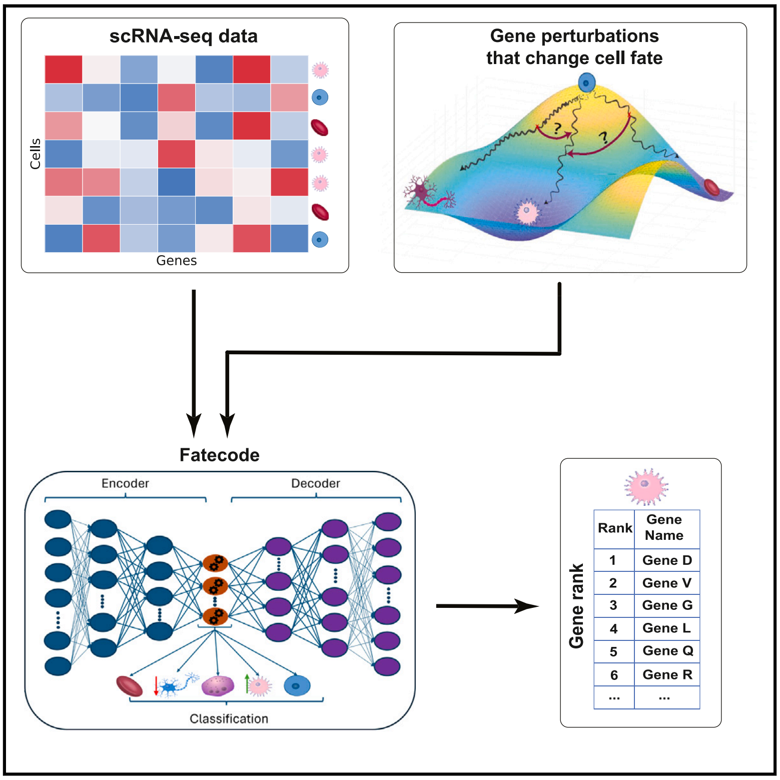
Cell Reports
Methods

Article

Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation

Graphical abstract



Rank	Gene Name
1	Gene D
2	Gene V
3	Gene G
4	Gene L
5	Gene Q
6	Gene R
...	...

Highlights

- Fatecode is a computational method to predict cell fate regulators from scRNA-seq data
- Fatecode uses autoencoder perturbation to identify genes that influence cell populations
- Simulations and real scRNA-seq data show Fatecode detects cell fate regulators
- Fatecode can accelerate discovery of cell fate regulators using widely available data

Authors

Mehrshad Sadria, Anita Layton,
Sidhartha Goyal, Gary D. Bader

Correspondence

msadria@uwaterloo.ca

In brief

Identifying the genes that control cell fate is essential for designing cellular reprogramming strategies. To provide an accessible, *in silico* complement to high throughput perturbation screens, Sadria et al. develop Fatecode, a deep learning-based computational method that can predict cell fate regulators from scRNA-seq data.

Sadria et al., 2024, Cell Reports Methods *4*, 100819
July 15, 2024 © 2024 The Author(s). Published by Elsevier Inc.
https://doi.org/10.1016/j.crmeth.2024.100819

CellPress

Cell Reports Methods

CellPress
OPEN ACCESS

Article

Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation

Mehrshad Sadria,[1,12,*] Anita Layton,[1,2,3,4] Sidhartha Goyal,[5] and Gary D. Bader[6,7,8,9,10,11]

[1]Department of Applied Mathematics, University of Waterloo, Waterloo, ON, Canada
[2]Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada
[3]Department of Biology, University of Waterloo, Waterloo, ON, Canada
[4]School of Pharmacy, University of Waterloo, Waterloo, ON, Canada
[5]Department of Physics, University of Toronto, Toronto, ON, Canada
[6]Department of Molecular Genetics, University of Toronto, Toronto, ON, Canada
[7]The Donnelly Centre, University of Toronto, Toronto, ON, Canada
[8]Department of Computer Science, University of Toronto, Toronto, ON, Canada
[9]The Lunenfeld-Tanenbaum Research Institute, Sinai Health System, Toronto, ON, Canada
[10]Princess Margaret Cancer Centre, University Health Network, Toronto, ON, Canada
[11]Canadian Institute for Advanced Research (CIFAR), Toronto, ON, Canada
[12]Lead contact
*Correspondence: msadria@uwaterloo.ca

https://doi.org/10.1016/j.crmeth.2024.100819

MOTIVATION How stem and progenitor cells decide which cell types they will generate via cell division is crucial for understanding tissue development and engineering cell types for use in regenerative medicine or cancer therapies. However, the identification of the regulators of these cell fate decisions within the complex and dynamic system of tissues is a major challenge. Experimental high-throughput perturbation screens can help to dissect regulators, but these are not practical or easy to implement in every context of interest. To address this challenge, we developed a computational method, Fatecode, to predict master regulators and key pathways controlling cell fate based on any single-cell transcriptomics dataset.

SUMMARY

Cell reprogramming, which guides the conversion between cell states, is a promising technology for tissue repair and regeneration, with the ultimate goal of accelerating recovery from diseases or injuries. To accomplish this, regulators must be identified and manipulated to control cell fate. We propose Fatecode, a computational method that predicts cell fate regulators based only on single-cell RNA sequencing (scRNA-seq) data. Fatecode learns a latent representation of the scRNA-seq data using a deep learning-based classification-supervised autoencoder and then performs *in silico* perturbation experiments on the latent representation to predict genes that, when perturbed, would alter the original cell type distribution to increase or decrease the population size of a cell type of interest. We assessed Fatecode's performance using simulations from a mechanistic gene-regulatory network model and scRNA-seq data mapping blood and brain development of different organisms. Our results suggest that Fatecode can detect known cell fate regulators from single-cell transcriptomics datasets.

INTRODUCTION

In tissue development, specific gene regulators control how cells change state and type to form a complete tissue.[1] These gene regulators are also important because they can be used to control cell fate for multiple applications, including in regenerative medicine and cancer.[2] However, it remains a challenge to identify these regulators within complex and dynamic tissue systems.[1]

Cell fate regulators can be identified using experimental methods such as high-throughput genetic perturbation screens (e.g., CRISPR-based) with single-cell gene expression (single-cell RNA sequencing [scRNA-seq]) readouts.[3,4] However, these methods are challenging to run on arbitrary biological systems.

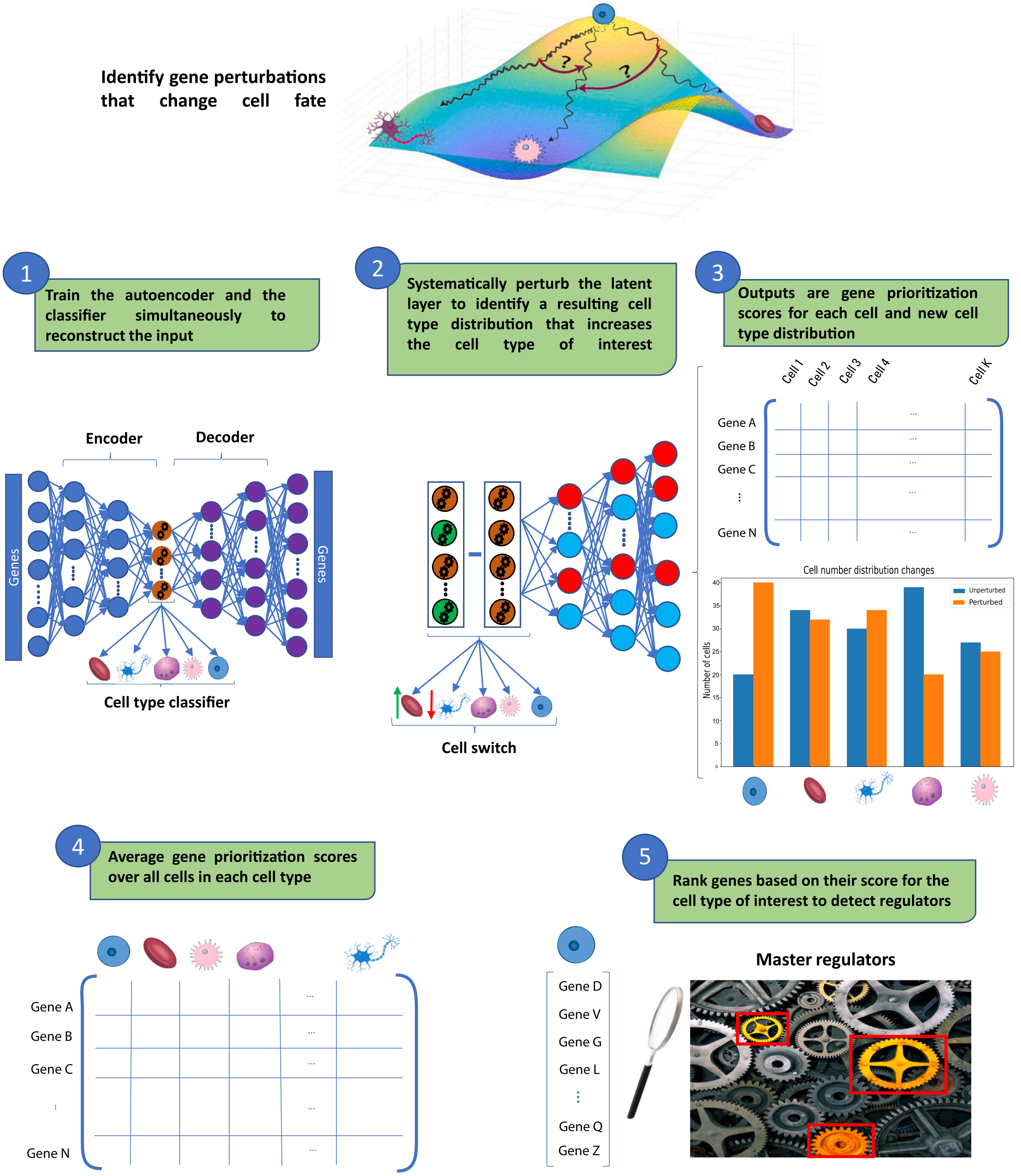

Identify gene perturbations that change cell fate
?
?
1
Train the autoencoder and the classifier simultaneously to reconstruct the input
Encoder
Decoder
Genes
Genes
Cell type classifier
2
Systematically perturb the latent layer to identify a resulting cell type distribution that increases the cell type of interest
Cell switch
3
Outputs are gene prioritization scores for each cell and new cell type distribution
Cell 1
Cell 2
Cell 3
Cell 4
Cell K
Gene A
Gene B
Gene C
Gene N
Cell number distribution changes
Number of cells
0
5
10
15
20
25
30
35
40
Unperturbed
Perturbed
4
Average gene prioritization scores over all cells in each cell type
Gene A
Gene B
Gene C
Gene N
5
Rank genes based on their score for the cell type of interest to detect regulators
Gene D
Gene V
Gene G
Gene L
Gene Q
Gene Z
Master regulators

Figure 1. Fatecode workflow for *in silico* perturbation experiments and cell fate regulator detection

The 3D model (top) represents a Waddington-like landscape depicting cellular reprogramming processes. We seek to identify genes (question marks) that regulate paths on this landscape (wavy lines) by transitioning them to another path (red arrows). A classification-supervised autoencoder learns a latent space representing the original data, optimized for both input reconstruction and cell type classification. The latent layer is systematically perturbed, and by investigating all resulting perturbation-generated cell type distributions, distributions with an increase or decrease in a cell type of interest are identified. Perturbation output is

(legend continued on next page)

Computational methods have been developed to predict gene expression programs that explain the difference between perturbed and unperturbed states[5–8] or to predict the linear effect of perturbing a particular transcription factor.[9] Also, computational methods that determine the ordering of cell states along a trajectory, based on their gene expression profiles using a pseudotime or actual time approach,[10–14] have been used to examine the cell decision-making process by identifying genes that are differentially expressed between trajectory branches. However, these latter methods often have trouble identifying accurate trajectories and branchpoints.[15,16] Furthermore, none of the above methods are designed to identify cell fate regulators in normal developmental processes.

We developed Fatecode, a computational method to predict important cell fate regulator genes for cell types of interest. Fatecode predicts cell fate regulators based only on scRNA-seq data covering a given range of cell types to be analyzed. Fatecode learns a latent representation of the scRNA-seq data using a deep learning-based classification-supervised autoencoder[17,18] and then performs *in silico* perturbation experiments on the latent representation to predict genes that, when perturbed, would alter the original cell type distribution to increase or decrease the population size of a cell type of interest. Fatecode can be thought of as an *in silico* CRISPR perturbation screen that identifies genes that influence cell fate, based on a cell type readout. These genes can be traditional (e.g., transcription factors) or non-traditional regulators (any other genes). We assessed Fatecode's performance using simulated data produced by a mechanistic model based on pre-defined gene-regulatory networks with known cell fate regulators[19] and tested it on scRNA-seq maps of blood and developing brain from zebrafish and mouse.[20–23]

RESULTS

Fatecode method overview

Fatecode uses a classification-supervised autoencoder to detect key genes that can shift the cell type frequencies in an input scRNA-seq dataset toward a desired distribution of cell types. Taking single-cell gene expression profiles as input, the autoencoder learns a latent space with reduced dimensions capturing the input information (reduce gene dimension x cell matrix). A supervised cell type classifier is included as part of the loss function to create a latent space composed of features that support optimal cell type classification in addition to input data reconstruction. Known cell type annotations in the input data are used to train the classifier. This ensures that the latent space is relevant for cell type classification used in later stages. Each latent layer node of the autoencoder, which represents a reduced dimension of the input, is systematically perturbed to simulate altering key gene expression programs (sets of genes that are correlated with each other that are represented by individual learned latent layer dimension). Cell types are then reclassified to characterize the effect of the perturbation, and the autoencoder's decoder uses the perturbed and unperturbed latent embeddings to generate a gene-by-cell matrix of gene prioritization scores. This matrix is used to identify genes important for the perturbation effect (STAR Methods; Figures 1 and S1). Resulting cell type distributions are generated for each possible perturbation and then manually evaluated to identify those that increase or decrease proportions of desired cell types. In this way, regulator genes are identified to increase or decrease a given cell type proportion relative to all other cell types, and these are predicted to be cell fate regulators for the given cell type. An average of the cell fate regulator prioritization scores across cells for the cell type is computed to produce a final regulator list for each cell type.

Our latent layer perturbation approach is inspired by latent vector operations used in natural language processing and computer vision applications to generate novel text and images.[24–26] In those applications, perturbation operations performed on the latent layer generally yield superior results compared to operations performed directly in the input space. The classification component of Fatecode is used to exclude possible latent space regions that do not conform to the overall structure of the data. This helps in learning a model that is more representative of the underlying data distribution.[27]

Optimizing model architecture and hyperparameters

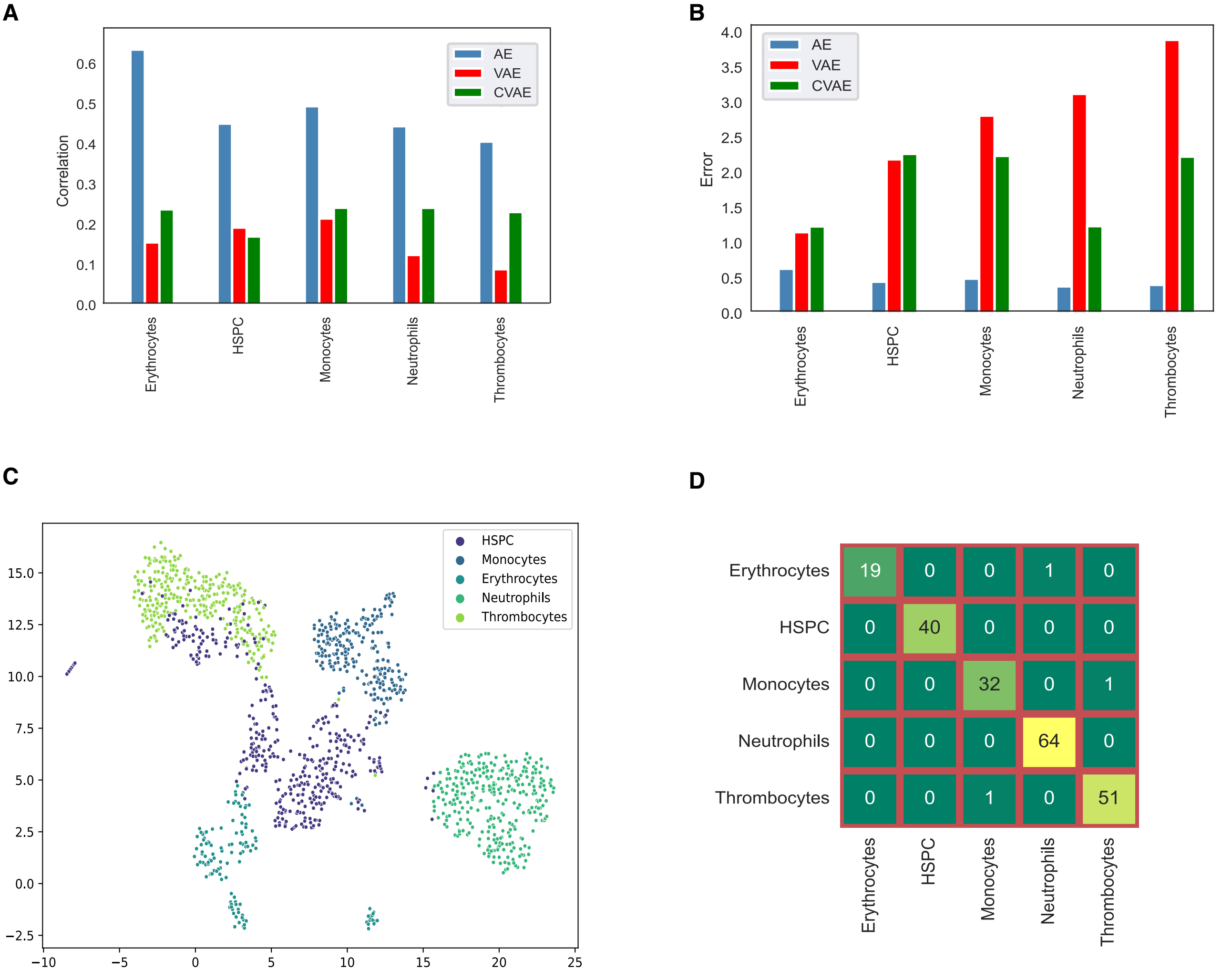
Fatecode relies on the latent embedding of an autoencoder, but different types of autoencoders may produce different results, depending on the input data (see supplemental information).[6,28–30] To investigate this in our problem context, we evaluated the performance of three common autoencoder architectures: under-complete autoencoder (AE), variational autoencoder (VAE), and conditional VAE (CVAE).[31] The first step of Fatecode evaluates these three autoencoder architectures and other hyperparameters (see the Hyperparameter search section) to find the ones that best reconstruct the input data, measured by mean squared error for reconstruction and cross-entropy for cell type classification. To illustrate the importance of this step, we compared how the choice of autoencoder affects learning the underlying representation for two single-cell gene expression datasets in adult zebrafish blood[20] and murine pancreatic development.[32] AE produced the lowest reconstruction error for the zebrafish data (averaged over cell types) (Figures 2A and 2B). AE also produced a latent layer that successfully reduces the dimension and cleanly separates the five known cell types in the data (Figure 2C), and its cell type classifier yields a high accuracy (Figure 2D). However, for the mouse data, VAE achieved a higher accuracy compared to the other autoencoders (Figure S2).

Fatecode accurately detects known regulators from simulated scRNA-seq data

To assess the accuracy by which Fatecode identifies cell fate regulators using gene expression profiles, we applied the

simulated by subtracting the perturbed from unperturbed latent layers and feeding it to the decoder to identify a cell-by-gene matrix of prioritization scores that can help us to prioritize genes predicted to be important for achieving a desired cell population distribution. An average of the cell fate regulator prioritization scores across cells in each cell type is computed. By sorting these genes based on their prioritization scores for a cell type of interest, the model predicts genes that are important for regulating the levels of a given cell type.

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819



	Erythrocytes	HSPC	Monocytes	Neutrophils	Thrombocytes
Erythrocytes	19	0	0	1	0
HSPC	0	40	0	0	0
Monocytes	0	0	32	0	1
Neutrophils	0	0	0	64	0
Thrombocytes	0	0	1	0	51

Figure 2. Comparison of autoencoder architectures for analyzing data for hematopoiesis regulation in zebrafish blood

(A) Comparison of correlation between input and output of AE, variational autoencoder (VAE), and conditional variational autoencoder (CVAE).

(B) MSE between input and output of the three autoencoder architectures, showing that AE produces the lowest error rate for this dataset.

(C) Uniform Manifold Approximation and Projection (UMAP) visualization of the latent layer of the under-complete autoencoder (AE).

(D) Confusion matrix for the classifier connected to the latent layer of AE demonstrating excellent classification performance.

method to simulated scRNA-seq data generated from known gene regulatory network (GRN) structures using SERGIO.[19] SERGIO allows users to specify the number of cell types and key regulators in the simulated GRN (Figure 3A). While Fatecode is not specific to GRNs (i.e., it can identify a list of genes of any type, not just transcription factors), a GRN-based simulation is expected to provide a good benchmark for our method. A matrix of 400 cells and 2,700 genes with 20 known regulators and 9 cell types was generated and run through Fatecode. Predicted cell fate regulator genes and their prioritization scores were compared to the known SERGIO regulator list. The number of known regulator genes identified increases as more genes are prioritized (Figure 3B). Almost all of the known regulator genes (18 of 20) were identified when 150 genes were prioritized (of 2,700). To compare with a naive baseline, we identified cell type markers (top 20 genes) using differential gene expression (DGE) analysis on the same data using Seurat's non-parametric Wilcoxon rank-sum test.[33] Fatecode identifies a larger number of known regulators compared to DGE analysis when examining up to 150 prioritized genes (Figure 3B). As SERGIO is a stochastic method, we analyzed five additional simulated datasets of the same size, all of which yielded similar results (plotted as shading in Figure 3B). We repeated this analysis on a larger dataset consisting of 2,700 cells, 1,200 genes, with 65 predefined regulators, and 9 distinct cell types. We used Fatecode to identify the top 180 key genes of these data, and DGE analysis to identify the top 20 differentially expressed genes from each cell type. Also, for comparison, we included scFates, a method specifically designed for trajectory-based DGE analysis.[34] Fatecode consistently outperformed both DGE methods in detecting known regulators. We further evaluated performance by varying the top k gene threshold of DGE, and Fatecode consistently outperformed

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

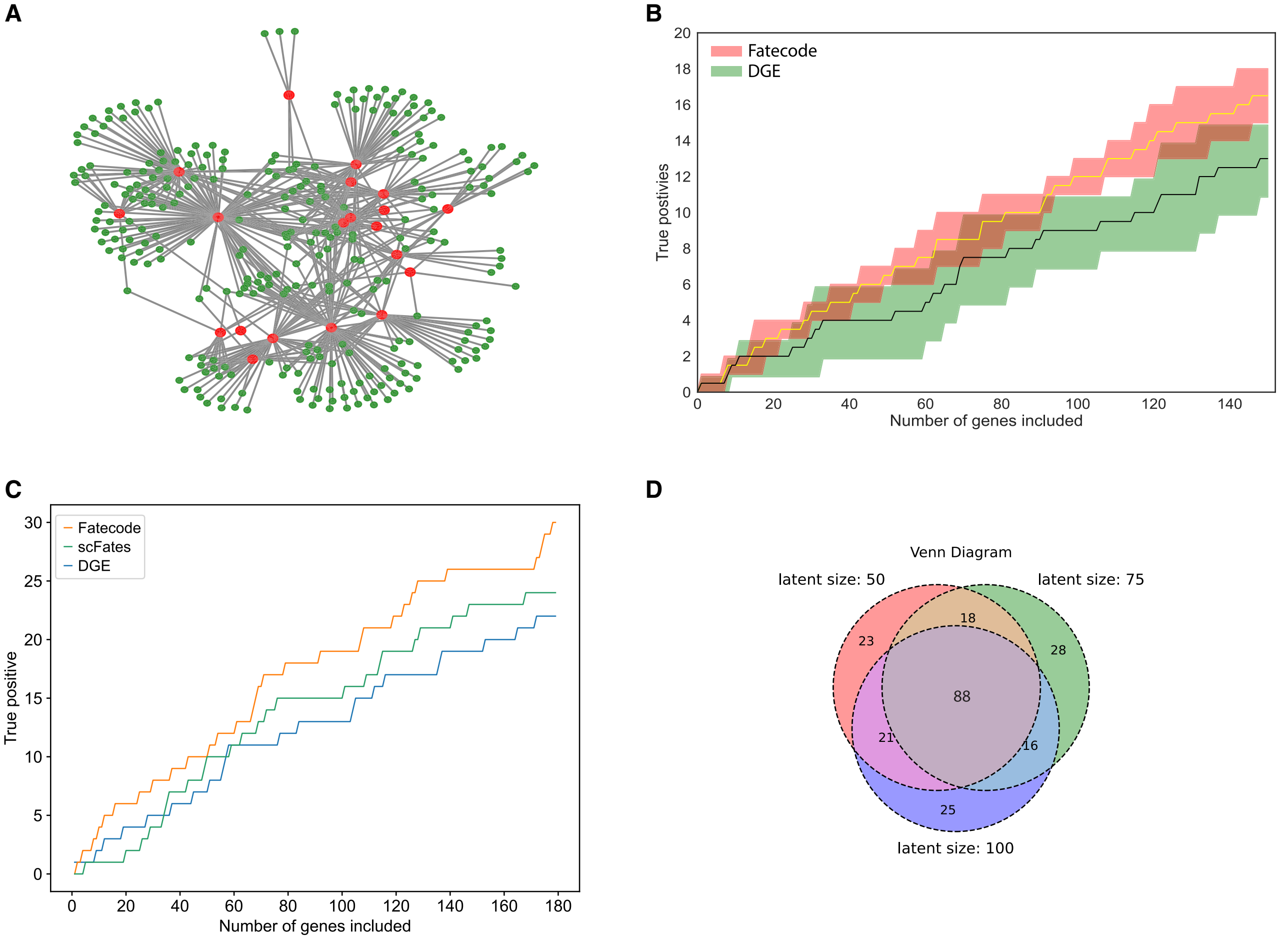


Figure 3. Fatecode detects known regulators using simulated data generated by SERGIO

(A) The schematic structure of the GRN to generate scRNA-seq. Red nodes are known regulators, and green nodes are non-regulators whose production rates are determined by their associated regulators. Our goal is to identify known regulators from the generated scRNA-seq data using Fatecode.

(B) Benchmark comparisons of the detection rate of predefined regulators generated by SERGIO using Fatecode compared with a naive differential gene expression (DGE) baseline. The red and green areas represent the performance of Fatecode and DGE, respectively, on the simulated data with 400 cells.

(C) Benchmark comparisons of the detection rate of known regulators using Fatecode, scFates, and DGE on simulated data with 2,700 cells.

(D) Venn diagram showing the similarity between the number of known regulators uncovered by Fatecode across various latent layer sizes.

DGE across all tested thresholds, demonstrating its robustness while varying the number of genes considered (Figures 3C and S3). Thus, Fatecode performs well at identifying known regulators in simulated scRNA-seq data.

We also examined the sensitivity of our model by the size of the latent layer in the autoencoder by training Fatecode with different latent layer sizes (n = 50, 75, and 100 dimensions) using the 2,700-cell simulated data (Figure 3D). Our results show general consistency across the different latent layer sizes, indicating that Fatecode exhibits robustness across a range of latent layer sizes.

Fatecode identifies known cell fate regulator genes in mouse hematopoiesis

Hematopoiesis is a cell differentiation process by which the body produces mature blood cells from stem cells. We applied Fatecode to a published mouse hematopoiesis single-cell differentiation dataset, which involves the differentiation of myeloid progenitors into 9 cell types (Figure 4A).[22] We then examined Fatecode's accuracy in predicting cell fate regulators that lead to the desired cell type distribution by comparing the results with ground-truth experimental perturbation data and known regulator genes.[9,22,35,36] Fatecode learned a latent node that, when perturbed, simultaneously increases the monocyte population and decreases erythrocytes and granulocytes (Figure 4B). Previous studies have demonstrated that *Irf8* is important in promoting the differentiation of the GM (granulocyte-monocyte) lineage, particularly monocytes, and functions as a key regulator in determining the fate between granulocytes and monocytes. Fatecode accurately predicted *Irf8* as an important cell fate regulator in the monocyte differentiation process. It correctly assigned a high positive score for monocytes and late_GMP (granulocyte-macrophage progenitor) and negative scores for granulocytes and MEP (megakaryocyte-erythroid progenitor)

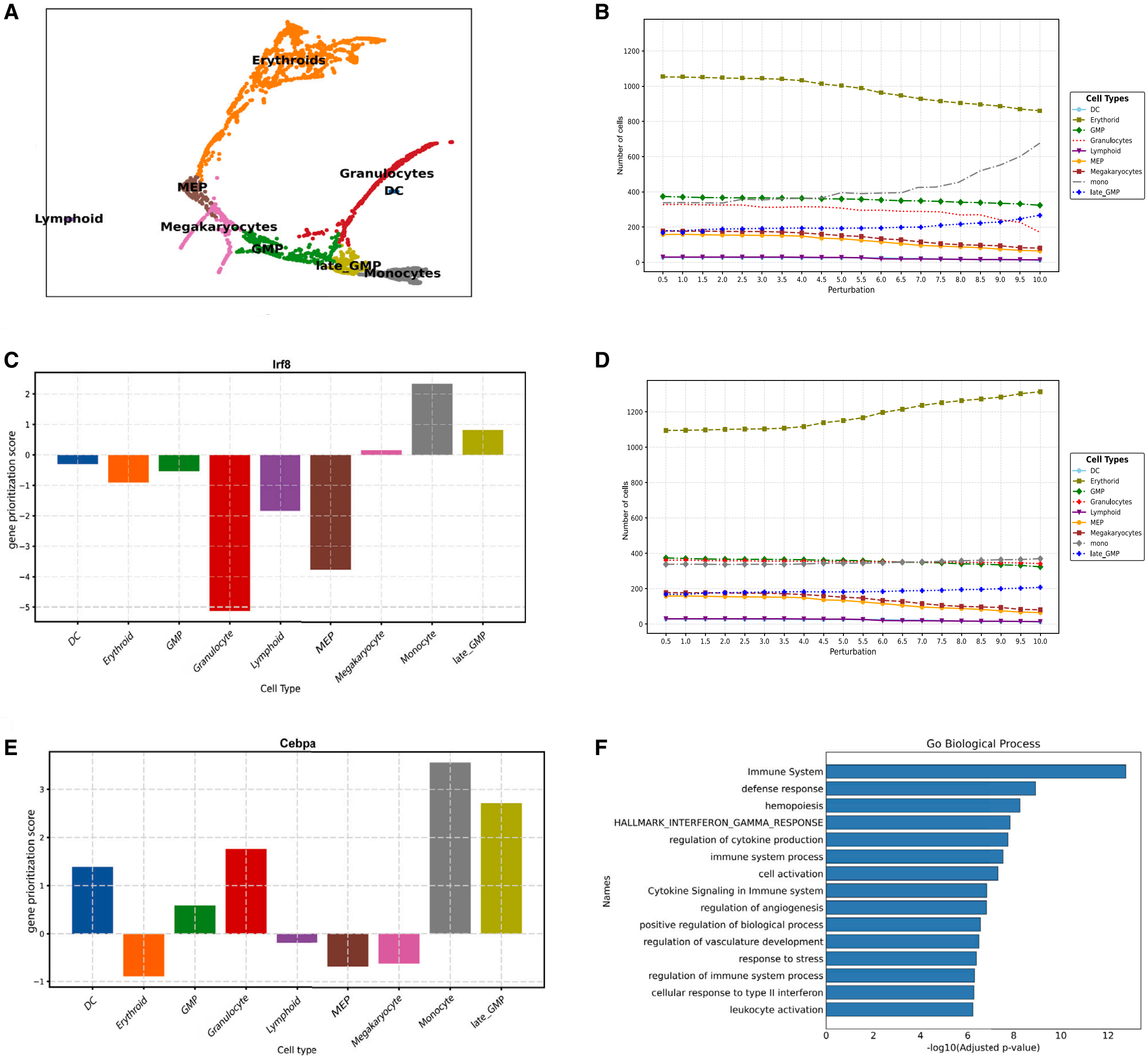


Figure 4. Fatecode accurately detects regulators and predicts the effect of single-cell perturbations

(A) Hematopoiesis data from Paul et al.,[22] visualized as a UMAP and clustered into 9 cell types.

(B and D) The results of *in silico* perturbations that change the initial cell frequency to the desired distribution. For (B), our objective was to promote monocytes while reducing the number of erythrocytes. For (D), we aimed for an increase in the erythroid population and a decline in MEPs and megakaryocytes.

(C and E) Gene prioritization scores per cell type for *Irf8* and *Cebpa*.

(F) Pathway enrichment analysis results. Gene Ontology biological processes show significant processes related to cell development and hematopoiesis.

lineages, consistent with previous studies (Figure 4C). Next, we investigated the prediction results for *Cebpa,* knockout of which leads to a decline in the population of differentiated myeloid cells, while concurrently increasing the number of erythrocytes. Fatecode accurately assigned a high positive score to *Cebpa* for monocytes and granulocytes and a negative score to erythrocytes and MEPs (Figures 4D and 4E). In another example, *Klf1* is a key regulator in driving differentiation toward the ME (megakaryocyte-erythroid) lineage, specifically promoting the development of erythrocytes, while simultaneously inhibiting the GMP lineage. Fatecode correctly assigned a set of positive scores to *Klf1* for erythrocytes and MEPs, indicating its ability to capture a key regulator in ME lineage differentiation (Figure S4A). We also tested Fatecode's ability to detect genes that are known to be important in maintaining stemness and inhibiting differentiation. Fatecode correctly predicted *Runx1* as a candidate that has negative scores for perturbations that increase all mature cell types (all cell types expect MEPs and GMPs) (Figure S4B). Last, we examined the prediction results for *Fli1*, which has diverse effects on differentiation. Fatecode accurately gives

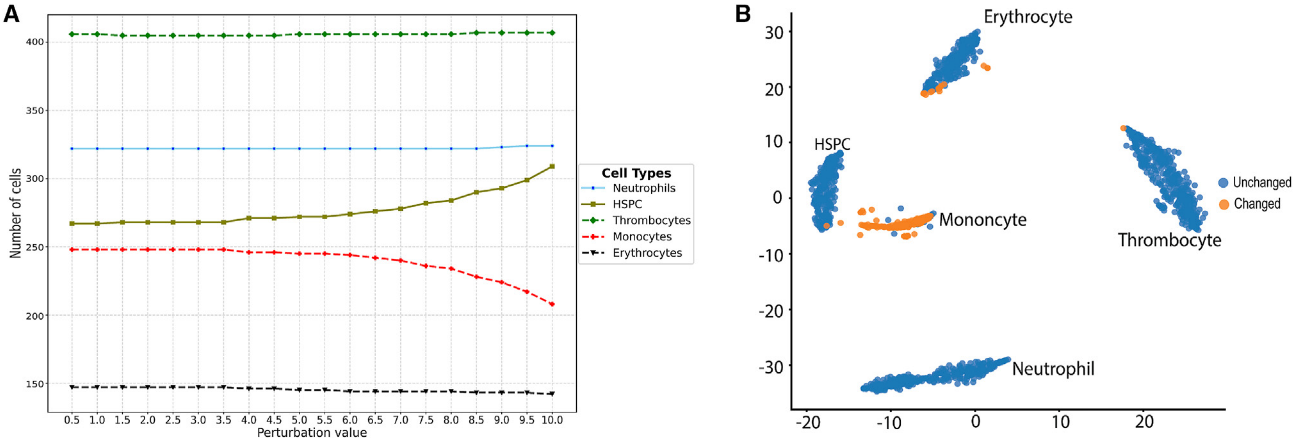


Figure 5. ***In silico*** **experiments to induce hematopoietic stem/progenitor cells using hematopoiesis in zebrafish**

(A) A series of latent layer perturbations and their effect on cell distribution.

(B) Cells that switch from their initial cell type to HSPCs are highlighted.

positive scores for the association between *Fli1* and megakaryocytes, monocytes, and granulocytes and also assigns a notable negative score to erythrocytes, in agreement with the literature[9,37] (Figure S4C). These simulations show that Fatecode accurately identifies known cell fate regulators that have been reported in previous perturbation-based experimental studies.

Furthermore, to evaluate the role of the top 200 genes detected by Fatecode for monocytes, we performed pathway enrichment analysis. Pathways that are significantly enriched in these 200 genes include those related to the immune system, hemopoiesis, cell development, and cell differentiation, which agrees with their Fatecode-predicted role in monocyte development (Figure 4F).

We extended our analysis to larger hematopoiesis single-cell differentiation data that involve differentiation into 12 cell types (Figure S5A).[23] We applied Fatecode to detect genes that can increase the pool of undifferentiated cells in this system (Figure S5B). One candidate detected by Fatecode in this process is *Entpd8*, the deletion of which in mice elevates the neutrophil and monocyte population.[38] Fatecode predictions are consistent with this experimental result. Fatecode also predicted *Nlrp6* as a regulator of neutrophil and monocyte differentiation. Cai et al. showed that the number of hematopoietic stem cells and GM progenitors is reduced in Kp-infected *Nlrp6*$^{-/-}$ mice, while the survival of mature neutrophils in bone marrow is increased.[39] We repeated gene set enrichment analysis using the top 200 genes detected by Fatecode. Biological processes related to mouse hematopoiesis, stem cell development, and metabolic signaling were enriched, showing that Fatecode can again capture relevant pathways for this biological process (Figure S5C).

Fatecode detects important regulators in cell differentiation and lineage commitment in zebrafish

We applied Fatecode to zebrafish hematopoiesis data[20] as an additional demonstration and test. From all possible perturbations on the latent layer performed by Fatecode, we selected ones that resulted in the greatest predicted relative increase in hematopoietic stem and progenitor cells (HSPCs) (Figure 5A). As shown in Figure 5B, following the perturbation, some cells (mostly monocytes) are predicted to switch to HSPCs (Figure 5B). Fatecode gives a significant score to signal transducer and activator of transcription 5A (*stat5a*) as one of the most important genes for HSPCs. *Stat5a* is a key regulator of normal hematopoiesis, with pleiotropic roles in hematopoietic stem cells.[40] Also, knockout studies have shown that the deletion of *stat5a* led to an increase in HSPC cycling, gradually reduced survival, and depleted the HSPC pool.[41] Next, Fatecode gives *irf8* a high positive score for monocytes. *Irf8* is a key regulator of monocyte development, and it has been known to be important for myelopoiesis in different model organisms.[42,43] It functions at an early step of the transcriptional program that governs differentiation from myeloid progenitors to monocytes/macrophages and plays a key role in stem cell renewal and maintenance.[43,44] Fatecode also identified a strong negative connection between *foxo3* and myeloid cell differentiation, consistent with *foxo3* knockout studies, which show a significant increase in granulocyte/monocyte progenitors in the spleen, bone marrow, and blood and enhance short-term hematopoietic stem cell proliferation.[45–47] Fatecode found an important role played by the *otud* gene family, a subgroup of deubiquitination enzymes, by assigning a high positive score between HSPCs and the *otud* gene family. Consistent with our prediction, knockout of *otud* genes in *Xenopus* results in developmental impairments.[48] Also, elevated expression of *otud* genes leads to the acquisition of stem cell properties.[49] Fatecode also predicted the negative score between *thbs1* and HSPCs, where *thbs1* has been shown previously to limit the expression of essential self-renewal transcription factors, including *oct3* and *oct4*, *sox2*, *klf4*, and *c-myc*, within cells.[50] Other key gene candidates identified by Fatecode for this perturbation are also known to be involved in hematopoiesis (Table 1).

Fatecode identifies cell fate regulators in mouse hippocampus development

To demonstrate Fatecode on a larger biological dataset, we applied it to developing mouse hippocampus scRNA-seq

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

Table 1. List of zebrafish hematopoiesis regulator genes predicted by Fatecode, with literature evidence for involvement in this process

Gene	Roles	Reference
cdk1	plays an important role in the maintenance of pluripotency and genomic stability in human pluripotent stem cells	Neganova et al.[70]
top2a	controls the survival of human pluripotent stem cells	Ben-David et al.[71]
hmgb2a	regulates hematopoietic stem cell maintenance	Zhang et al.[72]
ube2c	highly expressed in human embryonic stem cells (hESCs) and a biomarker of cancer stemness	Fatima et al.[73]; Liu et al.[74]
fbxo11	depletion leads to the hematopoietic population with stem cell characteristics	Mo et al.[75]
hmgn2	facilitates the maintenance of active chromatin states required for stem cell identity in a pluripotent stem cell model	Garza-Manero et al.[76]
aspm	regulates symmetric stem cell division by tuning Cyclin E ubiquitination	Capecchi et al.[77]
myb	regulates hematopoietic stem cell and myeloid progenitor cell development	Baker et al.[78]
kpna2	exhibits strong interactions with oct4 in embryonic stem cells	Li et al.[79]

data,[21] composed of 18,213 cells and 3,001 genes. The data are clustered in 14 annotated cell types (Figure 6A). We first sought to identify regulators in the differentiation process that preferentially increase mature granule cells (Figure 6B). Fatecode predicts the ZFP gene family (*Zfp94*, *Zfp189*, and *Zfp706*) as positively important in granule cell differentiation. The Zfp family is a definitive marker for the cerebellar granule neuron lineage and plays a critical role in granule cell specification within the developing cerebellum.[51] For example, lack of *Zfp521* results in a significant reduction in the number of granule cells.[52] *Id2* and *Id3* are important in maintaining the size and cellular structure of the brains of adult mice. It has also been shown that the absence of *Id2*$^{-/-}$ leads to a decrease in the number of granule neurons.[53,54] In line with this earlier research, Fatecode assigns a high positive score between both *Id2* and *Id3* for mature granule cells. These two transcriptional regulators have also been found to determine the fate of differentiating CD8^{+} T cells.[55]

Next, we applied Fatecode to determine regulators that mediate the differentiation process, which preferentially increases oligodendrocyte progenitor cells (OPCs), and decreases granule cells (both mature and immature) and oligodendrocytes. Fatecode predicted *Igfbpl1* as having an impact on OPC-to-oligodendrocyte differentiation, which is consistent with published experimental studies.[56,57] Furthermore, we considered *Fth1*, which provides neuroprotection and is enriched in oligodendrocytes. Mice lacking *Fth1* have more microglia cells compared to the control and a significant reduction in neurons and oligodendrocytes.[58] Fatecode accurately assigned a high positive score linking *Fth1* to oligodendrocytes and mature granule cells and a negative score for *Fth1* and microglia cells, showing that knocking out of *Fth1* leads to an increase in microglia cells, consistent with the experimental studies. Thymosin beta 4 (*Tmsb4x*) is a key candidate in the context of neurogenesis during brain development.[59] Its expression is linked to neurogenic processes and exerts regulatory control over the expansion of the stem cell pool within the early neuroepithelium. *Tmsb4x* gene knockout elicits a pronounced effect on the differentiation process *in vitro*. Specifically, it significantly promotes the differentiation of stem cells, further emphasizing its role in orchestrating cellular fate determination.[60] Our method correctly assigns a negative score for *Tmsb4x* and all cells except neuroblasts and radial glia-like cells. To further validate the performance of Fatecode in detecting key genes, we performed pathway enrichment analysis on the top 200 Fatecode-predicted regulators. This analysis showed that pathways related to brain development, synaptic signaling, and protein synthesis were significantly enriched in these genes (Figure 6C).

To illustrate further downstream analysis that is possible based on Fatecode results, we applied single-cell regulatory network inference and clustering (SCENIC) on the mouse hippocampus development dataset to construct a GRN consisting of the top 2,000 interactions based on their SCENIC importance measure scores, which shows the significance of the input gene (referred to as the "TF") in determining the prediction outcome for the target.[61] We then mapped the top 400 Fatecode-predicted regulators to the SCENIC-inferred GRN. The resulting networks can be used as a guide for identifying specific GRN mechanisms to target in follow-up experiments (*Ybx1* example, Figure S6) to test the regulatory relationships and potential roles of regulators in cellular reprogramming. While SCENIC predicts useful additional information to support experiment planning, it only considers transcription factor regulators. Other types of genes in Fatecode's output can be identified as cell fate regulators and should also be examined.

DISCUSSION

Cell reprogramming is a promising technology for tissue repair and regeneration, with the ultimate goal of accelerating recovery from diseases or injuries, as well as the development of novel therapies.[62] An important component in successful cell reprogramming is to correctly identify the regulators and trajectories from single-cell transcriptomics data. However, the number of genes in these datasets is large, and the number of underlying regulatory interactions is much larger. Recent studies have demonstrated that the expression of a single regulator is insufficient to produce an endpoint phenotype.[63] Instead, a group of control networks acts together across a variety of biological processes and pathways to induce a complete lineage conversion.[64] To efficiently and accurately map these control networks, we developed a deep learning method, Fatecode, which we have successfully applied to analyze diverse datasets. First,

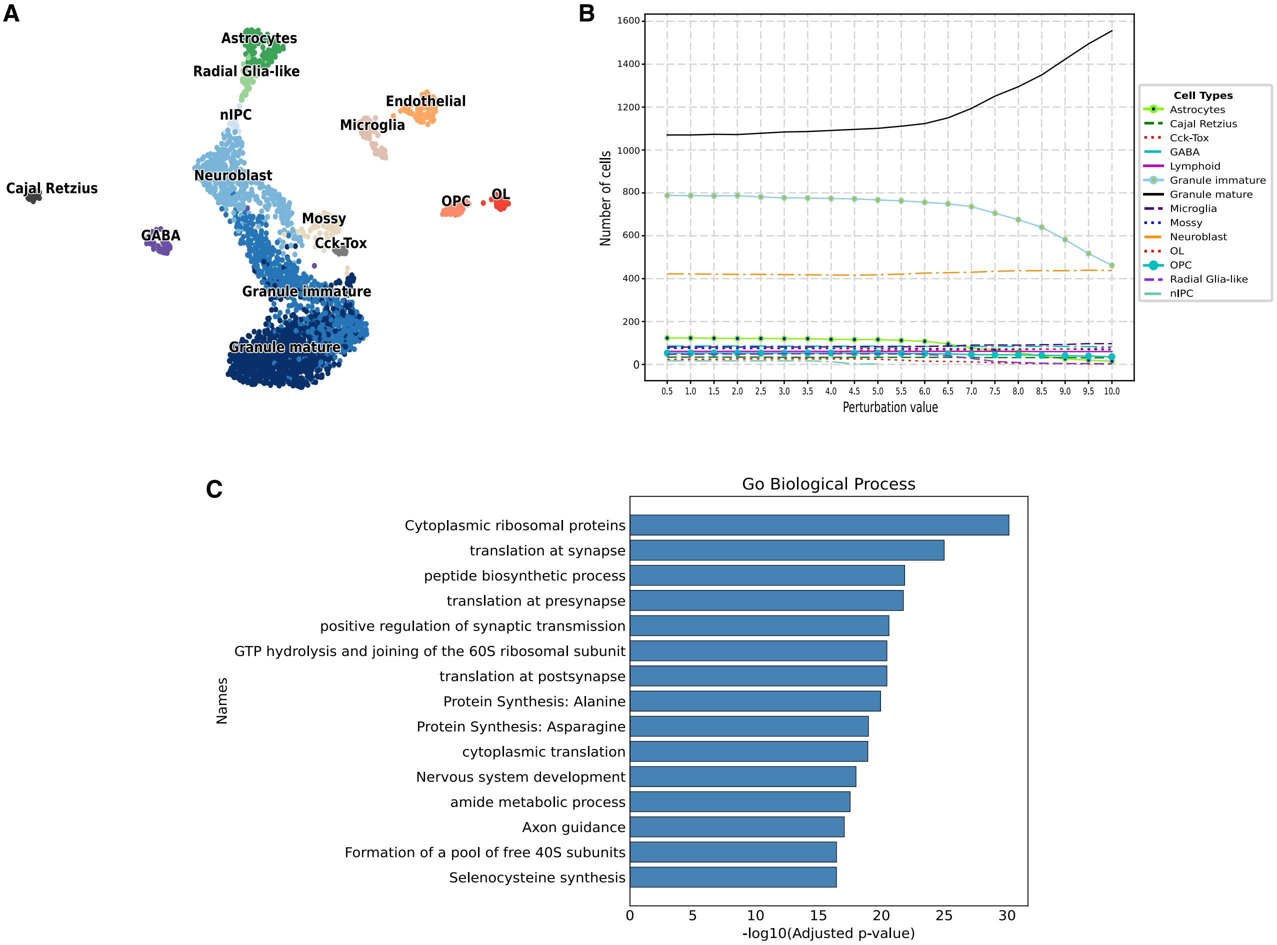


Figure 6. Fatecode identifies key genes in mouse neurogenesis

(A) UMAP embedding of fourteen major cell types.

(B) Latent layer node perturbation leads to an increase in mature granule cells while a decrease in immature granule cells.

(C) Pathway enrichment analysis shows the relevant biological process using the top 200 genes selected based on their prioritization scores for mature granule cells.

our method discovers an efficient architecture and latent layer for an input single-cell dataset. Then, by performing operations on the latent layer, it is able to predict perturbations for cell fate reprogramming. Fatecode was validated using simulated scRNA-seq data with predefined regulators and by predicting regulators in a variety of biological scRNA-seq data and manually comparing the results to the literature.

The fundamental idea in Fatecode is similar to the minimum Hamiltonian in physics and the potential energy landscape concept and representation learning.[27,65] These authors have shown that the most common autoencoders are naturally associated with an energy function, independent of the training procedure. This reasoning suggests that regulators can be seen as genes that allow the system to achieve a target cell type distribution via the most efficient path through the energy landscape. Fatecode uses the classifier as a guide to determine what node in the latent layer must be perturbed to achieve the desired reprogramming effect. Then the decoder maps the modified latent layer to gene space for gene identification. It is also useful to understand whether regulators are cell type specific. For example, the mammalian target of rapamycin complex (mTORC1) is widely important in cell fate decision-making and also important in the regulation of T cell fate.[66–69] Running Fatecode for different cell conversions can help identify cell-type-specific and non-specific regulators.

In conclusion, we developed an effective computational framework for predicting key players in cell fate control and the consequences of perturbations on cell type frequencies. Fatecode's modular design enables users to select an autoencoder architecture that produces an accurate model for their data. By leveraging the power of classification-supervised autoencoders and the associated energy manifold learning process, Fatecode generates useful hypotheses about genes that could be manipulated to achieve desired cell transitions. We hope this method accelerates the discovery of novel cell fate regulators that can be used to engineer and grow cells for therapeutic use in regenerative medicine applications.

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

Limitation of study

Fatecode can be thought of as an *in silico* CRISPR perturbation screen that identifies genes that can influence cell fate. Unfortunately, we were not able to find a published genome-wide CRISPR perturbation screen of an appropriate cell line and with a cell fate readout. Most genome-wide CRISPR-screens use standard cell lines that are not naturally multi-potent and, thus, are not expected to generate multiple cell fates. CRISPR has been used to evaluate cell fate regulators, but only by examining one or a few candidate genes in a single paper. We used these latter small-scale results to verify that Fatecode results agree with these experiments. Because we could not find genome-wide CRISPR screens with a cell fate readout, we used GRN simulations with predefined regulators and small-scale CRISPR experiments to validate our findings. In the future, we hope genome-scale CRISPR screens for cell fate regulators will be published for us to compare to.

Despite offering a useful input data representation, how the autoencoder latent layer represents the input data may be difficult to understand. Future work will need to better understand how the input data are represented and learned in the latent layer given diverse input data. However, our results showed that Fatecode predictions are relatively stable when changing the size of the latent layer, indicating that latent information is likely captured consistently.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - ○ Lead contact
  - ○ Materials availability
  - ○ Data and code availability
- METHOD DETAILS
  - ○ Deep representation learning
  - ○ VAE
  - ○ CVAE
  - ○ Overall network architecture of Fatecode
  - ○ Classification
  - ○ Identifying key regulators in cell differentiation
  - ○ Hyperparameter search
  - ○ Data visualization
  - ○ Data preprocessing
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.crmeth.2024.100819.

ACKNOWLEDGMENTS

This work is supported by the Canada 150 Research Chair program and by the National Science and Engineering Research Council of Canada via a Discovery award (RGPIN-2019-03916) to A.L.

AUTHOR CONTRIBUTIONS

Conceptualization, M.S., G.D.B., and S.G.; methodology, investigation, and data curation, M.S.; writing – original draft and review & editing, M.S., G.D.B., S.G., and A.L.; resources, G.D.B. and A.L.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: January 11, 2023
Revised: November 20, 2023
Accepted: June 18, 2024
Published: July 9, 2024

REFERENCES

1. Wang, H., Yang, Y., Liu, J., and Qian, L. (2021). Direct cell reprogramming: approaches, mechanisms and progress. Nat. Rev. Mol. Cell Biol. *22*, 410–424. https://doi.org/10.1038/s41580-021-00335-z.
2. Zimmermannova, O., Caiado, I., Ferreira, A.G., and Pereira, C.-F. (2021). Cell fate reprogramming in the era of cancer immunotherapy. Front. Immunol. *12*, 714822. https://doi.org/10.3389/fimmu.2021.714822.
3. Fleck, J.S., Jansen, S.M.J., Wollny, D., Zenk, F., Seimiya, M., Jain, A., Okamoto, R., Santel, M., He, Z., Camp, J.G., and Treutlein, B. (2023). Inferring and perturbing cell fate regulomes in human brain organoids. Nature *621*, 365–372. https://doi.org/10.1038/s41586-022-05279-8.
4. Alyagor, I., Berkun, V., Keren-Shaul, H., Marmor-Kollet, N., David, E., Mayseless, O., Issman-Zecharya, N., Amit, I., and Schuldiner, O. (2018). Combining Developmental and Perturbation-Seq Uncovers Transcriptional Modules Orchestrating Neuronal Remodeling. Dev. Cell *47*, 38–52.e6. https://doi.org/10.1016/j.devcel.2018.09.013.
5. Chen, S., Rivaud, P., Park, J.H., Tsou, T., Charles, E., Haliburton, J.R., Pichiorri, F., and Thomson, M. (2020). Dissecting heterogeneous cell populations across drug and disease conditions with PopAlign. Proc. Natl. Acad. Sci. USA *117*, 28784–28794. https://doi.org/10.1073/pnas.2005990117.
6. Lotfollahi, M., Wolf, F.A., and Theis, F.J. (2019). scGen predicts single-cell perturbation responses. Nat. Methods *16*, 715–721. https://doi.org/10.1038/s41592-019-0494-8.
7. Wei, X., Dong, J., and Wang, F. (2022). scPreGAN, a deep generative model for predicting the response of single cell expression to perturbation. Bioinformatics *38*, 3377–3384. https://doi.org/10.1093/bioinformatics/btac357.
8. Sadria, M., and Layton, A. (2023). The Power of Two: integrating deep diffusion models and variational autoencoders for single-cell transcriptomics analysis. Preprint at bioRxiv. https://doi.org/10.1101/2023.04.13.536789.
9. Kamimoto, K., Stringa, B., Hoffmann, C.M., Jindal, K., Solnica-Krezel, L., and Morris, S.A. (2023). Dissecting cell identity via network inference and in silico gene perturbation. Nature *614*, 742–751. https://doi.org/10.1038/s41586-022-05688-9.
10. Street, K., Risso, D., Fletcher, R.B., Das, D., Ngai, J., Yosef, N., Purdom, E., and Dudoit, S. (2018). Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. BMC Genom. *19*, 477. https://doi.org/10.1186/s12864-018-4772-0.
11. Ji, Z., and Ji, H. (2016). TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. Nucleic Acids Res. *44*, e117. https://doi.org/10.1093/nar/gkw430.
12. Tong, A., Huang, J., Wolf, G., van Dijk, D., and Krishnaswamy, S. (2020). Trajectorynet: A dynamic optimal transport network for modeling cellular dynamics. Proc. Mach. Learn. Res. *119*, 9526–9536. https://doi.org/10.48550/arxiv.2002.04461.
13. Pandey, K., and Zafar, H. (2022). Inference of cell state transitions and cell fate plasticity from single-cell with MARGARET. Nucleic Acids Res. *50*, e86. https://doi.org/10.1093/nar/gkac412.
14. Setty, M., Kiseliovas, V., Levine, J., Gayoso, A., Mazutis, L., and Pe'er, D. (2019). Characterization of cell fate probabilities in single-cell data with Palantir. Nat. Biotechnol. *37*, 451–460. https://doi.org/10.1038/s41587-019-0068-4.

15. Tritschler, S., Büttner, M., Fischer, D.S., Lange, M., Bergen, V., Lickert, H., and Theis, F.J. (2019). Concepts and limitations for learning developmental trajectories from single cell genomics. Development *146*, dev170506. https://doi.org/10.1242/dev.170506.

16. Sadria, M., and Bury, T.M. (2024). FateNet: an integration of dynamical systems and deep learning for cell fate prediction. Preprint at bioRxiv. https://doi.org/10.1101/2024.01.16.575913.

17. Zhu, Q., and Zhang, R. (2019). A Classification Supervised Auto-Encoder Based on Predefined Evenly-Distributed Class Centroids. Preprint at arXiv preprint arXiv:1902.00220. https://doi.org/10.48550/arXiv.1902.00220.

18. Abdolhosseini, F., Azarkhalili, B., Maazallahi, A., Kamal, A., Motahari, S.A., Sharifi-Zarchi, A., and Chitsaz, H. (2019). Cell Identity Codes: Understanding Cell Identity from Gene Expression Profiles using Deep Neural Networks. Sci. Rep. *9*, 2342. https://doi.org/10.1038/s41598-019-38798-y.

19. Dibaeinia, P., and Sinha, S. (2020). SERGIO: A Single-Cell Expression Simulator Guided by Gene Regulatory Networks. Cell Syst. *11*, 252–271.e11. https://doi.org/10.1016/j.cels.2020.08.003.

20. Athanasiadis, E.I., Botthof, J.G., Andres, H., Ferreira, L., Lio, P., and Cvejic, A. (2017). Single-cell RNA-sequencing uncovers transcriptional states and fate decisions in haematopoiesis. Nat. Commun. *8*, 2045. https://doi.org/10.1038/s41467-017-02305-6.

21. Hochgerner, H., Zeisel, A., Lönnerberg, P., and Linnarsson, S. (2018). Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing. Nat. Neurosci. *21*, 290–299. https://doi.org/10.1038/s41593-017-0056-2.

22. Paul, F., Arkin, Y., Giladi, A., Jaitin, D.A., Kenigsberg, E., Keren-Shaul, H., Winter, D., Lara-Astiaso, D., Gury, M., Weiner, A., et al. (2015). Transcriptional heterogeneity and lineage commitment in myeloid progenitors. Cell *163*, 1663–1677. https://doi.org/10.1016/j.cell.2015.11.013.

23. Weinreb, C., Rodriguez-Fraticelli, A., Camargo, F.D., and Klein, A.M. (2020). Lineage tracing on transcriptional landscapes links state to fate during differentiation. Science *367*, eaaw3381. https://doi.org/10.1126/science.aaw3381.

24. Press, O., Galanti, T., Benaim, S., and Wolf, L. (2020). Emerging Disentanglement in Auto-Encoder Based Unsupervised Image Content Transfer. Preprint at arXiv. 10.48550/arxiv.2001.05017. https://doi.org/10.48550/arXiv.2001.05017.

25. Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. Preprint at arXiv. https://doi.org/10.48550/arXiv.1511.06434.

26. Klys, J., Snell, J., and Zemel, R. (2018). Learning latent subspaces in variational autoencoders. Adv. Neural Inf. Process. Syst. *31*.

27. Khemakhem, I., Monti, R.P., Kingma, D.P., and Hyvärinen, A. (2020). ICE-BeeM: Identifiable Conditional Energy-Based Deep Models Based on Nonlinear ICA. Preprint at arXiv. https://doi.org/10.48550/arxiv.2002.11537.

28. Plumerault, A., Le Borgne, H., and Hudelot, C. (2021). AVAE: adversarial variational auto encoder. In 2020 25th International Conference on Pattern Recognition (ICPR) (IEEE)), pp. 8687–8694. https://doi.org/10.1109/ICPR48806.2021.9412727.

29. An, J., and Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. Special Lecture on IE.

30. Dai, B., Wang, Z., and Wipf, D. (2020). The usual suspects? Reassessing blame for VAE posterior collapse. International Conference on Machine Learning, p. 2313.

31. Li, P., Pei, Y., and Li, J. (2023). A comprehensive survey on design and application of autoencoder in deep learning. Appl. Soft Comput. *138*, 110176. https://doi.org/10.1016/j.asoc.2023.110176.

32. Bastidas-Ponce, A., Tritschler, S., Dony, L., Scheibner, K., Tarquis-Medina, M., Salinno, C., Schirge, S., Burtscher, I., Böttcher, A., Theis, F.J., et al. (2019). Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. Development *146*, dev173849. https://doi.org/10.1242/dev.173849.

33. Hao, Y., Hao, S., Andersen-Nissen, E., Mauck, W.M., Zheng, S., Butler, A., Lee, M.J., Wilk, A.J., Darby, C., Zager, M., et al. (2021). Integrated analysis of multimodal single-cell data. Cell *184*, 3573–3587.e29. https://doi.org/10.1016/j.cell.2021.04.048.

34. Faure, L., Soldatov, R., Kharchenko, P.V., and Adameyko, I. (2023). scFates: a scalable python package for advanced pseudotime and bifurcation analysis from single-cell data. Bioinformatics *39*, btac746. https://doi.org/10.1093/bioinformatics/btac746.

35. Rosenbauer, F., and Tenen, D.G. (2007). Transcription factors in myeloid development: balancing differentiation with transformation. Nat. Rev. Immunol. *7*, 105–117. https://doi.org/10.1038/nri2024.

36. Pijuan-Sala, B., Griffiths, J.A., Guibentif, C., Hiscock, T.W., Jawaid, W., Calero-Nieto, F.J., Mulas, C., Ibarra-Soria, X., Tyser, R.C.V., Ho, D.L.L., et al. (2019). A single-cell molecular map of mouse gastrulation and early organogenesis. Nature *566*, 490–495. https://doi.org/10.1038/s41586-019-0933-9.

37. Ben-David, Y., Gajendran, B., Sample, K.M., and Zacksenhaus, E. (2022). Current insights into the role of Fli-1 in hematopoiesis and malignant transformation. Cell. Mol. Life Sci. *79*, 163. https://doi.org/10.1007/s00018-022-04160-1.

38. Tani, H., Li, B., Kusu, T., Okumura, R., Nishimura, J., Okuzaki, D., Motooka, D., Arakawa, S., Mori, A., Yoshihara, T., et al. (2021). The ATP-hydrolyzing ectoenzyme E-NTPD8 attenuates colitis through modulation of P2X4 receptor-dependent metabolism in myeloid cells. Proc. Natl. Acad. Sci. USA *118*, e2100594118. https://doi.org/10.1073/pnas.2100594118.

39. Cai, S., Paudel, S., Jin, L., Ghimire, L., Taylor, C.M., Wakamatsu, N., Bhattarai, D., and Jeyaseelan, S. (2021). NLRP6 modulates neutrophil homeostasis in bacterial pneumonia-derived sepsis. Mucosal Immunol. *14*, 574–584. https://doi.org/10.1038/s41385-020-00357-4.

40. Wang, Z., and Bunting, K.D. (2013). STAT5 in hematopoietic stem cell biology and transplantation. JAK-STAT *2*, e27159. https://doi.org/10.4161/jkst.27159.

41. Wang, Z., Li, G., Tse, W., and Bunting, K.D. (2009). Conditional deletion of STAT5 in adult mouse hematopoietic stem cells causes loss of quiescence and permits efficient nonablative stem cell replacement. Blood *113*, 4856–4865. https://doi.org/10.1182/blood-2008-09-181107.

42. Yáñez, A., Ng, M.Y., Hassanzadeh-Kiabi, N., and Goodridge, H.S. (2015). IRF8 acts in lineage-committed rather than oligopotent progenitors to control neutrophil vs monocyte production. Blood *125*, 1452–1459. https://doi.org/10.1182/blood-2014-09-600833.

43. Lee, J., Zhou, Y.J., Ma, W., Zhang, W., Aljoufi, A., Luh, T., Lucero, K., Liang, D., Thomsen, M., Bhagat, G., et al. (2017). Lineage specification of human dendritic cells is marked by IRF8 expression in hematopoietic stem cells and multipotent progenitors. Nat. Immunol. *18*, 877–888. https://doi.org/10.1038/ni.3789.

44. Hambleton, S., Salem, S., Bustamante, J., Bigley, V., Boisson-Dupuis, S., Azevedo, J., Fortin, A., Haniffa, M., Ceron-Gutierrez, L., Bacon, C.M., et al. (2011). IRF8 mutations and human dendritic-cell immunodeficiency. N. Engl. J. Med. *365*, 127–138. https://doi.org/10.1056/NEJMoa1100066.

45. Xie, X.w., Liu, J.-X., Hu, B., and Xiao, W. (2011). Zebrafish foxo3b negatively regulates canonical Wnt signaling to affect early embryogenesis. PLoS One *6*, e24469. https://doi.org/10.1371/journal.pone.0024469.

46. Tsuchiya, K., Westerterp, M., Murphy, A.J., Subramanian, V., Ferrante, A.W., Tall, A.R., and Accili, D. (2013). Expanded granulocyte/monocyte compartment in myeloid-specific triple FoxO knockout increases oxidative stress and accelerates atherosclerosis in mice. Circ. Res. *112*, 992–1003. https://doi.org/10.1161/CIRCRESAHA.112.300749.

47. Tothova, Z., Kollipara, R., Huntly, B.J., Lee, B.H., Castrillon, D.H., Cullen, D.E., McDowell, E.P., Lazo-Kallanian, S., Williams, I.R., Sears, C., et al. (2007). FoxOs are critical mediators of hematopoietic stem cell resistance to physiologic oxidative stress. Cell *128*, 325–339. https://doi.org/10.1016/j.cell.2007.01.003.

48. Job, F., Mai, C., Villavicencio-Lorini, P., Herfurth, J., Neuhaus, H., Hoffmann, K., Pfirrmann, T., and Hollemann, T. (2023). OTUD3: A Lys6 and Lys63 specific deubiquitinase in early vertebrate development. Biochim. Biophys. Acta. Gene Regul. Mech. *1866*, 194901. https://doi.org/10.1016/j.bbagrm.2022.194901.

49. Zhang, Z., Zhao, W., Li, Y., Li, Y., Cheng, H., Zheng, L., Sun, X., Liu, H., and Shao, R. (2022). YOD1 serves as a potential prognostic biomarker for pancreatic cancer. Cancer Cell Int. *22*, 203. https://doi.org/10.1186/s12935-022-02616-9.

50. Kaur, S., Soto-Pantoja, D.R., Stein, E.V., Liu, C., Elkahloun, A.G., Pendrak, M.L., Nicolae, A., Singh, S.P., Nie, Z., Levens, D., et al. (2013). Thrombospondin-1 signaling through CD47 inhibits self-renewal by regulating c-Myc and other stem cell transcription factors. Sci. Rep. *3*, 1673. https://doi.org/10.1038/srep01673.

51. Yang, X.W., Zhong, R., and Heintz, N. (1996). Granule cell specification in the developing mouse brain as defined by expression of the zinc finger transcription factor RU49. Development *122*, 555–566. https://doi.org/10.1242/dev.122.2.555.

52. Bu, S., Lv, Y., Liu, Y., Qiao, S., and Wang, H. (2021). Zinc Finger Proteins in Neuro-Related Diseases Progression. Front. Neurosci. *15*, 760567. https://doi.org/10.3389/fnins.2021.760567.

53. Havrda, M.C., Harris, B.T., Mantani, A., Ward, N.M., Paolella, B.R., Cuzon, V.C., Yeh, H.H., and Israel, M.A. (2008). Id2 is required for specification of dopaminergic neurons during adult olfactory neurogenesis. J. Neurosci. *28*, 14074–14086. https://doi.org/10.1523/JNEUROSCI.3188-08.2008.

54. Pleasure, S.J., Collins, A.E., and Lowenstein, D.H. (2000). Unique expression patterns of cell fate molecules delineate sequential stages of dentate gyrus development. J. Neurosci. *20*, 6095–6105. https://doi.org/10.1523/JNEUROSCI.20-16-06095.2000.

55. Yang, C.Y., Best, J.A., Knell, J., Yang, E., Sheridan, A.D., Jesionek, A.K., Li, H.S., Rivera, R.R., Lind, K.C., D'Cruz, L.M., et al. (2011). The transcriptional regulators Id2 and Id3 control the formation of distinct memory CD8+ T cell subsets. Nat. Immunol. *12*, 1221–1229. https://doi.org/10.1038/ni.2158.

56. Kühl, N.M., De Keyser, J., De Vries, H., and Hoekstra, D. (2002). Insulin-like growth factor binding proteins-1 and -2 differentially inhibit rat oligodendrocyte precursor cell survival and differentiation in vitro. J. Neurosci. Res. *69*, 207–216. https://doi.org/10.1002/jnr.10293.

57. La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M.E., Lönnerberg, P., Furlan, A., et al. (2018). RNA velocity of single cells. Nature *560*, 494–498. https://doi.org/10.1038/s41586-018-0414-6.

58. Mukherjee, C., Kling, T., Russo, B., Miebach, K., Kess, E., Schifferer, M., Pedro, L.D., Weikert, U., Fard, M.K., Kannaiyan, N., et al. (2020). Oligodendrocytes provide antioxidant defense function for neurons by secreting ferritin heavy chain. Cell Metabol. *32*, 259–272.e10. https://doi.org/10.1016/j.cmet.2020.05.019.

59. Qian, L., Huang, Y., Spencer, C.I., Foley, A., Vedantham, V., Liu, L., Conway, S.J., Fu, J.d., and Srivastava, D. (2012). In vivo reprogramming of murine cardiac fibroblasts into induced cardiomyocytes. Nature *485*, 593–598. https://doi.org/10.1038/nature11044.

60. Wirsching, H.-G., Krishnan, S., Florea, A.-M., Frei, K., Krayenbühl, N., Hasenbach, K., Reifenberger, G., Weller, M., and Tabatabai, G. (2014). Thymosin β 4 gene silencing decreases stemness and invasiveness in glioblastoma. Brain *137*, 433–448. https://doi.org/10.1093/brain/awt333.

61. Aibar, S., González-Blas, C.B., Moerman, T., Huynh-Thu, V.A., Imrichova, H., Hulselmans, G., Rambow, F., Marine, J.-C., Geurts, P., Aerts, J., et al. (2017). SCENIC: single-cell regulatory network inference and clustering. Nat. Methods *14*, 1083–1086. https://doi.org/10.1038/nmeth.4463.

62. Lin, B., Srikanth, P., Castle, A.C., Nigwekar, S., Malhotra, R., Galloway, J.L., Sykes, D.B., and Rajagopal, J. (2018). Modulating cell fate as a therapeutic strategy. Cell Stem Cell *23*, 329–341. https://doi.org/10.1016/j.stem.2018.05.009.

63. Davis, T.L., and Rebay, I. (2017). Master regulators in development: Views from the Drosophila retinal determination and mammalian pluripotency gene networks. Dev. Biol. *421*, 93–107. https://doi.org/10.1016/j.ydbio.2016.12.005.

64. Oestreich, K.J., and Weinmann, A.S. (2012). Master regulators or lineage-specifying? Changing views on CD4+ T cell transcription factors. Nat. Rev. Immunol. *12*, 799–804. https://doi.org/10.1038/nri3321.

65. Kamyshanska, H., and Memisevic, R. (2015). The potential energy of an autoencoder. IEEE Trans. Pattern Anal. Mach. Intell. *37*, 1261–1273. https://doi.org/10.1109/TPAMI.2014.2362140.

66. Chi, H. (2012). Regulation and function of mTOR signalling in T cell fate decisions. Nat. Rev. Immunol. *12*, 325–338. https://doi.org/10.1038/nri3198.

67. Sadria, M., and Layton, A.T. (2021). Interactions among mTORC, AMPK and SIRT: a computational model for cell energy balance and metabolism. Cell Commun. Signal. *19*, 57. https://doi.org/10.1186/s12964-021-00706-1.

68. Sadria, M., Seo, D., and Layton, A.T. (2022). The mixed blessing of AMPK signaling in Cancer treatments. BMC Cancer *22*, 105. https://doi.org/10.1186/s12885-022-09211-1.

69. Tatapudy, S., Aloisio, F., Barber, D., and Nystul, T. (2017). Cell fate decisions: emerging roles for metabolic signals and cell morphology. EMBO Rep. *18*, 2105–2118. https://doi.org/10.15252/embr.201744816.

70. Neganova, I., Tilgner, K., Buskin, A., Paraskevopoulou, I., Atkinson, S.P., Peberdy, D., Passos, J.F., and Lako, M. (2014). CDK1 plays an important role in the maintenance of pluripotency and genomic stability in human pluripotent stem cells. Cell Death Dis. *5*, e1508. https://doi.org/10.1038/cddis.2014.464.

71. Ben-David, U., Cowell, I.G., Austin, C.A., and Benvenisty, N. (2015). Brief reports: Controlling the survival of human pluripotent stem cells by small molecule-based targeting of topoisomerase II alpha. Stem Cell. *33*, 1013–1019. https://doi.org/10.1002/stem.1888.

72. Zhang, C., Fondufe-Mittendorf, Y.N., Wang, C., Chen, J., Cheng, Q., Zhou, D., Zheng, Y., Geiger, H., and Liang, Y. (2020). Latexin regulation by HMGB2 is required for hematopoietic stem cell maintenance. Haematologica *105*, 573–584. https://doi.org/10.3324/haematol.2018.207092.

73. Fatima, A., Irmak, D., Noormohammadi, A., Rinschen, M.M., Das, A., Leidecker, O., Schindler, C., Sánchez-Gaya, V., Wagle, P., Pokrzywa, W., et al. (2020). The ubiquitin-conjugating enzyme UBE2K determines neurogenic potential through histone H3 in human embryonic stem cells. Commun. Biol. *3*, 262. https://doi.org/10.1038/s42003-020-0984-3.

74. Liu, P.-F., Chen, C.-F., Shu, C.-W., Chang, H.-M., Lee, C.-H., Liou, H.-H., Ger, L.-P., Chen, C.-L., and Kang, B.-H. (2020). UBE2C is a potential biomarker for tumorigenesis and prognosis in tongue squamous cell carcinoma. Diagnostics *10*, 674. https://doi.org/10.3390/diagnostics10090674.

75. Mo, A.Y.-C., Kincross, H., Wang, X., Chang, L.Y.-T., Duns, G., Kwan, H., Lau, T., Docking, T.R., Tran, J., Colborne, S., et al. (2022). Loss of FBXO11 function establishes a stem cell program in acute myeloid leukemia through dysregulation of the mitochondrial protease LONP1. Preprint at bioRxiv. https://doi.org/10.1101/2022.09.10.507366.

76. Garza-Manero, S., Sindi, A.A.A., Mohan, G., Rehbini, O., Jeantet, V.H.M., Bailo, M., Latif, F.A., West, M.P., Gurden, R., Finlayson, L., et al. (2019). Maintenance of active chromatin states by HMGN2 is required for stem cell identity in a pluripotent stem cell model. Epigenet. Chromatin *12*, 73. https://doi.org/10.1186/s13072-019-0320-7.

77. Capecchi, M.R., and Pozner, A. (2015). ASPM regulates symmetric stem cell division by tuning Cyclin E ubiquitination. Nat. Commun. *6*, 8763. https://doi.org/10.1038/ncomms9763.

78. Baker, S.J., Ma'ayan, A., Lieu, Y.K., John, P., Reddy, M.V.R., Chen, E.Y., Duan, Q., Snoeck, H.-W., and Reddy, E.P. (2014). B-myb is an essential regulator of hematopoietic stem cell and myeloid progenitor cell development. Proc. Natl. Acad. Sci. USA *111*, 3122–3127. https://doi.org/10.1073/pnas.1315464111.

79. Li, X., Sun, L., and Jin, Y. (2008). Identification of karyopherin-alpha 2 as an Oct4 associated protein. J. Genet. Genomics *35*, 723–728. https://doi.org/10.1016/S1673-8527(08)60227-1.

80. Mikolov, T., Yih, W.T., and Zweig, G. (2013). Linguistic regularities in continuous space word representations. Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, *746*.

81. Franz, M., Lopes, C.T., Fong, D., Kucera, M., Cheung, M., Siper, M.C., Huck, G., Dong, Y., Sumer, O., and Bader, G.D. (2023). Cytoscape.js 2023 update: a graph theory library for visualization and analysis. Bioinformatics *39*, btad031. https://doi.org/10.1093/bioinformatics/btad031.

82. Fang, Z., Liu, X., and Peltz, G. (2023). GSEApy: a comprehensive package for performing gene set enrichment analysis in Python. Bioinformatics *39*, btac757. https://doi.org/10.1093/bioinformatics/btac757.

83. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., and Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Zebrafish hematopoiesis data	Athanasiadis et.al.[20]	E-MTAB-5530
Raw and analyzed data	This paper	https://doi.org/10.5281/zenodo.11511340
Dentate Gyrus neurogenesis	Hochgerner et al.[21]	GSE95753
Hematopoiesis	Paul et al.[22]	GSE72859
Hematopoiesis	Weinreb et al.[23]	GSE140802
SERGIO	Dibaeinia et al.[19]	https://github.com/PayamDiba/SERGIO
Software and algorithms		
Fatecode	This paper	https://doi.org/10.5281/zenodo.11511340
Cytoscape	Franz et al.[81]	https://doi.org/10.1093/bioinformatics/btad031
scFates	Faure et al.[34]	https://doi.org/10.1093/bioinformatics/btac746
Scenic	Aibar et al.[61]	https://doi.org/10.1038/nmeth.4463
gseapy	Fang et al.[82]	https://github.com/zqfang/GSEApy
scikit-learn	Alex et al.[83]	https://scikit-learn.org/
Seurat	CRAN	https://satijalab.org/seurat/

RESOURCE AVAILABILITY

Lead contact

Lead contact Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Mehrshad Sadria (msadria@uwaterloo.ca)

Materials availability

This study did not generate new unique reagents.

Data and code availability

- The datasets used in the present study are openly accessible in public repositories. The zebrafish hematopoiesis data can be found under the accession number E-MTAB-5530 on ArrayExpress. We downloaded a preprocessed version of the "Dentate Gyrus neurogenesis" data (under accession number GSE95753) from https://scvelo.readthedocs.io/en/stable/. The hematopoiesis Paul et al. data can be downloaded from the GEO under accession code GSE72859 and the preprocessed version was downloaded from https://celloracle.org/. To generate simulated data, we used the same parameters for the differential equations as in https://github.com/PayamDiba/SERGIO.[19] The hematopoiesis Weinreb et al. data can be downloaded from GEO under accession number GSE140802 and the preprocessed version was downloaded from https://cospar.readthedocs.io/en/latest/.
- Code supporting this study is available on: https://github.com/MehrshadSD/Fatecode and https://doi.org/10.5281/zenodo.11511340.
- Any additional information required to re-analyze the results reported by this study are available from the lead contact upon request.

METHOD DETAILS

Deep representation learning

Autoencoders are a class of neural networks with a latent layer capable of learning nonlinear representations of the input data in an unsupervised manner. An autoencoder consists of an encoder that maps the input to the latent space and a decoder which transfers the latent space back to the original space. It can be used for denoising, reducing dimensionality, or learning the representation (or manifold) of the data. We implemented three autoencoder architectures: under-complete AutoEncoder (AE), Variational AutoEncoder (VAE), and Conditional VAriational Encoder (CVAE)[31] (Figure 1). AE has a single latent layer. VAE constrains the latent layer by modeling the latent space as a multivariate Gaussian distribution with a mean and a standard deviation. CVAE conditions the latent space on class labels and thus can generate data based on a given class label. The biological task for our autoencoder is to learn a

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

reduced dimension representation of a cell by gene matrix capturing measurements of a single-cell transcriptomics experiment mapping cellular trajectories. Only the gene dimension is reduced, so the latent space describes a reduced representation of each input cell transcriptome. To make the latent layer more specific for our biological task, we added a cell type classification task to the standard regression tasks. The classification task, described in more detail below, predicts the type of each latent cell and compares it to a known input cell type. The training process works to optimize both classification and regression performance simultaneously. This reduces the space of latent layer candidates since not all possible latent layers are useful for the classification task.

VAE

VAE is a type of autoencoder that estimates a latent set of probability density functions that model the input data. Unlike AE, which learns an unconstrained representation of the data, VAE assumes a Gaussian distribution for the prior. An input gene by cell matrix X is run through an encoder, which generates parameters for the set of distributions $Q(z\,|X)$. Then, from Q, a latent k-vector z is sampled and the decoder transforms z into an output, with the condition that the output is similar to the input, where k equals the number of components (or distributions) in the VAE. The VAE total loss consists of the reconstruction loss (first term) and the KL-divergence loss (second term):

$$E[\log P(X|z)] - D_{KL}[\mathrm{Q}(z|\mathrm{X})\|P(z)]$$

$$D_{KL}[Q(z|X)\|P(z)] = -\frac{1}{2}\sum_{k=1}^{K}\left[1 + log\,\sigma_k{}^2 - \mu_k{}^2 - \sigma_k{}^2\right]$$

where μ_k and σ_k are the k-th components of output vectors μ_k (X) and σ_k (X), respectively.

CVAE

CVAE is distinguished from VAE by its embedding of conditional information in the objective function. CVAE relies on two inputs: the features and the class labels, c, instead of using only the features, as is done with a VAE and AE. The CVAE architecture allows the encoder and the decoder to be conditioned by c. Hence, the variational lower bound objective is changed to the following form.

$$E[\log P(X|z, c)] - D_{KL}[\mathrm{Q}(z|\mathrm{X}, \mathrm{c})\|P(z|c)],$$

Overall network architecture of Fatecode

The Fatecode autoencoder architecture was chosen for each of the datasets analyzed in this study using a hyperparameter search (More details in Hyperparameter search section). Encoder and decoder architectures are constrained to have the same number of outer and inner layer nodes. For the analysis of hematopoiesis regulation in zebrafish, Fatecode consists of a fully connected encoder and decoder. The encoder and decoder are both two-layer networks of 92 (outer layer) and 48 (inner layer) nodes with the LeakyReLU activation function and the latent layer has 18 nodes. For the analysis of hematopoiesis in mouse data by Weinreb et al.,[23] the encoder/decoder has a 506-node outer layer and a 253-node inner layer, and the latent layer has 125 nodes. For the mouse hematopoiesis data by Paul et al.[22] the encoder/decoder has a 100-node outer layer and a 40-node and the latent layer has 20 nodes. For the developing mouse hippocampus data, we used a two-layer encoder/decoder of 50 (outer), 26 (inner), and a latent layer of 15 nodes. Our model was built using software packages and libraries, including TensorFlow V2.10.0, scikit-learn V1.1.3, scanpy V1.9.1, numpy V1.23.4, and pandas V1.5.1.

Classification

The classifier determines cell types using the latent layer as input to a single hidden layer and then an output layer (with one node per cell type), all fully connected. ReLu and softmax activation functions are used for the hidden and output layers, respectively. The number of nodes in the hidden layer is varied during the hyperparameter optimization. For adult zebrafish blood data,[20] we use 15 and 5 nodes for the hidden and output layers, respectively. We use 25 and 12 nodes for classifying hematopoiesis in mouse data by Weinreb et al.,[23] 20 and 9 nodes for data from Paul et al.,[22] and 22 and 14 for the developing mouse hippocampus data.[21] All cell labels are assigned by using the predefined cell type labels of the original studies.

Identifying key regulators in cell differentiation

Consider adjustments (e.g., one or more gene knock-outs or over-expressions) that will transition a baseline cell type distribution (“A”) to a given desired target distribution (“B”). For example, in the target cell distribution, our objective is to increase the number of cell type N while decreasing the number of cell type P (Figure S1). To detect genes that are important in a given transition, Fatecode analyzes the effects of perturbations on cell fate by systematically perturbing individual autoencoder latent nodes learned from a single-cell transcriptomics data capturing cellular trajectories. Each latent variable perturbation results in a single-cell transcriptome through the decoding process and a corresponding cell type distribution, proceeding as follows after training Fatecode.

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

(1) The gene expression data, denoted as E, corresponding to a mixture of cells with cell type distribution A, undergoes encoding to produce a matrix of latent variables represented as X ($X = encoder(E)$). Each column of X is associated with a cell in E; each row corresponds to a latent variable).

(2) In a series of simulations, finite perturbations of different sizes ***K*** (e.g., from a 50% reduction to a 10-fold increase) are applied to each row j (number of latent variables) in X sequentially. For each perturbed latent layer row, $X_j{}^*$

$$X_j{}^* = kX_j$$

(3) We then run the cell type classifier trained within Fatecode on the perturbed latent layer to predict the cell type distribution for each across all perturbation conditions.

$$New\_cell\_type\_distribution = classifier(X_j{}^*)$$

(4) Then, we can identify a perturbed latent layer row, $X_j{}^*$, and its associated perturbation size, k, that is closest to the desired target distribution B.

(5) To identify genes important for the transition from cell type distributions A to B, we compute the difference between the selected X^* and the X latent layers. For instance, if increasing latent node #9 5-fold can best approximate the desired distribution B, then the difference between the selected X^* and X latent layers is a latent node by cell matrix with all zero entries, except for the 9th row, which is 5 times X_9.

(6) With this selected perturbation matrix $(X^* - X)$, the decoder produces a gene-by-cell matrix. Then the average gene expression profile of all cells in each cell type is computed, resulting in a gene by cell_type matrix *M.* The (*i,j*)-th entry of *M* is the prioritization score for the *i*-th gene in cell_type *j.*

(7) To identify the regulators predicted to be important for transitioning initial cell type distribution A to target B, we rank the genes based on their prioritization scores for a cell type of interest.

$$Regulators = sort\left(M_{desired\_celltype}\right)$$

We note that *M* does not directly specify how much each gene should be perturbed to yield target B. Nonetheless, *M* contains information about genes that are important in transitioning cell type distribution from initial state A to the desired state B. This idea is similar to potential energy in physics and representation learning.[27,65]

We also examine the model's performance in detecting regulators when operating on the output of the decoder compared to the latent layer. To achieve this, we feed the perturbed vector to the decoder and subtract the result from the unperturbed condition. We then investigate the genes that show significant changes. Our results indicate that working on the latent layer leads to better outcomes in detecting regulators than operating on the output of the decoder. This observation is in line with previous research in computer vision and natural language processing, where using the latent space consistently yields superior results compared to the original data space.[25,80] We assume this is true in general when using an autoencoder with a non-linear activation function with reasonably complex data, as we have in biology (in contrast to the linear activation function case where $Decoder(X_{perturbed}) - Decoder(X) = Decoder(X_{perturbed} - X)$).

Hyperparameter search

Hyperparameter tuning was conducted using a grid search approach. We explored various combinations of hyperparameters, including learning rate, batch size, number of layers, number of neurons per layer, and activation functions. The hyperparameter space for each parameter was defined as follows.

(1) Autoencoder type: [AE, VAE, CVAE]
(2) Activation function: [LeakyReLU, Relu, linear]
(3) Learning rate: [0.001, 0.01, 0.1]
(4) Batch size: [400, 500, 600]
(5) Number of hidden layers (encoder): [1, 2]
(6) Number of neurons in latent_layer: [input_size/40, input_size/60, input_size/80, input_size/100, input_size/125, input_size/150]

Data visualization

Python package "UMAP" was used to visualize the latent layer as a reduced dimensionality space and for network visualizations we used Cytoscape.[81]

Data preprocessing

The scRNA-Seq gene expression data is log normalized, scaled, and centered. In the training process, 80% of the data is allocated for training the classification-supervised autoencoder, while the remaining 20% is used for testing purposes.

Please cite this article in press as: Sadria et al., Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation, Cell Reports Methods (2024), https://doi.org/10.1016/j.crmeth.2024.100819

QUANTIFICATION AND STATISTICAL ANALYSIS

Differential gene expression analysis was performed using the Wilcoxon rank-sum test. To account for multiple testing, we applied the Benjamini–Hochberg correction to the calculated P-values obtained from the DEG analysis. Genes with a corrected *p*-value below 0.05 were considered statistically significant. For scFates we used the default parameters. For the identification of enriched gene ontology terms in our study, we used the GSEApy package V1.0.4 with its default parameter settings.

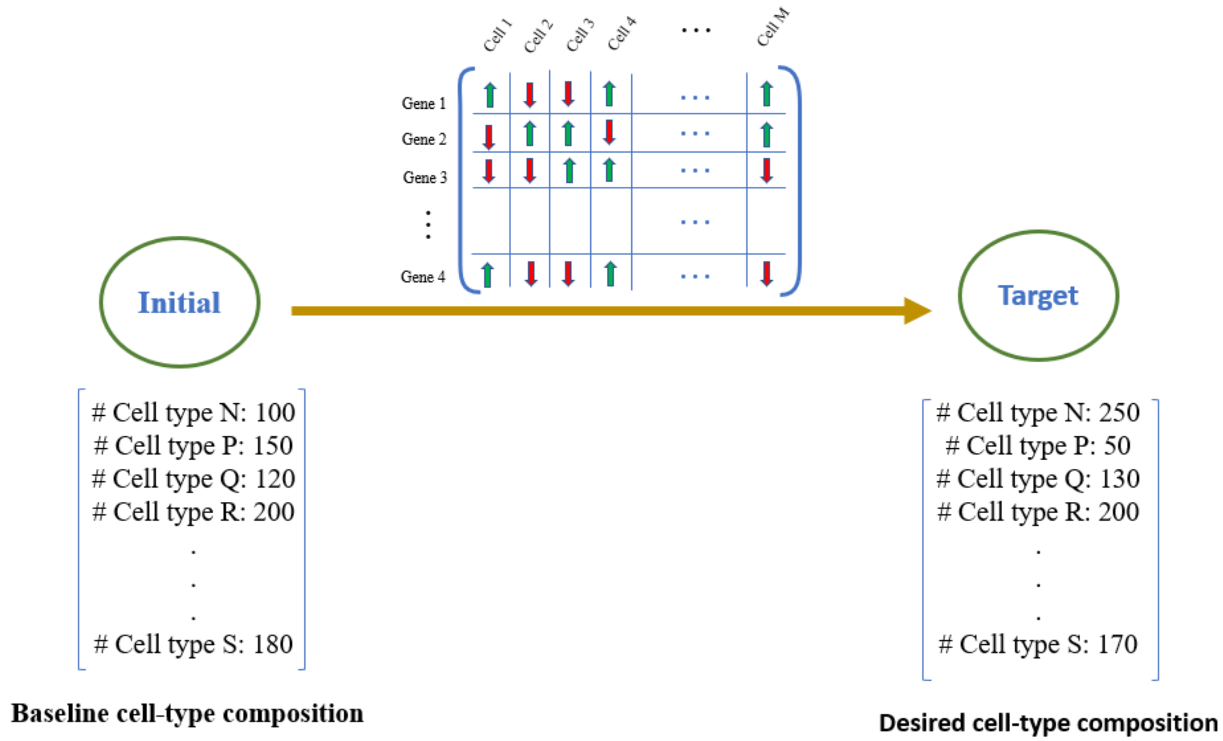
Cell Reports Methods, Volume *4*

Supplemental information

Fatecode enables cell fate regulator prediction using classification-supervised autoencoder perturbation

Mehrshad Sadria, Anita Layton, Sidhartha Goyal, and Gary D. Bader

Supplementary information

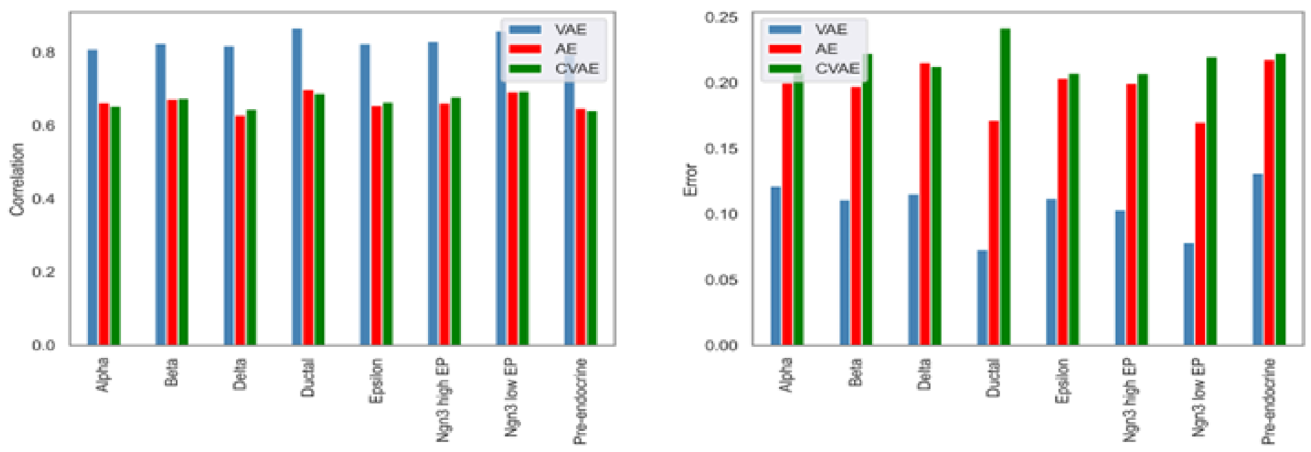


	Cell 1	Cell 2	Cell 3	Cell 4	…	Cell M
Gene 1	↑	↓	↓	↑	…	↑
Gene 2	↓	↑	↑	↓	…	↑
Gene 3	↓	↓	↑	↑	…	↓
⋮					…	
Gene 4	↑	↓	↓	↑	…	↓

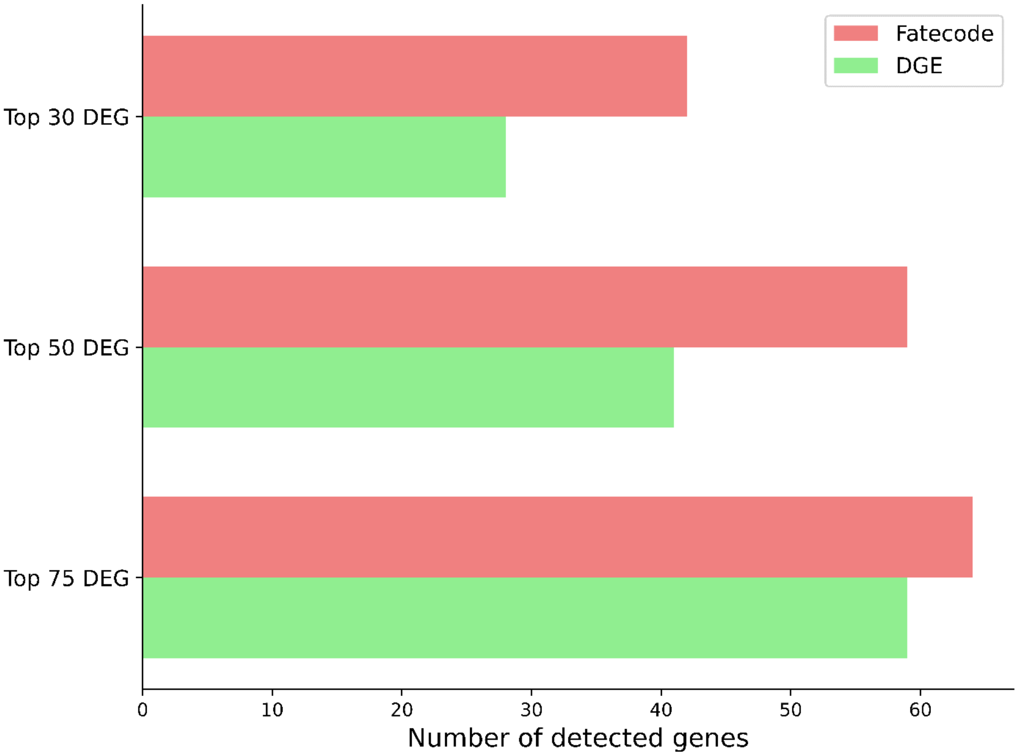
Cell type N: 100
Cell type P: 150
Cell type Q: 120
Cell type R: 200
.
.
.
Cell type S: 180

Cell type N: 250
Cell type P: 50
Cell type Q: 130
Cell type R: 200
.
.
.
Cell type S: 170

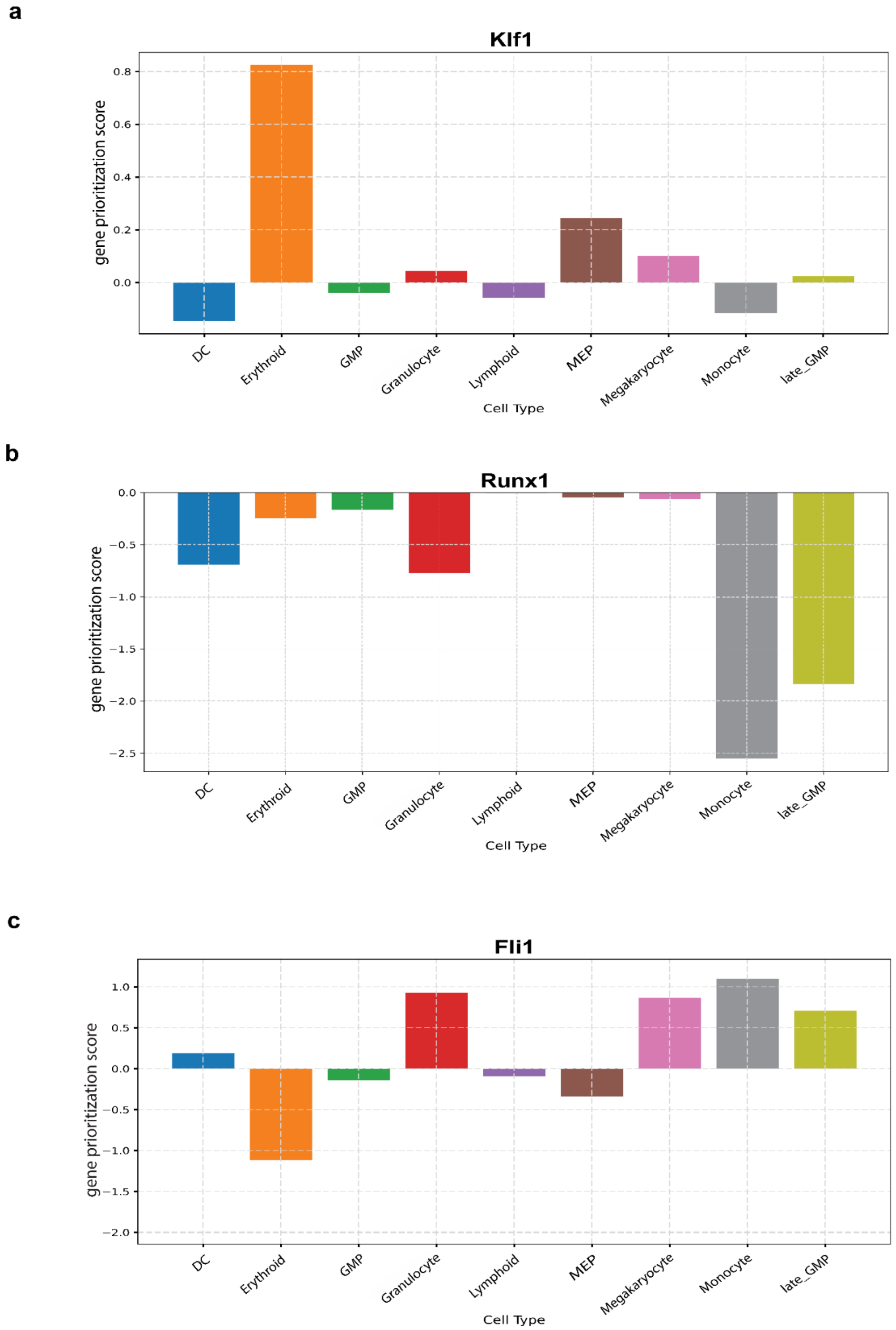
Supplementary Fig. 1: Gene expression perturbations to change the cell-type distribution This figure demonstrates the transition of an initial cell type distribution to a desired target distribution through gene-level adjustments. The circles represent the system's state, with the frequencies of cell types indicated below.



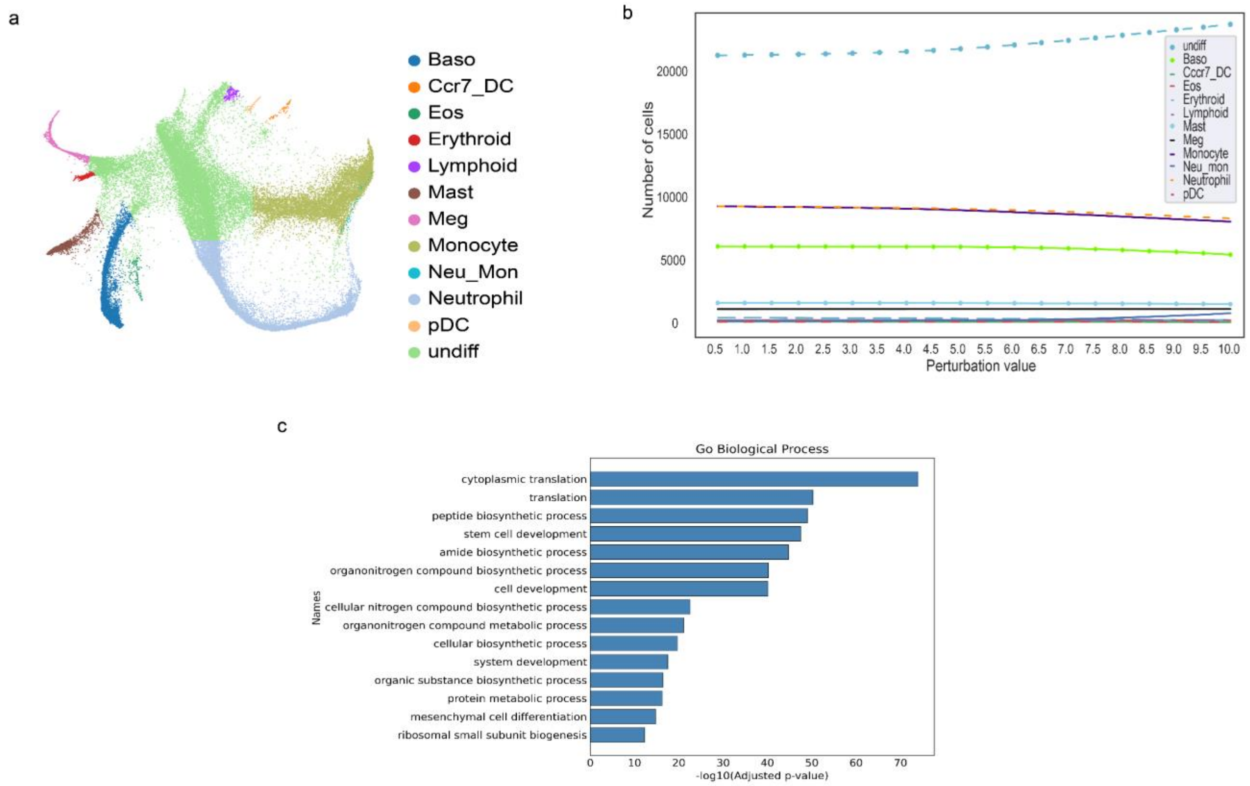
Supplementary Fig. 2: Comparison of autoencoder architectures for analyzing data for endocrine development in the mouse pancreas a, comparison of the input-output correlation for the AE, the variational autoencoder (VAE), and the conditional variational autoencoder (CVAE). b, the mean square error of the three autoencoder architectures' input and output. VAE performs better than other architectures for this data.



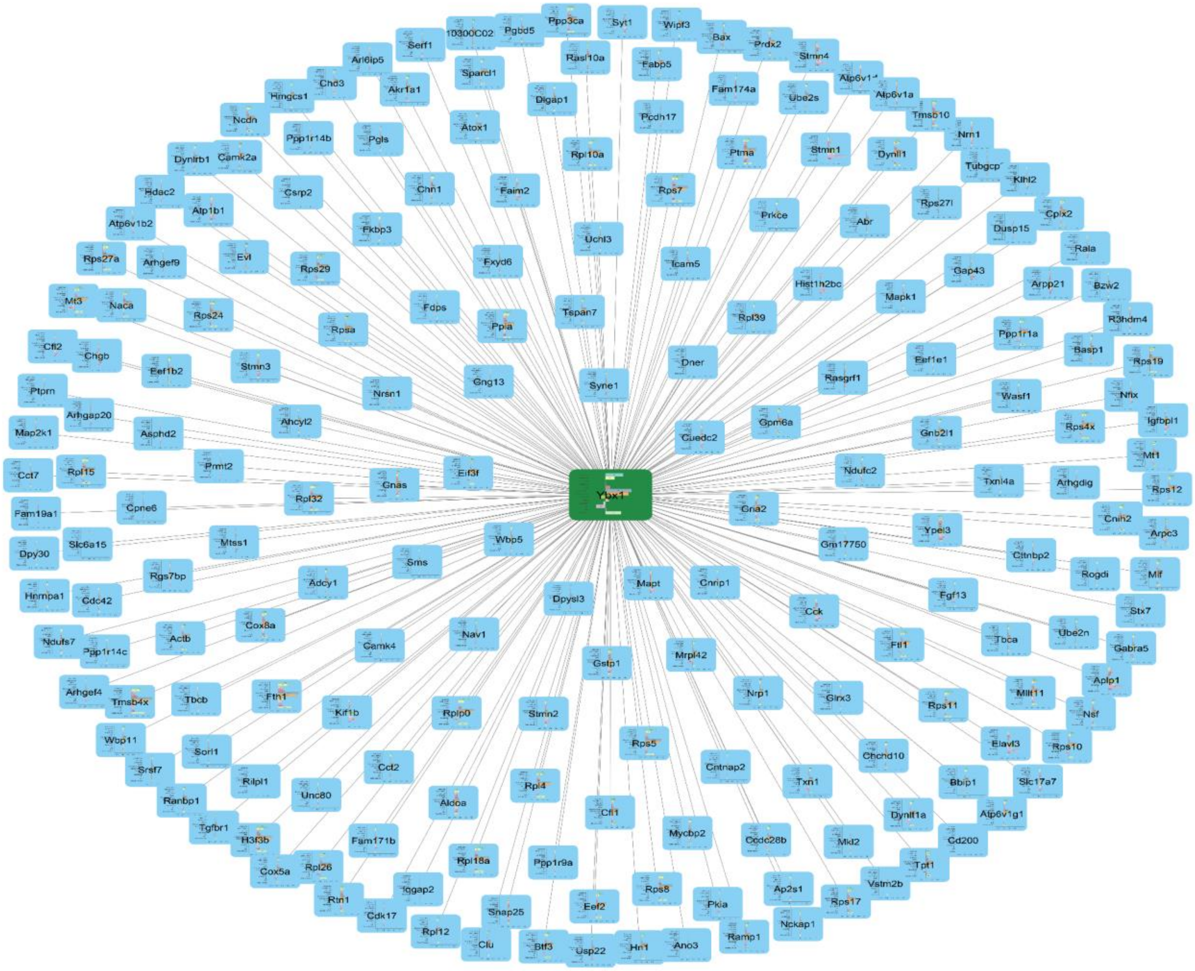
Supplementary Fig. 3: Performance comparison of Fatecode and DGE in detecting predefined gene regulators across different DEG thresholds. The bar plot shows the comparative performance of Fatecode (red) and DGE (green) in detecting known gene regulators. Performance is evaluated based on the number of top differentially expressed genes (DEGs) considered for analysis (30, 50, and 75). The height of each bar represents the accuracy of each method in identifying these predefined key regulators.



Supplementary Fig. 4: Gene prioritization score for the mouse hematopoiesis data.
a, b, c, Fatecode accurately determines gene prioritization scores for various genes, including *Klf1*, *Runx1*, and *Fli1*, across different cell types.



Supplementary Fig. 5: Fatecode analysis of hematopoiesis data identifies master regulators governing cell switching dynamics. a, visualization of the hematopoiesis dataset from Weinreb et al. hematopoietic progenitors differentiate into different cell types such as mast cell (Ma), basophil (Ba), eosinophil (Eos), megakaryocyte (Mk), lymphoid precursor (Ly), migratory dendritic cell (mDC) and plasmacytoid dendritic cell (pDC), erythrocyte (Er), neutrophil (Neu), monocyte (Mo). b, The effect of different perturbation sizes of a node in the latent layer on the cell distribution. c, gene set enrichment analysis results. Gene ontology (GO) biological processes enrichment analysis shows significant process terms related to mouse hematopoiesis, stem cell development, and mesenchymal cell differentiation.



Supplementary Fig. 6: Gene Regulatory Network of Ybx1 and its downstream target genes along with gene prioritization scores. The GRN was constructed using SCENIC, by filtering the top 2000 interactions with the highest SCENIC Importance Measure (IM) scores. Additionally, the top 400 predicted master regulators from Fatecode were mapped onto the GRN, and the resulting network is presented here using a network bar chart in which each value of a bar plot shows the Fatecode gene prioritization score of the gene for that cell type.

Supplementary Note 1- Hyperparameter search

Hyperparameter tuning was conducted using a grid search approach. We explored various combinations of hyperparameters, including learning rate, batch size, number of layers, number of neurons per layer, and activation functions. The hyperparameter space for each parameter was defined as follows:

- Autoencoder type: [AE, VAE, CVAE]
- Activation function: [LeakyReLU, Relu, linear]
- Learning rate: [0.001, 0.01, 0.1]
- Batch size: [400, 500, 600]
- Number of hidden layers (encoder): [1, 2]
- Number of neurons in latent_layer: [input_size/40, input_size/60, input_size/80, input_size/100, input_size/125, input_size/150]