbation.
- **Filter**: ***(Choice 1)*** Keep all genes. ***(Choice 2)*** Exclude genes without transcriptoprint.
- **Relate**: Compare perturbations using cosine similarity.

3.1.2 Phenomic maps

The pipeline starts with six-channel Recursion cell painting images of wells. We generated embeddings by extracting activation values from an intermediate layer of a weakly supervised convolutional neural network (CNN) and apply two post-embedding alignment methods: Centerscale (per-batch standardization) and TVN [8]. Below are the EFAAR steps specifying choices of the different pipelines we used to build the phenomic maps we benchmarked.

- **Embed & Align**: Pass images through a pre-trained CNN and store the activations from an intermediate layer to obtain a fixed-length vector representation of the image. This model was trained to be partially resilient to batch effects.

- **Filter**: Apply additional proprietary filters to remove outlier image embeddings.
- **Align**: ***(Choice 1)*** Batch-correct by center-scaling (z-scale) per batch using experimental controls included in each batch. ***(Choice 2)*** Apply TVN [8] using experimental controls from all batches.
- **Aggregate**: Compute the mean vector over each perturbation.
- **Filter**: ***(Choice 1)*** Keep all genes. ***(Choice 2)*** Exclude genes without phenoprint.
- **Relate**: Compare perturbations using cosine similarity.

3.2 Univariate benchmarks

Univariate benchmarks assess the reproducibility and robustness of the representations of individual perturbations in a map. We demonstrate two such metrics: (1) consistency of the perturbation profile across replicates quantified with the average cosine similarity between replicates, and (2) magnitude of the perturbation effect quantified with energy distance [15, 16] as in [1]. For both of these metrics, we provide the result of statistical significance tests, more details on which can be found in Appendices A.1 and A.2. We call the representations of perturbations that pass a certain significance threshold from the associated statistical tests "phenoprints" in phenomic maps, "transcriptoprints" in transcriptomic maps, or "perturbation prints" to cover both cases. Rates of perturbation print identification can be compared between different map processing pipelines (EFAAR parameter choices) and stratified by global annotations like gene expression or functional gene groups.

We measured the perturbation print rates with above univariate metrics in Replogle et al. [1] data and Recursion data [2]. For Replogle et al. [1] data, scVI-based EFAAR pipeline outperformed PCA-based one in terms of transcriptoprint rate with either metric (see Table 1), identifying slightly more genes. 38% of all targeted genes were identified as significant across both methods and metrics (see Figure B.1), while 29% of perturbed genes were not detected by any tested condition.

For Recursion data, we report results *relative to* the output of the first step in Section 3.1.2 above, which we call CNN-BC (convolutional neural network with batch correction). While TVN leads to a large improvement over CNN-BC in both consistency and distance, Centerscale result is only slightly better than CNN-BC (see Table 1). We hypothesized that this might be because of the batch effect correction component of CNN-BC. To test this hypothesis, we assessed, as our baseline, a different embedding model lacking the batch effect resiliency component, which we call CNN-noBC. We saw that applying Centerscale on top of CNN-noBC improved performance by 583% in consistency and by 493% in distance (see Table B.1). An important conclusion of this comparison is that different steps in an EFAAR pipeline may interact with each other in non-obvious ways; e.g., the optimal alignment strategy may differ between different choices of embedding steps. For the rest of the paper, we use CNN-BC as our baseline for the phenomic map results, as in Table 1.

Table 1: Perturbation print rates based on univariate metrics: consistency and distance.

	Transcriptomic data [1]		Phenomic data [2]	
	PCA	scVI	Centerscale	TVN
Consistency	52%	61%	-1.9%	+100%
Distance	51%	53%	+5.8%	+109.1%

3.3 Multivariate benchmarks

A typical use case for a map of biology is to discover novel, biologically-relevant relationships between genes or between a gene and a small molecule (e.g., a drug candidate). In this work, we focus on the relationships among genes since Replogle et al. [1] data only contain gene perturbations.

There are two main types of gene-based benchmark sources: pairwise relationships and gene clusters. Sources of the first type include pairs that directly interact in a signaling pathway or a small protein interaction network. Sources of the second type represent all genes involved in a pathway, biological process, or protein complex and provide higher-level information for biological processes or pathways.

Here we look at both pairwise relationship recapitulation and cluster identification results. An important EFAAR choice before the Relate step is whether or not to remove perturbations that do not have a perturbation print. As mentioned in Section 3.1, we explore both options.

For pairwise relationships, we consider two publicly-available sources: **Reactome** [17] protein-protein interactions from protein complexes with at most four proteins, and Signaling Network Open Resource (**SIGNOR**) [18] pathway interactions. For cluster identification metrics we use three publicly-available sources: **Reactome** (gene sets from MSigDB C2 collection) [17, 19], **SIGNOR** [18] pathways, and COmprehensive ResoUrce of Mammalian (**CORUM**) [20] protein complexes.

For both pairwise and cluster metrics, we report the recall of annotated pairs within the most extreme 10% of pairwise relationships (we consider 5% from both tails of the pairwise distance distribution since negative relationships can indicate a negative signaling between genes). For cluster metrics we calculate a recall value per cluster and then average the per-cluster values to get the final metric, as described in Appendix A.3. Recall results on Replogle et al. [1] data for different alignment and filtering choices can be found in Table 2, showing a slight advantage of using scVI for alignment over PCA. Known relationship counts for different comparisons in Table 2 can be found in Table B.2.

In Figure B.2, we look at how the two maps generated using scVI vs PCA compare in terms of the recall value per cluster in the CORUM dataset. Consistent with the summary metrics in Table 2, scVI performs better for most of the clusters. Figure B.3 shows the distribution of the recall values across clusters for different cluster sources and EFAAR choices.

Table 2: Multivariate metrics in Replogle et al. [1] data.

	All genes		Genes w/ transcriptoprint	
	PCA	scVI	PCA	scVI
Reactome pairs	18.6%	19.8%	24.4%	25.4%
SIGNOR pairs	12.4%	12.9%	13.2%	15.2%
CORUM clusters	30.2%	33%	38.9%	40.9%
Reactome clusters	16.1%	16.9%	18.6%	19.6%
SIGNOR clusters	11.4%	13.3%	11.9%	13%

For the Recursion phenomic data, we again report recall results *relative to* CNN-BC (a CNN model with a batch correction component) as our baseline. We see that an alignment step by TVN or Centerscale leads to a considerable increase in a majority of metrics compared to the baseline (see Table 3), and TVN typically performs better than Centerscale as it did for the univariate benchmarks in Table 1. Known relationship counts for different comparisons in Table 3 can be found in Table B.3.

Table 3: Multivariate metrics in Recursion phenomic data [2].

	All genes		Genes w/ phenoprint	
	Centerscale	TVN	Centerscale	TVN
Reactome pairs	+14.5%	+82.1%	+6.5%	+15%
SIGNOR pairs	-4.9%	+55.1%	+3.5%	+26.3%
CORUM clusters	+18.2%	+107.1%	+10.2%	+18.8%
Reactome clusters	+6.5%	+62.9%	+11%	+8.7%
SIGNOR clusters	-11.5%	+52.9%	+14.1%	+142.7%

As an example of the known biology identified by the benchmarked EFAAR pipeline choices, Figure B.4 examines the cosine similarity structure for the Integrator complex which was also explored in Replogle et al. [1]. We see that both of the scVI-based and PCA-based EFAAR pipelines we tested on Replogle et al. [1] data and both of the TVN-based and Centerscale-based EFAAR pipelines on Recursion data accurately identify the modular structure of the Integrator complex.

4 Conclusion

In this work we describe a framework for systematically constructing whole-genome maps of biology and benchmarking their performance globally with publicly-available gene annotation datasets. As a demonstration, we present several map options built using two orthogonal data types: single-cell transcriptomic data with treatment with CRISPR interference (Perturb-Seq) and array-based phenotypic screening with CRISPR knockout. Results demonstrate the impact of different processing pipelines and metric choices. This framework can be used for any large-scale biological map building and benchmarking effort regardless of data types and can be expanded to include settings where additional perturbation types (small molecules, proteins, antibodies, viruses, etc.) or assay variables (growth conditions, reagent timing, etc.) are assessed.

To our knowledge, the only previous related work is the *pycytominer* GitHub repository (`github.com/cytomining/pycytominer`) that conceptualizes a pipeline for the analysis of image data given CellProfiler or DeepProfiler features. However, this work only provides an API to perform different analysis steps. It does not provide case studies of building maps from different data types or comparisons of results for different pipeline choices. Moreover, the evaluation steps focus on individual perturbations and do not tackle how well the known relationships between perturbations are recapitulated, i.e., multivariate benchmarks here.

5 Acknowledgments

We would like to thank James Taylor and Renat Khaliullin for their help with developing the EFAAR pipeline and benchmarking methodologies. We also would like to thank Leslie Gaffney and Orit Rozenblatt-Rosen for their help with the graphical representation of the EFAAR pipeline.

References

[1] Joseph M Replogle, Reuben A Saunders, Angela N Pogson, Jeffrey A Hussmann, Alexander Lenail, Alina Guna, Lauren Mascibroda, Eric J Wagner, Karen Adelman, Gila Lithwick-Yanai, et al. Mapping information-rich genotype-phenotype landscapes with genome-scale perturb-seq. *Cell*, 2022.

[2] Aurora S Blucher, Safiye Celik, James D Jensen, James Taylor, Michael F Cuccarese, Jacob C Cooper, Jacob M Rinaldi, Carl Brooks, Michael A Statnick, Marta Fay, Nathan Lazar, Berton Earnshaw, and Imran S Haque. Poster: Mapping biology with a unified representation space for genomic and chemical perturbations to enable accelerated drug discovery. In *Learning Meaningful Representation of Life Workshop at NeurIPS*, 2021.

[3] Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P Fulco, Livnat Jerby-Arnon, Nemanja D Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, et al. Perturb-seq: dissecting molecular circuits with scalable single-cell rna profiling of pooled genetic screens. *cell*, 167(7):1853–1866, 2016.

[4] Mark-Anthony Bray, Shantanu Singh, Han Han, Chadwick T Davis, Blake Borgeson, Cathy Hartland, Maria Kost-Alimova, Sigrun M Gustafsdottir, Christopher C Gibson, and Anne E Carpenter. Cell painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes. *Nature protocols*, 11(9):1757–1774, 2016.

[5] Michael F Cuccarese, Berton A Earnshaw, Katie Heiser, Ben Fogelson, Chadwick T Davis, Peter F McLean, Hannah B Gordon, Kathleen-Rose Skelly, Fiona L Weathersby, Vlad Rodic, Ian K Quigley, Elissa D Pastuzyn, Brandon M Mendivil, Nathan H Lazar, Carl A Brooks, Joseph Carpenter, Brandon L Probst, Pamela Jacobson, Seth W Glazier, Jes Ford, James D Jensen, Nicholas D Campbell, Michael A Statnick, Adeline S Low, Kirk R Thomas, Anne E Carpenter, Sharath S Hegde, Ronald W Alfa, Mason L Victors, Imran S Haque, Yolanda T Chong, and Christopher C Gibson. Functional immune mapping with deep-learning enabled phenomics applied to immunomodulatory and covid-19 drug discovery. *bioRxiv*, 2020. doi: 10.1101/2020.08.02.233064.

[6] 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.

[7] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

[8] D Michael Ando, Cory Y McLean, and Marc Berndl. Improving phenotypic measurements in high-content imaging screens. *BioRxiv*, page 161422, 2017.

[9] Laleh Haghverdi, Aaron TL Lun, Michael D Morgan, and John C Marioni. Batch effects in single-cell rna-sequencing data are corrected by matching mutual nearest neighbors. *Nature biotechnology*, 36(5):421–427, 2018.

[10] Krzysztof Polański, Matthew D Young, Zhichao Miao, Kerstin B Meyer, Sarah A Teichmann, and Jong-Eun Park. Bbknn: fast batch alignment of single cell transcriptomes. *Bioinformatics*, 36(3):964–965, 2020.

[11] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

[12] John W Tukey. Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematicians, Vancouver, 1975*, volume 2, pages 523–531, 1975.

[13] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[14] Akshay Agrawal, Alnur Ali, Stephen Boyd, et al. Minimum-distortion embedding. *Foundations and Trends® in Machine Learning*, 14(3):211–378, 2021.

[15] Gabor J Szekely. Potential and kinetic energy in statistics. *Lecture Notes, Budapest Institute*, 1989.

[16] Maria L Rizzo and Gábor J Székely. Energy distance. *wiley interdisciplinary reviews: Computational statistics*, 8(1):27–38, 2016.

[17] Marc Gillespie, Bijay Jassal, Ralf Stephan, Marija Milacic, Karen Rothfels, Andrea Senff-Ribeiro, Johannes Griss, Cristoffer Sevilla, Lisa Matthews, Chuqiao Gong, et al. The reactome pathway knowledgebase 2022. *Nucleic acids research*, 50(D1):D687–D692, 2022.

[18] Luana Licata, Prisca Lo Surdo, Marta Iannuccelli, Alessandro Palma, Elisa Micarelli, Livia Perfetto, Daniele Peluso, Alberto Calderone, Luisa Castagnoli, and Gianni Cesareni. Signor 2.0, the signaling network open resource 2.0: 2019 update. *Nucleic acids research*, 48(D1): D504–D510, 2020.

[19] Arthur Liberzon, Chet Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P Mesirov, and Pablo Tamayo. The molecular signatures database hallmark gene set collection. *Cell systems*, 1 (6):417–425, 2015.

[20] Madalina Giurgiu, Julian Reinhard, Barbara Brauner, Irmtraud Dunger-Kaltenbach, Gisela Fobo, Goar Frishman, Corinna Montrone, and Andreas Ruepp. Corum: the comprehensive resource of mammalian protein complexes—2019. *Nucleic acids research*, 47(D1):D559–D563, 2019.

[21] Kevin Drew, John B Wallingford, and Edward M Marcotte. hu.map 2.0: integration of over 15,000 proteomic experiments builds a global compendium of human multiprotein assemblies. *Mol Syst Biol*, 17(5):e10016, 2021.

A Details on benchmark computations

A.1 Univariate benchmark: perturbation consistency

We introduce the following notation. For a genetic perturbation g, we assume access to a total number of n_g query perturbation units. For each perturbation unit $i = 1, \ldots, n_g$, we have an embedding vector $x_{g,i}$. Moreover, each perturbation unit is associated with a batch $b_{g,i} \in \{1, \ldots, B\}$. Let g_b denote all perturbation units of g in batch b, and let $|g_b| = n_{g,b}$. Thus, $g = \bigcup_{b=1}^{B} g_b$ and $|g| = n_g$.

As the test statistic, we use avgsim, defined as the mean of the cosine similarity between each perturbation unit's profile and the profiles of all other perturbation units for g (i.e., in all batches). Formally,

$$\mathrm{avgsim}_g = \frac{1}{n_g n_g} \sum_{i}^{n_g} \sum_{j}^{n_g} \frac{\langle x_{g,i}, x_{g,j} \rangle}{\|x_{g,i}\| \|x_{g,j}\|}. \tag{1}$$

Parametric tests are not preferred for univariate metrics because the underlying population of distances do not typically follow a well-defined probability distribution. Consequently, we assess statistical significance of a gene g's perturbation profile using a non-parametric test on K empirical null perturbation samples that are generated considering the batch distribution of the n_g cells to $b \in \{1, \ldots, B\}$. This is needed because there could be batch effects remaining even after batch correction. The kth null sample for g, denoted as g'_k, is generated as follows. From each batch b with $n_{g,b} > 0$, draw $n_{g,b}$ cells uniformly at random, denoted by $g'_{k,b}$. Thus, $g'_k = \bigcup_{b=1}^{B} g'_{k,b}$. We then compute $\mathrm{avgsim}_{g'_k}$ for $k = 1, \ldots, K$ ($K = 1000$) as above and assign a p-value to perturbation g by

$$p_g = \frac{\max\{\#\{\mathrm{avgsim}_{g'_k} \leq \mathrm{avgsim}_g\}, 1\}}{K}. \tag{2}$$

For the transcriptomic data, we used cells as our query perturbation units, and for the phenomic data, we used CRISPR guides as our query perturbation units. Replacing avgsim with a leave-one-out average cosine similarity (loosim) allows for better outlier handling, and this is what we did for the phenomic data. Below is how we calculate loosim in this case.

$$\mathrm{loosim}_g = \frac{1}{n_g} \sum_{i}^{n_g} \frac{\langle x_{g,i}, \bar{x}_{g,i'} \rangle}{\|x_{g,i}\| \|\bar{x}_{g,i'}\|}. \tag{3}$$

where $\bar{x}_{g,i'}$ represents the average representation over all but the ith unit:

$$\bar{x}_{g,i'} = \frac{1}{n_g - 1} \sum_{j \neq i} x_{g,j}. \tag{4}$$

A.2 Univariate benchmark: energy distance

The energy distance [15, 16] measures how distant the replicate units of a perturbation are from the controls, essentially measuring the effect size of the perturbation in a high-dimensional space. For each query perturbation, we compute the distance of the replicate perturbation units' distribution to the control units' distribution using tests derived from energy statistics. Assuming access to two sets of embeddings $x_1, \ldots, x_{n_1}$ (representing query perturbation units) and $y_1, \ldots, y_{n_2}$ (representing control units), the energy distance is defined as

$$\frac{2}{n_1 n_2} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \|x_i - y_j\| - \frac{1}{n_1^2} \sum_{i=1}^{n_1} \|x_i - x_j\| - \frac{1}{n_2^2} \|y_i - y_j\|. \tag{5}$$

This distance will be zero when the distributions are identical, and positive between non-identical distributions. The statistical significance is then assessed using a permutation test comparing the

distance of the query perturbation against a large number of null samples generated through shuffling the labels of the query perturbation and control units. We used 1000 null samples in this study and computed the p-value in a similar fashion to Eq. (2).

Similar to the perturbation consistency computation, here we used, as our perturbation units, cells for the transcriptomic data, and CRISPR guides for the phenomic data. For transcriptomic data, to construct the null distribution to compare against each perturbation, we randomly sub-sampled 5% of all perturbation units that received the non-targeting control in any of the batches containing the query perturbation. Subsampling was necessary to reduce computation time.

A.3 Multivariate benchmark: recall

To assess how well a map embedding recapitulates known biology, we calculated recall measures on known pairwise relationships and known clusters as follows.

For pairwise relationships, we calculated pairwise cosine similarities between the aggregated perturbation embeddings of all perturbed genes and selected the top 5% and bottom 5% as predicted links. We excluded self-links as the cosine similarity for these is one and biases the recall computation. We then calculated the recall as the proportion of the intersection of those predicted links with a known relationship based on sources Reactome or SIGNOR to the total number of interactions in the same source between the perturbed genes.

For cluster relationships, we stratified the above calculation by cluster for Reactome, SIGNOR, or CORUM clusters. That is, for each cluster, we generated all gene pairs excluding self-links and used this set as our ground truth known gene relationships for that cluster. Then, similar to the calculation above for pairwise relationships, we calculated recall at the top 5% and bottom 5% of the cosine similarity distribution of all possible pairs of perturbed genes. This type of cluster stratification allows us to identify which areas of biology can and which cannot be captured using the built map.

B Supplementary Tables and Figures

Table B.1: Phenoprint rates for the phenomic map when using CNN-noBC as the baseline.

	Centerscale	TVN
Consistency	+583%	+1292.7%
Distance	+493.3%	+1072.3%

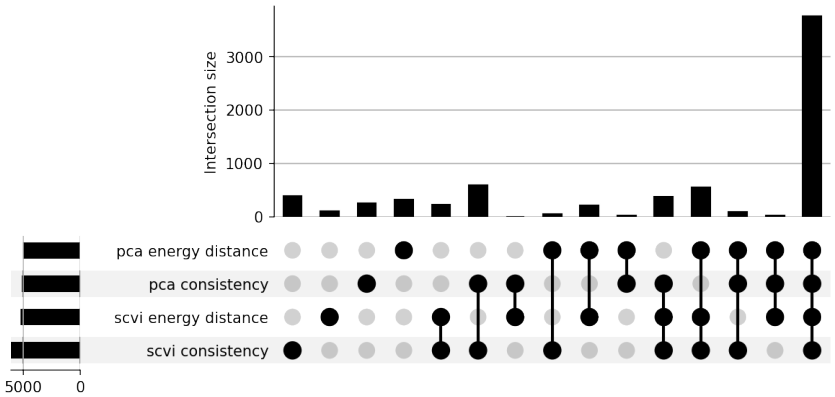

Figure B.1: UpSet plot of the intersection of transcriptoprints from two EFAAR embedding/alignment choices and two univariate benchmark metrics. Bar height reflects the number of genes with transcriptoprints (p-value < 0.01) in the group(s) represented by the solid circles below. Bar plot on the left shows the totals for each EFAAR choice and univariate metric.

Table B.2: Known relationship counts for multivariate benchmarks on Replogle et al. [1] data.

	All genes		Genes w/ transcriptoprint	
	PCA	scVI	PCA	scVI
Reactome pairs	10,379	10,379	3812	5120
SIGNOR pairs	5909	5909	2014	2751
CORUM clusters	33,175	33,175	19,934	24,122
Reactome clusters	2,609,505	2,609,505	896,961	1,166,333
SIGNOR clusters	5981	5981	1766	2413

Table B.3: Known relationship counts for multivariate benchmarks on Recursion phenomic data [2].

	All genes		Genes w/ phenoprint	
	Centerscale	TVN	Centerscale	TVN
Reactome pairs	18,486	18,486	3273	7209
SIGNOR pairs	10,484	10,484	1570	3858
CORUM clusters	27,890	27,890	15,227	23,050
Reactome clusters	5,022,964	5,022,964	592,737	1,528,163
SIGNOR clusters	11,312	11,312	1537	3610

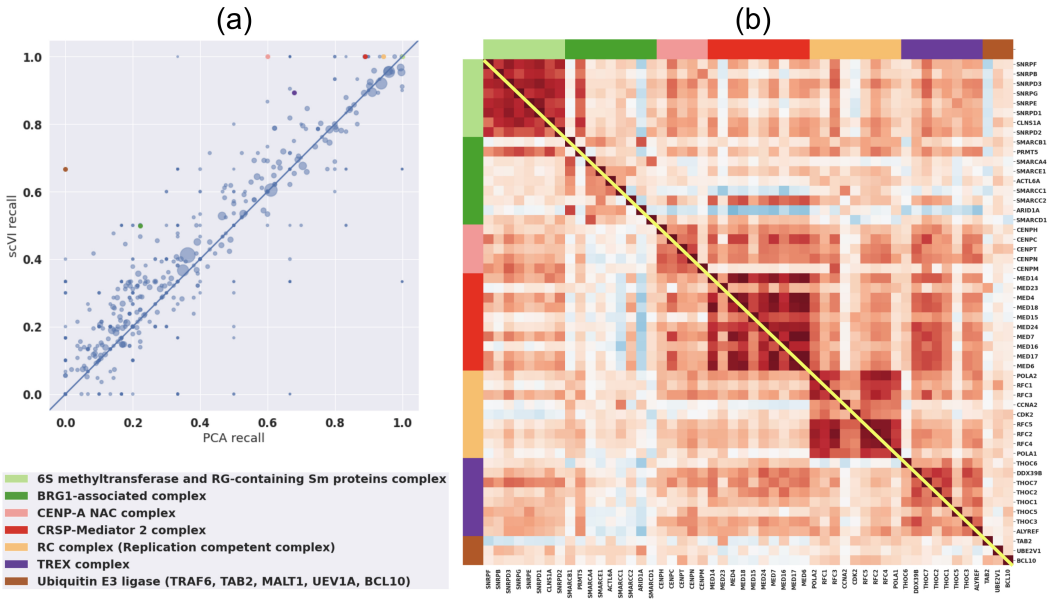

Figure B.2: (a) Scatter plot representing the recall value for each of the CORUM protein complexes from the scVI (y-axis) vs PCA (x-axis) transcriptomic maps. Each dot represents a complex, and the size of a dot represents the number gene subunits in the associated complex. (b) Cosine similarity heatmap for genes in seven of the CORUM complexes, where the scVI map is shown below the diagonal and PCA map is shown above the diagonal. Each color on the axes represents a different complex, as annotated in the legend. We look at all genes with no transcriptoprint filtering. Clusters are more visible on the scVI side of the heatmap, as consistent with the larger scVI recall for those clusters, as indicated by the dots with corresponding colors on the scatter plot in (a).

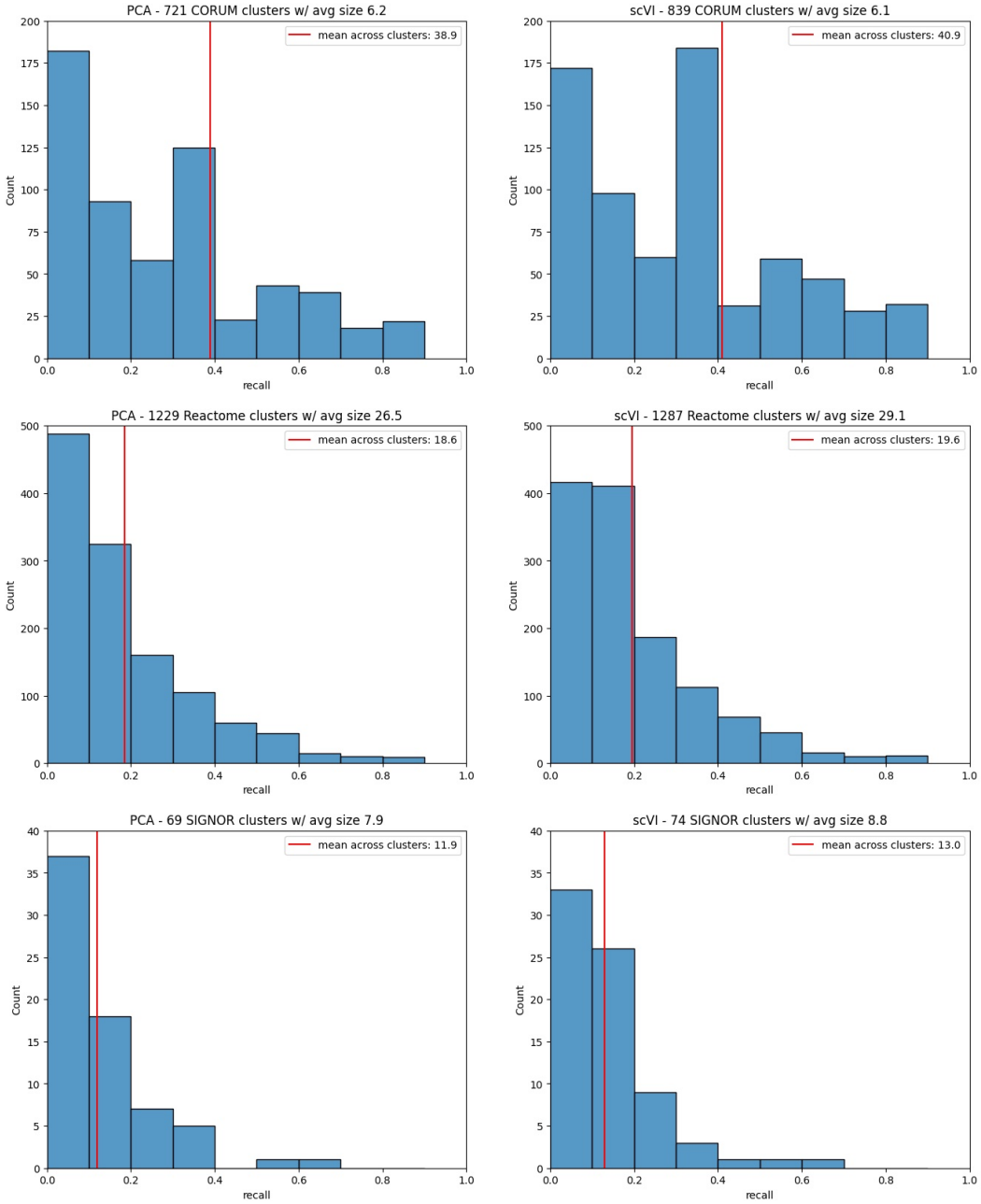

Figure B.3: Histograms representing the distribution of recall values across clusters for each benchmark source (rows) and embedding model (columns) when multivariate metrics are computed on Replogle et al. [1] data. Number of clusters and average cluster size are different for each cluster source, as indicated in the title for each plot. They are also different for PCA vs scVI for the same source since scVI leads to more genes with transcriptoprints and the recall values shown here are computed after filtering for such genes.

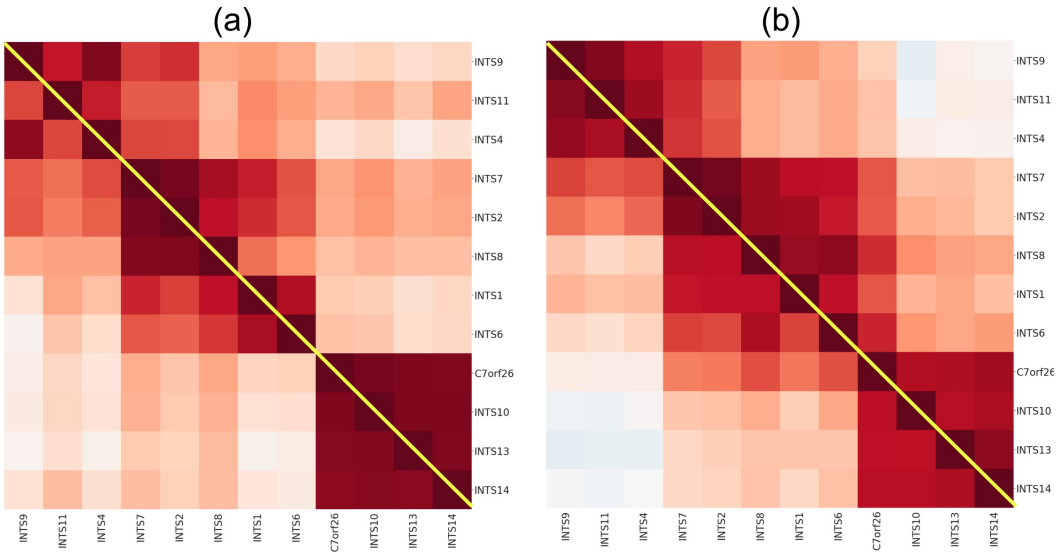

Figure B.4: Cosine similarity heatmaps of the Integrator complex subunits from (a) the two transcriptomic maps based on Replogle et al. [1] data and (b) the two Recursion phenomic maps. In (a), scVI map is shown below the diagonal and PCA map is shown above the diagonal, and in (b), TVN map is shown below the diagonal and Centerscale map is shown above the diagonal. We look at all genes with no perturbation print filtering. There are three main clusters visible in each of the four maps, which correspond to the three main modules of the integrator complex: endonuclease module including INTS4, INTS9, and INTS11 (top cluster); structural shoulder and backbone including INTS1, INTS2, INTS6, INTS7, and INTS8 (middle cluster), and enhancer module including INTS10, INTS13, and INTS14 (bottom cluster). C7orf26, which is clustered by each of the four maps as part of the enhancer module, was officially renamed INTS15 in January 2022 after it was suggested to be a subunit of the Integrator complex by Drew et al. [21] and Replogle et al. [1].