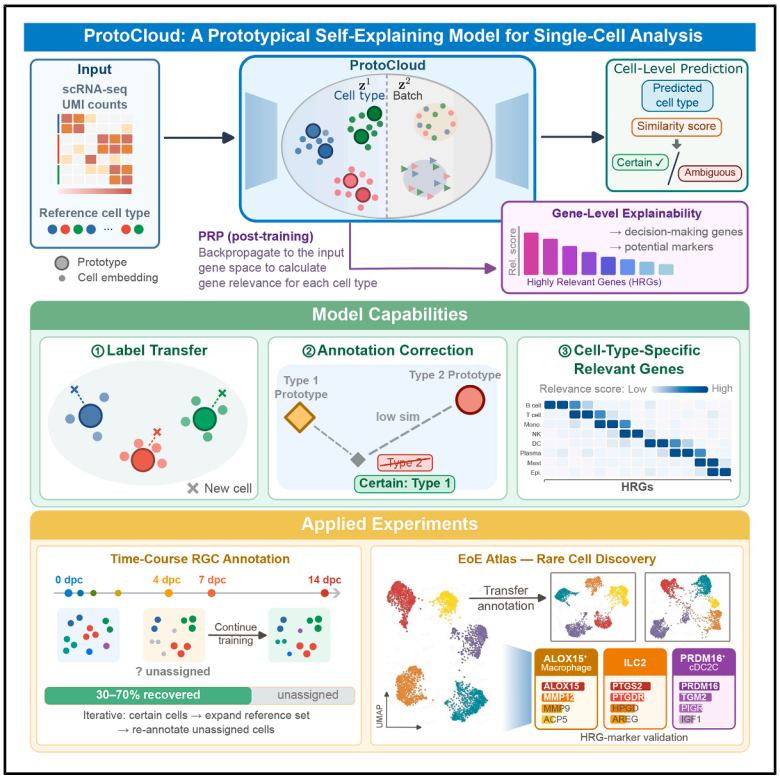
Cell Genomics

Technology

ProtoCloud: A prototypical self-explaining model for single-cell analysis

Graphical abstract



Highlights

- Accurate and efficient annotation of single-cell data, including rare cell types
- A disentangled latent space separates biological identity from other variations
- Built-in uncertainty estimates identify and correct potential misannotations
- Instant explainability reveals genes driving cell type classification

Authors

Kaiyun Guo, Jiarui Ding

Correspondence

jiarui.ding@ubc.ca

In brief

Guo et al. introduce ProtoCloud, a deep learning model for cell type annotation of individual cells. By embedding cells around prototypes, the model enables transparent classification, identifies genes driving cell type decisions, and quantifies prediction uncertainty. ProtoCloud accurately detects rare cell states and corrects annotation errors, helping construct reliable cellular atlases.

Guo & Ding, 2026, Cell Genomics 6, 101217
June 10, 2026 © 2026 The Author(s). Published by Elsevier Inc.
https://doi.org/10.1016/j.xgen.2026.101217

CellPress

Cell Genomics

CellPress
OPEN ACCESS

Technology

ProtoCloud: A prototypical self-explaining model for single-cell analysis

Kaiyun Guo[1] and Jiarui Ding[1,2,*]
[1]Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada
[2]Lead contact
*Correspondence: jiarui.ding@ubc.ca
https://doi.org/10.1016/j.xgen.2026.101217

SUMMARY

Cell type annotation is a fundamental task in single-cell genomics. Although various methods have been developed for automatic annotation, they often function as black-box models lacking explainability, proper uncertainty estimation, and robustness for rare cell types. We introduce ProtoCloud, a self-explanatory deep generative model that embeds cells into a structured, low-dimensional space organized around cell-type-specific prototypes. ProtoCloud matches or outperforms existing methods across 11 large-scale datasets, particularly for rare cell types. Its built-in uncertainty quantification mechanism, based on cell-prototype similarity, identifies and re-annotates misannotated training cells. By backpropagating cell prototype similarities to the gene space, ProtoCloud identifies key genes driving its classifications, facilitating the discovery of both known and novel marker genes. Applied to a time-course dataset of post-injury retinal neurons, ProtoCloud successfully annotates previously unassigned cells; in an esophageal cell atlas, it identifies rare but potentially important cell populations and their marker genes associated with esophageal inflammation.

INTRODUCTION

In recent years, technological advancements in single-cell genomics, especially single-cell and single-nucleus RNA sequencing (scRNA-seq and snRNA-seq),[1–7] have enabled the construction of comprehensive tissue atlases,[8–11] each encompassing hundreds of thousands to millions of cells. These atlases provide detailed insights into tissue cellular composition and intercellular interactions,[12–14] cell-type-specific and shared biological processes,[15–17] in development,[18–20] homeostasis,[21] aging,[22] inflammation, and disease.[12,23–25] Computational methods have been developed to map newly sequenced cells to these reference atlases, facilitating rapid cell type annotation and accelerating biological discoveries.[8,26–28] This supervised, reference mapping approach contrasts with unsupervised clustering strategies that rely on either manual annotation[29] or automated marker gene scoring[30] for cell type assignment and remain indispensable in settings where annotated reference datasets are unavailable.

Among the reference-mapping approach, deep learning models have emerged as powerful tools for analyzing single-cell datasets.[26,31–40] Because of their scalability in processing large-scale datasets, flexibility in handling different data modalities with potential experiment-specific nuisance factors, and effectiveness in analyzing complex, high-dimensional data, deep learning models are well-suited for modeling scRNA-seq datasets. Despite these successes, these models often function as black boxes, excelling at predictions but offering limited interpretability regarding their decision-making process.[41] For large-scale cell type annotation, an ideal deep learning model should be self-explanatory, offering not only accurate predictions but also clear insights into its decision-making process.[42] For example, if a cell is predicted to be a B cell, a self-explaining model should provide dual explainability: (1) gene-level features highlighting key expressed genes driving the decision and (2) cell-level similarities to reference B cell populations or B cell prototypes. Such transparency is crucial for assessing prediction uncertainties and building reliable cell atlases.

Single-cell data presents additional challenges that complicate accurate cell type annotation, even with deep learning methods. First, tissues typically consist of dominant and rare cell types, complicating the identification of rare cell populations.[25,43] The origins of these rare cell types may be biological or technical. For instance, basophils play essential roles in allergy, but they account for less than 1% of circulating blood cells.[44] Conversely, eosinophils are enriched in the diseased human esophagus, yet these cells are frequently undetected in single-cell studies due to technical limitations in conventional scRNA-seq pipelines.[45–47] Second, cell-type-irrelevant nuisance factors introduce another layer of complexity, further hindering accurate cell type annotation and interpretation.[48,49] While specialized tools have been developed to mitigate these nuisance factors,[32,50] identifying which specific factors require correction in a given dataset remains challenging.

Motivated by these gaps, we introduce ProtoCloud, a self-explanatory deep learning model that achieves state-of-the-art performance in single-cell cell type annotation with both cell-level and gene-level explainability. The model embeds cells around cell-type-specific prototypes in a low-dimensional latent

This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

space, quantifying prediction uncertainty through distance-based similarities between cell embeddings and the closest prototypes of the predicted cell types. By backpropagating cell-prototype similarities in the latent space to the observed gene space, it highlights the most relevant genes that drive the model's decisions. Furthermore, ProtoCloud partitions its latent space into two components that isolate cell type information from cell-type-irrelevant factors.[51,52] To improve performance on rare cell types, the model employs a unique data augmentation technique that models gene counts using multinomial distribution-based sampling. Through its model architecture, loss functions, and training strategy, ProtoCloud effectively captures the structures of complex high-dimensional scRNA-seq data in the latent space, resulting in accurate and robust predictions with meaningful biological insights.

Across multiple datasets, ProtoCloud performs favorably compared to several widely used scRNA-seq annotation tools. It also successfully disentangles cell types from technical batch effects in the latent embeddings. We further demonstrate its effectiveness by re-annotating potentially misannotated cells in a widely used peripheral blood mononuclear cell dataset.[3,32] In a time-course study of retinal ganglion cells (RGCs) following acute injury,[53] ProtoCloud successfully annotates previously unassigned cells and outperforms competing methods. Additionally, when trained on an esophageal mucosal cell atlas[25] (421,312 cells, 72 cell types) to annotate cells from two additional studies, ProtoCloud identifies disease-associated rare cell populations, including the novel *PRDM16*$^{+}$ dendritic cells in both test cohorts. Collectively, these results highlight ProtoCloud's versatility in refining reference cell type annotations, identifying rare cell populations, and supporting the construction of comprehensive cell atlases, thereby establishing it as a valuable tool for single-cell data analysis.

DESIGN

ProtoCloud (Figure 1A) is a self-explanatory deep generative model that embeds high-dimensional gene expression profiles of individual cells into a structured, low-dimensional latent space organized around cell-type-specific prototypes. Specifically, each cell type is associated with a set of prototypes (six by default) that also reside in the latent space. These prototypes are designed to capture (1) inter-cell type variation, ensuring that prototypes of the same cell type are proximal to each other while those of different types are well separated, and (2) intra-cell type variation, enabling the prototypes of a given cell type to capture its inherent heterogeneity. For cell type annotation, ProtoCloud achieves interpretability at two levels. To achieve cell-level interpretability of a prediction, cell type classification is based on the similarity of a cell's embedding to these prototypes (see STAR Methods). For gene-level interpretability of a prediction, ProtoCloud applies prototypical relevance propagation (PRP)[54–56] to the trained prototypes of the predicted class to identify decision-driving genes. For each cell, it assigns a relevance score to each gene by backpropagating the initial relevance, which is positive for the prototypes of the cell type to which it is assigned and negative for other prototypes, through the encoder network to the input space (Figure 1B and STAR Methods). Unlike other post hoc explanation methods, such as Shapley additive explanations (SHAP)[57] that rely on computationally expensive external perturbations, PRP is a model-inherent approach that directly backpropagates cell-to-prototype similarity scores through the encoder network. Importantly, this design requires only a single backward pass after training to compute the gene relevance used for classification across all cell types.

ProtoCloud is built upon a variational autoencoder (VAE) architecture trained end-to-end for single-cell genomics data analysis.[58,59] Designed for ease of use with minimal preprocessing requirements, ProtoCloud operates directly on raw unique molecular identifier (UMI) counts. During the supervised training phase, the model utilizes a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ consisting of N individual cells. In this formulation, $\mathbf{x}_i$ is the gene expression vector of G genes in cell i, and y_i denotes the corresponding cell type label. Unlike existing annotation methods, which typically function as black-box classifiers, ProtoCloud classifies each cell based on its similarity to learned prototypes in the latent space, providing intuitive cell-level interpretability. Importantly, the maximum similarity between a cell embedding and its corresponding prototypes provides a robust measure of prediction uncertainty, unlike softmax probabilities commonly output by classifiers, which tend to saturate near extreme values regardless of the actual reliability of the prediction.

A key design choice in ProtoCloud is the partitioning of the latent space of each cell into two components with distinct roles. The first half is dedicated to encoding cell type identity, on which the prototypes and classifier operate exclusively. The second half captures cell-type-irrelevant variations, such as batch effects, and is regularized toward a standard normal prior. This architectural separation ensures that the prototypes represent biological cell type identity rather than technical artifacts, without requiring explicit batch information. However, training this architecture with a standard VAE objective may lead to latent space fragmentation, where cells of the same type are scattered into disconnected regions due to batch effects. To address this issue, ProtoCloud employs a two-stage training curriculum. In the first stage, only the VAE negative evidence lower bound loss and the classifier cross-entropy loss are used, allowing embeddings within each class to converge into cohesive regions while channeling batch information into the second latent half. In the second stage, an orthogonal loss and an atomic loss are introduced to encourage prototype diversity. To mitigate class imbalance, which is common in single-cell datasets, ProtoCloud augments rare populations by sampling from a multinomial distribution parameterized by each cell's count profile. Full algorithmic and mathematical details are provided in the STAR Methods section.

By using consistent hyperparameters across all datasets, we demonstrate the robustness and versatility of ProtoCloud across diverse biological contexts. We highlight three major applications of ProtoCloud in scRNA-seq data analysis: (1) reliable label transfer between datasets (Figure 1C), (2) detection and correction of annotation anomalies to improve data quality (Figure 1D), and (3) identification of candidate cell type marker genes (Figure 1E).

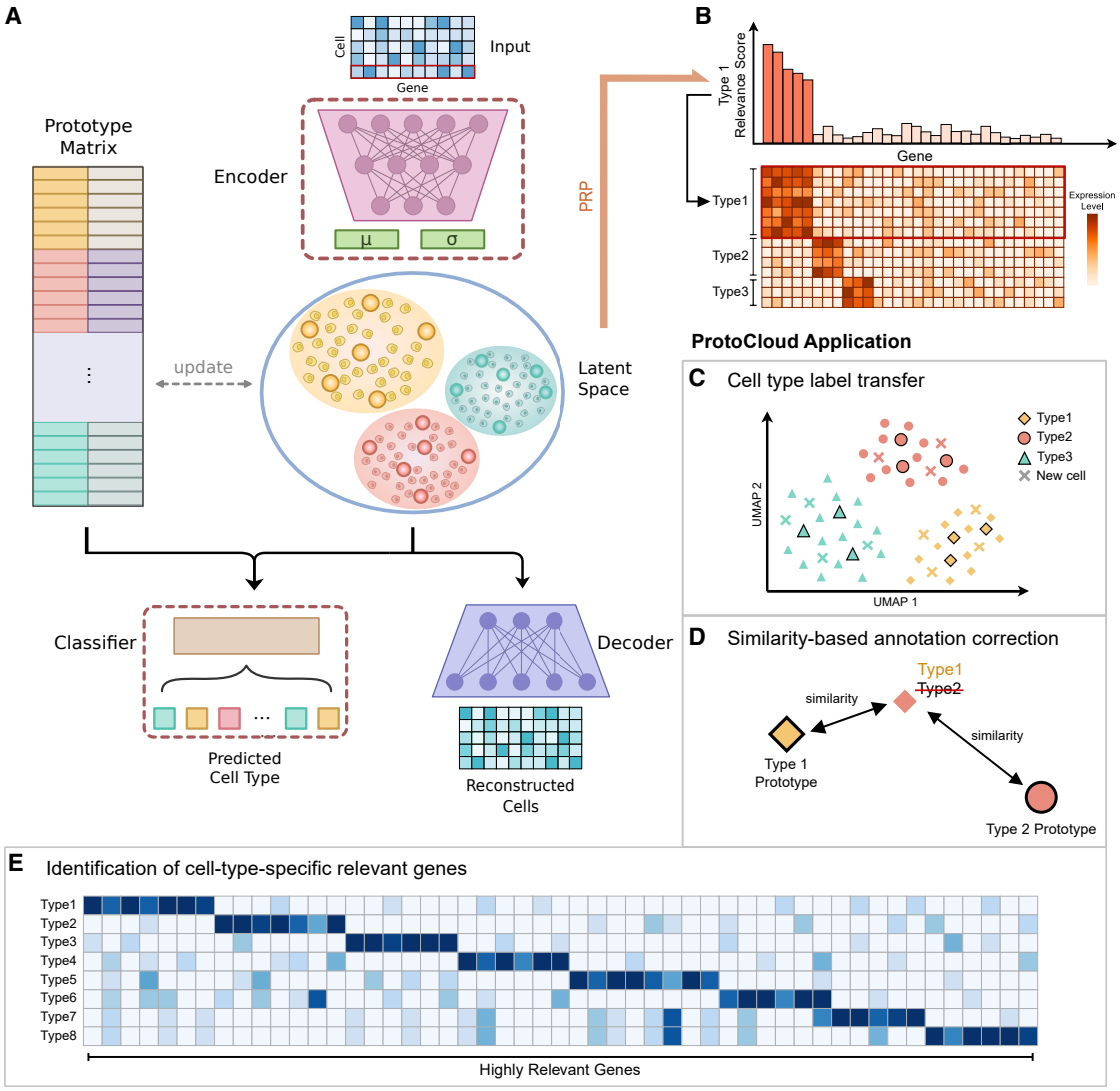


Figure 1. ProtoCloud overview

(A) An illustration of the ProtoCloud model. The model has four major components: a probabilistic encoder, a probabilistic decoder, a prototype matrix, and a linear classifier. The model takes only raw UMI counts (cell-by-gene count matrix) as input during inference. Cell inputs are encoded into a low-dimensional latent space (d_z = 20 by default), with different colors indicating cell types. Larger points in the latent space represent prototypes, which are pre-initialized (six per cell type) and share the same latent space as the cell embeddings. Cell type information (brighter color) is encoded in the first half of the latent dimensions. The latent embeddings are used for both cell type prediction, based on similarity to the prototypes, and for reconstructing the gene expression through the decoder.

(B) ProtoCloud provides inherent interpretability through prototypical relevance propagation (PRP), which automatically generates gene-level relevant scores once the model is trained. Each prototype undergoes PRP to produce gene-level relevance scores. Genes with higher relevance scores are the decision-relevant genes of the corresponding cell type.

(C–E) Applications of ProtoCloud. (C) The model enables accurate and robust transfer of cell type annotations across datasets. Shapes denote ground truth cell types, and colors indicate predicted labels. Symbols with black edges represent prototypes, while those without edges are cell embeddings. Crosses mark newly added query cells predicted to belong to the corresponding cell types (indicated by color). (D) ProtoCloud's similarity-based classification process enables the detection of anomalous annotations, improving the quality of the ground truth annotations. A cell with ground truth type 1 (diamond shape) was initially mislabeled as type 2. ProtoCloud corrects this annotation by assigning the cell to type 1 based on a higher similarity score. (E) The identified cell-type-relevant genes can act as candidate markers and provide molecular insights into cell identities.

RESULTS

Accurate and robust cell type annotation

To evaluate ProtoCloud's performance in cell type annotation, we performed a comparative analysis against eight state-of-the-art methods, including two probabilistic machine learning approaches (Seurat[27] and CellTypist[8]), four deep learning models (scANVI,[26] TOSICA,[60] scPoli,[61] and SIMS[62]), and two NLP-inspired foundation models (scGPT[39] and scBERT[37]). We benchmarked these methods across eight diverse datasets

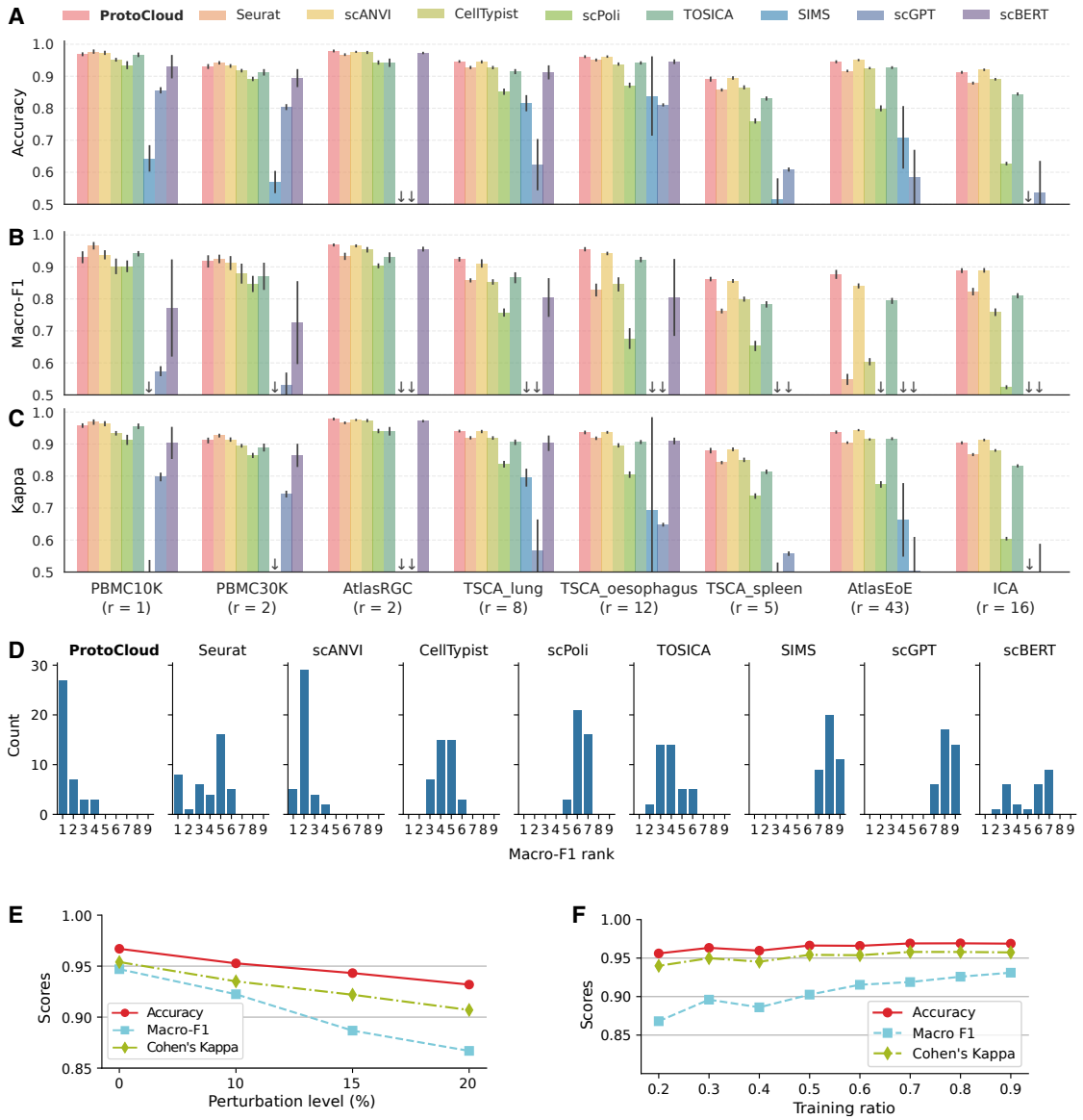


Figure 2. Performance of ProtoCloud on cell type classification

(A–C) Benchmarking of cell type annotation performance. We compared ProtoCloud against Seurat V4,[27] scANVI,[26] CellTypist,[8] scPoli,[61] TOSICA,[60] SIMS,[62] scGPT,[39] and scBERT[37] for cell type annotation across eight datasets, ranging from approximately 10,000 to 400,000 cells. The *x* axis represents datasets, with the numbers in the parentheses indicating the number of rare cell types in each dataset. Downward-pointing arrows indicate values below 0.5. Error bars: standard error of the mean. (A) Evaluation metrics include accuracy, (B) macro F1 score, (C) and Cohen's kappa coefficient. The metrics were averaged over five random seed experiments, with 80% of the data used for training and 20% for validation in each run.

(D) Macro F1 score rank distributions across methods and datasets over all experimental repetitions.

(E and F) Model performance analysis under varying conditions using the PBMC10K dataset.[3] (E) Validation accuracy as the proportion of label perturbation in the training set increases from 0% to 20%. (F) Validation accuracy as training data ratios decrease, ranging from 0.8 to 0.1.

spanning multiple species, organs, and sequencing technologies[3,8,25,53,63,64] (STAR Methods). Model performance was assessed across multiple random seeds, with average accuracy, macro F1 score, and Cohen's kappa coefficient[65] reported as summary metrics (STAR Methods). Using default hyperparameters across all datasets, ProtoCloud exhibited strong and consistent performance, often matching or surpassing the baselines across datasets and evaluation metrics (Figures 2A–2C; Table S1; STAR Methods). Other state-of-the-art methods, such as scANVI, also showed high accuracy but require batch information as input. Importantly, ProtoCloud demonstrated notably higher accuracy for low-abundance cell types, as reflected in the macro F1 score (Figure 2D; Table S2). These results underscore ProtoCloud's ability to achieve robust and reliable

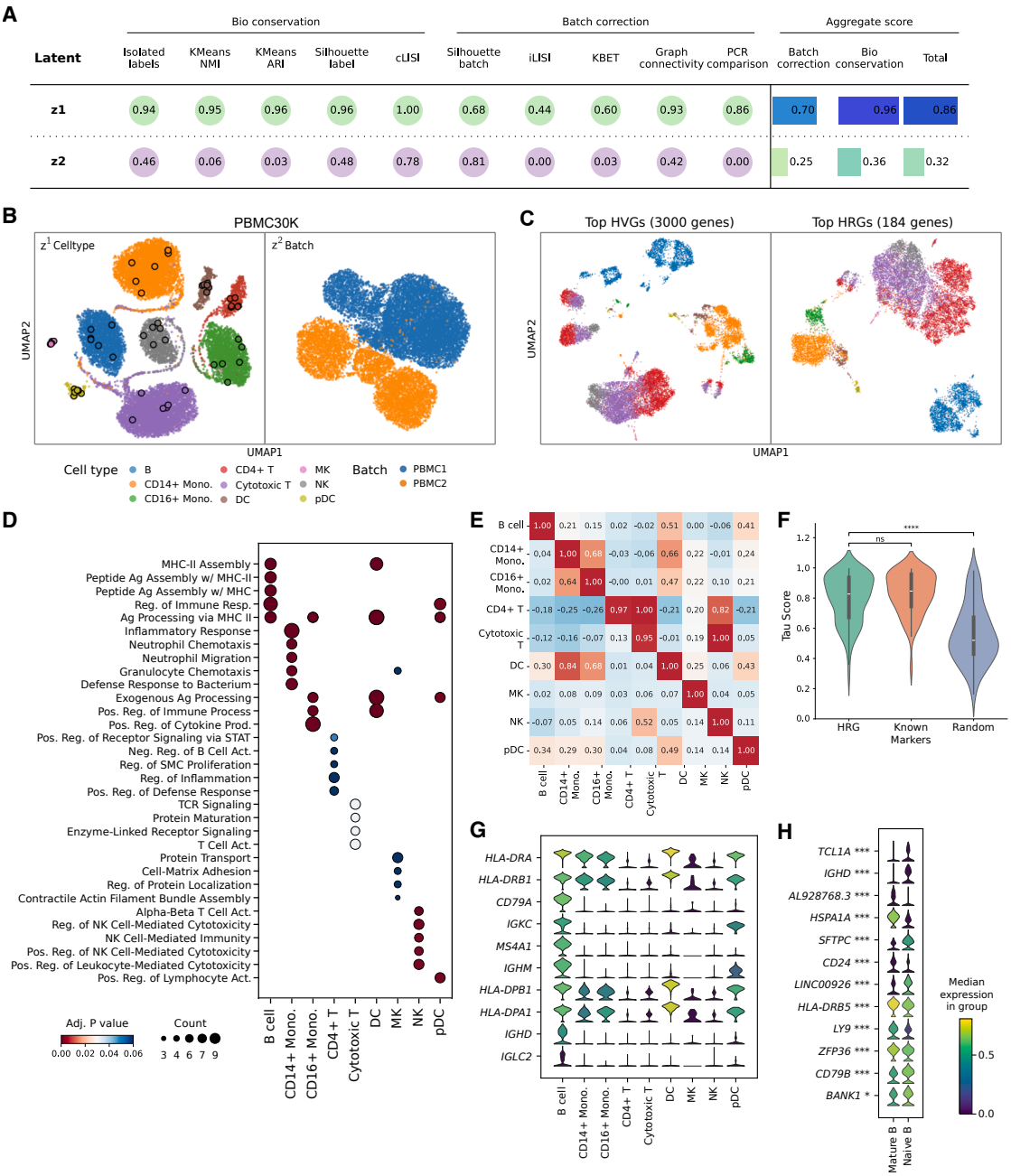


Figure 3. Latent space organization and HRG verification in ProtoCloud

(A) Evaluation of latent space disentanglement using the PBMC30K[63] dataset. Comparison of single-cell integration benchmarking (scIB) metrics for the biological subspace (first-half latent space, $\mathbf{z}^1$) and the batch subspace (second-half latent space, $\mathbf{z}^2$). Metrics are grouped into biological conservation and batch correction categories. A score closer to 1 indicates better performance.

(B) Visualization of ProtoCloud latent representations of PBMC30K. UMAP projections show cell-type-specific clustering in $\mathbf{z}^1$ (left) and batch effects in $\mathbf{z}^2$ (right) of the latent space. Prototypes are represented by dots with black outlines.

(C) Visualization of ProtoCloud latent representations of PBMC30K. UMAP projections show the representation using all 3,000 HVGs (left) compared to the representation using the union of the top 30 HRGs of each cell type (right), resulting in 184 unique genes in total.

(D) Gene Ontology (GO) biological process enrichment analysis of HRGs across cell types in PBMC30K. The dot plot displays the top enriched pathways for each cell type. Dot size represents the number of HRGs associated with the pathway, and color indicates statistical significance.

(E) Cell type specificity of HRGs in PBMC30K. Heatmaps comparing row-normalized gene signature scores based on the top 15 HRGs across nine cell types. Diagonal elements (matching cell types) represent on-target specificity, where high values indicate relatively high expression in the respective cell type, while off-diagonal elements represent off-target expression.

(legend continued on next page)

classification performance, even for rare cell types. As a deep learning model trained with mini-batches, ProtoCloud is scalable to process large datasets (Figure S1A; STAR Methods).

To evaluate ProtoCloud's robustness, we assessed its performance under varying conditions using the peripheral blood mononuclear cell (PBMC) dataset.[3,32] We introduced perturbations by randomly shuffling different proportions of the training labels (i.e., 10%, 15%, and 20%). Even under these label perturbations, the model maintained high validation accuracies above 93.18% (Figure 2E), demonstrating its resilience to label noise. We further investigated ProtoCloud's performance under limited training data scenarios by varying the training data ratio from 0.10 to 0.80. The model maintained stable performance across these training set sizes, with accuracy ranging from 95.34% to 97.64% (Figure 2F). Notably, even when trained on only 10% of the data, the macro F1 score remained high at 0.87, indicating robust performance even with limited training data. The consistently strong performance and robustness demonstrated by ProtoCloud in cell type classification tasks underscore its reliability in scRNA-seq analysis and form the basis for its robust explainability.

Batch-separated informative latent space

Despite not requiring batch information as input, ProtoCloud's two-stage curriculum and structured latent space design (STAR Methods) effectively separate and capture batch-related variability in its latent subspace. To assess this separation in the latent space, we used two datasets with known batch annotations: PBMC30K[63] and AtlasRGC.[53]

We assessed the information content of the separated latent spaces by calculating biological conservation and batch mixing metrics (Figure 3A; STAR Methods). Consistent with our design, the first component, $\mathbf{z}^1$, is highly enriched for biological information, as evidenced by a near-perfect cell type local inverse Simpson's index (cLISI) of 1.00 and high clustering evaluation metrics (e.g., the normalized mutual information metric NMI = 0.95). Conversely, the second half of the latent embedding ($\mathbf{z}^2$) shows substantially lower biological conservation (e.g., cLISI = 0.78 and NMI = 0.06) but preserves batch effects, with an integration local inverse Simpson's index (iLISI) of zero, compared to 0.44 from $\mathbf{z}^1$. These trends are consistent across additional evaluation metrics (e.g., the *k*-nearest-neighbor batch-effect test [kBET] and Graph connectivity), with the exception of the silhouette batch metric, which is known to be unreliable for assessing batch correction.[66]

To visualize the separation of information encoded in different segments of the latent space, we used uniform manifold approximation and projection (UMAP)[67] to project both the first and the second halves of the latent space into two dimensions. Similarly, the UMAP of the first half of the latent space ($\mathbf{z}^1$) shows clear separation by cell types (Figure 3B), but no apparent clustering by batch (Figure S1B), whereas the UMAP of the second half of the latent space ($\mathbf{z}^2$) reveals predominant grouping of cells by batch, with mixed cell type composition within these batch groups. Direct visualization of the latent space reveals that batch effects are confined to specific dimensions of the latent space (Figure S1C). The consistent pattern is also observed in the AtlasRGC dataset, with cells grouped by type in the first half of the latent space and by batch in the second half of the latent space (Figures S1D–S1F). This clear dimensional disentanglement suggests that different components of the latent space effectively encode distinct technical (batch) and biological (cell type) sources of variation.

We performed ablation studies to evaluate the impact of different training strategies on latent space organization (STAR Methods). On the UMAP of the latent representation obtained from the model with a non-separated latent space, marked batch effects were observed with a low batch entropy score (BES) of 0.014 (Figure S2A; STAR Methods). We then evaluated the impact of our two-stage curriculum training strategy compared to a single-stage end-to-end approach. While both strategies maintained high biological conservation in $\mathbf{z}^1$, the two-stage approach demonstrated superior disentanglement capabilities (Figure S2B). Specifically, the two-stage training yielded a significantly higher aggregate batch correction score of 0.78 for $\mathbf{z}^1$, compared to 0.67 for the single-stage model. Next, we dissected the contribution of each loss component (Figure S2C). The baseline VAE configuration (without the atomic and orthogonal losses; STAR Methods) resulted in collapsed prototypes with intra-class similarities approaching 1.0, indicating insufficient within-class diversity. Adding only the atomic loss improved cell type separation (decreased inter-class similarity), although prototypes remained overly compact (high intra-class similarity). Incorporating only the orthogonal loss improved (reduced) intra-class similarity to 0.953 but increased inter-class similarity to 0.218, compared to 0.130 in the standard configuration. These results confirm that the complete loss formulation in our standard setting learns prototypes that better capture both between-cell type separation and within-cell type diversity (Figure S2D). Together, these ablation studies demonstrate that each auxiliary component contributes to ProtoCloud's overall performance.

Gene-level explainability of cell predictions

ProtoCloud employs PRP (STAR Methods)[54–56] to identify the key genes driving its classification decisions. This approach provides a direct, data-driven justification for the model's predictions.

(F) Quantitative comparison of gene specificity across three gene sets in PBMC30K. Violin plots display the distributions of Tau specificity scores for HRGs (top 20 per cell type, $n = 120$), canonical marker genes ($n = 69$), and randomly selected genes ($n = 100$). HRGs exhibit cell type specificity comparable to known markers, achieving a median Tau score of 0.828, while randomly selected genes show significantly lower specificity. Statistical significance was assessed using two-sided Mann-Whitney U test (ns, non-significant with $p > 0.05$ and ****$p \leq 0.0001$).

(G) Expression profiles of the top ten B cell-specific HRGs in PBMC30K. The *y* axis lists the top ten B cell-specific HRGs in descending rank of relevance. Expression distributions of these top ten B cell-specific HRGs across cell types are shown. High expression in B cells (left column) contrasted with low expression in other cell types demonstrates strong cell type specificity.

(H) Expression profiles of the top differentially ranked HRGs between mature and naive B cells in the TSCA lung[64] dataset. Differential expression significance was assessed using the Wilcoxon rank-sum test with Benjamini-Hochberg correction, denoted by asterisks (*$p \leq 0.05$ and ***$p \leq 0.001$).

CellPress
OPEN ACCESS

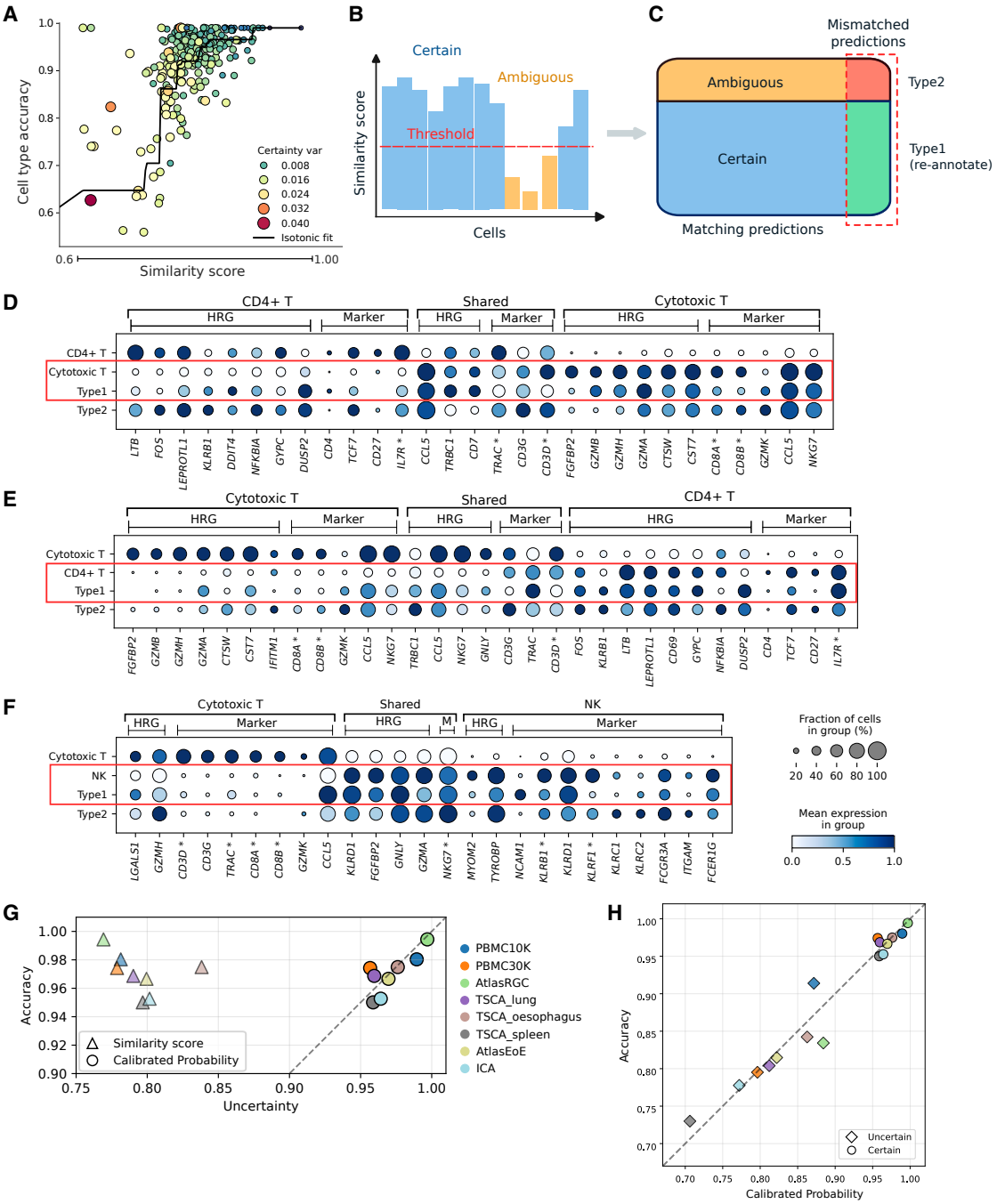


Figure 4. Certainty-based annotation refinement in ProtoCloud

(A) The relationship between per-cell-type accuracy and similarity scores across eight experimental datasets. The *x* axis denotes the similarity score of each cell type, and the *y* axis denotes its prediction accuracy. Each point represents a distinct cell type from one of the experimental datasets. Point size and color intensity jointly encode the variance of similarity scores for each cell type. The black curve represents an isotonic regression fit, demonstrating the positive correlation between similarity score and accuracy.

(B and C) Workflow for annotation refinement in ProtoCloud. (B) Step 1: assign a confidence score by classifying each cell prediction as "certain" (blue) or "ambiguous" (orange) using class-specific thresholds derived from the training data. Cells with a similarity score above the threshold are classified as "certain," while those below the threshold are categorized as "ambiguous." (C) Step 2: if the original annotations are available, we can re-annotate confidently predicted cells by comparing predictions with the original annotations. Type 1 cells are cells with high prediction confidence that do not match the original labels (green) and are therefore re-annotated. Type 2 cells (red) are ambiguous cells and have unaligned labels that will not be re-annotated.

(D–F) Expression patterns of major type 1 annotation pairs in PBMC30K, comparing original annotations (first row), predicted types (second row), type 1, and type 2 cells. Type 1 cells should align more closely with the predicted cell type than with the original annotation. Type 2 cells remain ambiguous, showing limited

(legend continued on next page)

Genes assigned high relevance scores are referred to as highly relevant genes (HRGs). Users can leverage HRGs for downstream analyses instead of relying only on prior knowledge of established cell type markers,[68–70] ensuring broad applicability across diverse biological contexts (Figure S3A).

To validate the representational power of HRGs, we visualized the UMAP using only the union of the top 30 HRGs of each cell type and compared it against the baseline constructed from the top 3,000 highly variable genes (HVGs; Figure 3C). Despite utilizing approximately 16-fold fewer genes, the HRG-based embedding successfully preserved the global topology of the data (e.g., the continuum arrangement of $CD4^+$ T cells to cytotoxic T cells, and then to natural killer [NK] cells in the UMAP; similarly, $CD14^+$ monocytes and $CD16^+$ monocytes are close to each other). We next assessed the biological validity of these signatures through Gene Ontology (GO) enrichment analysis (Figure 3D). The enriched pathways exhibited high lineage specificity, strictly aligning with known cellular functions. For instance, $CD14^+$ monocytes were enriched for inflammatory response and defense response to bacteria, consistent with their primary phagocytic role. In contrast, $CD16^+$ monocytes were distinctively characterized by immune regulation and antigen presentation pathways. These results align with the established roles of classical monocytes as first responders to bacterial infection and non-classical monocytes as patrolling immune modulators.

To quantify the cell type specificity of the selected HRGs, we calculated the gene scores[71] using the top 15 HRGs of each cell type (Figure 3E). The resulting correlation matrix revealed a sharp diagonal structure, indicating that HRGs are mainly relevant to their respective cell identities, with little off-target assignment. We systematically compared ProtoCloud HRGs against curated lists of known cell type markers from ScType.[30] The top cell-type-specific HRGs also overlap with differentially expressed genes specific to cell types (Figure S3B). As expected, the specificity index (Tau score[72]) of HRGs ($\tau = 0.801$) was significantly higher than that of random background genes and comparable to the curated markers (Figure 3F). Furthermore, HRGs showed elevated expression levels in their own cell types and distinctive expression profiles across other populations (Figure 3G; Figure S3C).

The relevance of HRGs highlighted subtle biological differences between closely related cell subtypes. For example, differential HRGs between mature and naive B cells identified key distinguishing genes (Figure 3H; STAR Methods). Highly relevant genes such as *CD79B*, *BANK1*, and MHC class II genes were more highly expressed in naive B cells, while *LY9* was more specific to mature B cells.

We further investigated the consistency of HRGs across different organs using three datasets (lung, esophagus, and spleen) from the Tissue Stability Cell Atlas (TSCA).[64] Shared cell types exhibited a remarkable overlap in HRGs. For example, the top ten HRGs of mature B cells in the lung and spleen datasets had a Jaccard index of 0.5, and the top ten HRGs of lung mature B cells and esophageal $CD27^+$ B cells had a Jaccard index of 0.75 (Table S3). The HRGs of mature B cells in both the lung and the spleen datasets included canonical markers such as *IGKC*, *CD79A*, and *MS4A1*, and this consistency extended to $CD27^+$ B cells in the esophagus dataset. These results highlight ProtoCloud's ability to uncover key features of individual cell types, demonstrating the robustness of its explainability across diverse tissue contexts.

Similarity-guided annotation correction with justification

Reliability is a key aspect of model interpretability, reflecting how trustworthy cell type annotations are. ProtoCloud assesses prediction confidence through multiple complementary approaches. For each cell, our model calculates its similarity score to the closest prototype of the assigned class, enabling quality control of the annotation. These similarity scores between cells and their prototypes exhibit a strong positive correlation with prediction accuracy (Figure 4A; STAR Methods). We evaluated similarity scores using two calibration metrics (STAR Methods), yielding a Brier score[73] of 0.092 and an expected calibration error (ECE)[74] of 0.198, indicating that the similarity scores themselves are informative measures of prediction confidence.

From these observations, ProtoCloud categorizes its predictions based on the similarity score into two classes: certain and ambiguous (Figure 4B). The similarity threshold used for this classification is automatically calculated and tailored to individual cell types (STAR Methods). Predictions with scores exceeding their respective thresholds are classified as "certain," indicating high confidence in these predictions. Conversely, predictions falling below the threshold are marked as "ambiguous," signaling the need for cautious interpretation. Given ground truth annotations, ProtoCloud further identifies two distinct types of annotation discrepancies, characterized by prediction confidence and inconsistencies between the model's prediction and the original annotation (Figure 4C). We illustrate each scenario with explanatory examples from the PBMC30K dataset (Figures 4D–4F).

Type 1: Misannotated

ProtoCloud confidently predicts a cell type that differs from the dataset's original annotation. In these cases, the cells are likely

separation between the original and predicted labels. The shared region contains genes that are top-ranked HRGs or known markers in both cell types. Asterisks denote marker genes that are also present in the respective HRG set. (D) $CD4^+$ T cells versus cytotoxic T cells. Type 1 cells were originally labeled as $CD4^+$ T cells and re-annotated as cytotoxic T cells. (E) Cytotoxic T cells versus $CD4^+$ T cells. (F) Cytotoxic T cells versus natural killer cells. HRGs, highly relevant genes and M, Marker.

(G) Reliability diagram of similarity scores and calibrated similarities of benchmarking datasets. The *x* axis of each triangle represents the average similarity score of a dataset, while that of a circle represents the average calibrated probability. The diagonal dashed line indicates perfect calibration, where predicted uncertainty precisely matches observed accuracy rates.

(H) Reliability diagram of calibrated probability between dichotomous confidence groups of benchmarking datasets. Each point represents one dataset, stratified into "certain" (circles) and "ambiguous" (diamonds) subsets. The dashed diagonal indicates perfect calibration, where predicted certainty matches the empirical accuracy.

misannotated in the original dataset and should be re-annotated according to ProtoCloud's prediction. The gene expression of HRGs for type 1 cells aligns closely with the predicted cell type rather than the original annotation, providing strong evidence for annotation errors in the dataset. For example, in the PBMC30K dataset, cells originally annotated as $CD4^+$ T cells but reclassified by ProtoCloud as cytotoxic T cells exhibited an average *NKG7* expression of 3.248, compared to only 0.932 in the originally annotated $CD4^+$ T cells and 3.809 in cytotoxic T cells (Figure 4D). This substantial increase in *NKG7* expression supports the reclassification. Furthermore, the average Pearson correlation of gene expression between the re-annotated cells and originally annotated $CD4^+$ T cells was 0.695, while the correlation with cytotoxic T cells was 0.831, further supporting ProtoCloud's reclassification.

Type 2: Ambiguous misprediction

This category includes ambiguous cells whose predicted labels do not align with the original annotations. The expression patterns of the top HRGs for these cells do not distinctly characterize specific cell types. Although the similarity scores may favor one cell type, our model lacks sufficient evidence from key decision-making genes to support a confident prediction. Given their potential to introduce noise into downstream analyses, these cells are excluded from re-annotation.

To provide granular, per-cell uncertainty estimates, we map the similarity scores to calibrated probability scores using isotonic regression[75] (STAR Methods). These calibrated scores achieved substantially improved performance on both calibration metrics, with the Brier score reduced to 0.048 (from 0.092) and ECE to 0.031 (from 0.198) (Figure 4G). Consistent with our dichotomous confidence categorization, cells designated as "certain" exhibited a markedly higher average calibrated score (0.960) compared to those classified as "ambiguous" (0.779). Importantly, observed accuracy aligned with these confidence estimates (0.960 and 0.783), demonstrating that the calibrated probability scores reliably reflect true classification performance (Figure 4H).

ProtoCloud supports time-course data annotation

To further demonstrate ProtoCloud's ability to assess prediction uncertainty and leverage this functionality for downstream data analysis, we used ProtoCloud to analyze a challenging time-course dataset of mouse RGCs following optic nerve crush (ONC).[53] The cells were collected at seven time points: 0, 0.5, 1, 2, 4, 7, and 14 days post-crush (dpc). The initial model was trained on the control group (0 dpc, Figure 5A). The trained model was then applied to the data from the subsequent time point (0.5 dpc), generating predictions categorized as "certain" or "ambiguous." Cells with "certain" predictions constituted the new training set, using their predicted labels for subsequent training iterations, while ambiguous cases were temporarily held out for validation and re-annotation by the updated model. The initial ProtoCloud model was subsequently trained for 30 epochs on these cells and applied to cells from 1 dpc to obtain "certain" and "ambiguous" cells. This iterative training and annotation process was repeated for each subsequent time point.

By capturing temporal dynamics in gene expression patterns, ProtoCloud demonstrated its ability to track changes in cellular states throughout the course of injury progression. For confidently annotated cells, the model achieved over 94% accuracy at time points before 4 dpc and maintained accuracy above 68% even at 14 dpc (Figure 5B; Table S4). Notably, the iterative training strategy improved ProtoCloud prediction accuracy for the initially ambiguous cells by 2%–10% (Figure 5C). In contrast, both scANVI and CellTypist's performance declined substantially when using the same limited set of annotated cells for training (Table S4).

The proportion of "unassigned" cells increased over time in the original annotations, particularly after 4 dpc, where it increased sharply to approximately 23%. ProtoCloud confidently reclassified 30%–70% of these previously unassigned cells (Figure 5D; Table S5). For example, cluster 1 (C1) RGCs, a T5 RGC subtype characterized by marker gene *Tusc5*,[53] constituted the largest portion of these re-annotated cells. As a result, the estimated survival rate of this subtype increased from 1.2% to 5.6% based on our re-annotations, compared to the original labels. Multiple lines of evidence support the validity of these classifications. The UMAP visualization shows a strong alignment of newly predicted C1 cells with C1 prototypes (Figure 5D), and a comparative analysis according to T5 RGC markers and HRGs confirmed that the predicted and reference C1 cells shared highly similar expression patterns (Figures 5E and 5F; Figures S4A and S4B). These predicted C1 cells can be distinguished from other closely related T5 RGCs by HRGs. For example, for the cells at 4 dpc (Figure 5E), C1 and predicted C1 cells did not express *Ntrk1* (mainly expressed by C13 RGCs), *Chrm2* (mainly expressed by C23 RGCs), *Gabrg3*, *Cpne4*, and *Pcdh7* (expressed highly in other T5 RGCs compared to C1 RGCs). The top C1 HRGs showed a systematic but moderate downregulation pattern in predicted C1 cells compared to reference C1 cells, consistent with an injury-responsive state rather than a distinct cell type (Figures S4C and S4D).

The next largest portion of re-assigned cells belonged to cluster 10 (C10) RGCs, an ON-OFF direction-selective ganglion cell (ooDSGC) subtype distinctly marked by *Gpr88* expression (Figures 5G and 5H; Figures S4E and S4F). Although the marker gene exhibited distinct expression in the control group, this signal diminished at later time points. The identified HRGs effectively track these molecular changes, providing reliable references for cell classification throughout the damage progression. At 7 dpc, C10 and predicted C10 cells did not express *Fam19a4* (expressed by C24 RGCs), *Ccer2*, and *Mmp17* (mainly expressed by C12 RGCs) but expressed *Col25a1* and *Sparcl1* at a lower level compared to C16 RGCs (Figure 5H). At other time points (i.e., 4 dpc and 14 dpc), C10 and predicted C10 cells were highly similar to each other yet clearly distinct from other ooDSGCs (Figures S4E and S4F). Again, we showed that the predicted C10 cells were close to C10 cells in the UMAP (Figure 5D).

We further validated ProtoCloud's robust transferability across different techniques by applying the model trained on AtlasRGC to an RGC dataset,[76] generated using the Patch-seq technology. ProtoCloud confidently classified 52% of the cells with an accuracy of 97.6%. We then applied the calibrator trained on AtlasRGC to the Patch-seq RGC predictions. Notably, the calibrated scores remained well aligned with the empirical accuracy on this new dataset (Figures S4G–S4I).

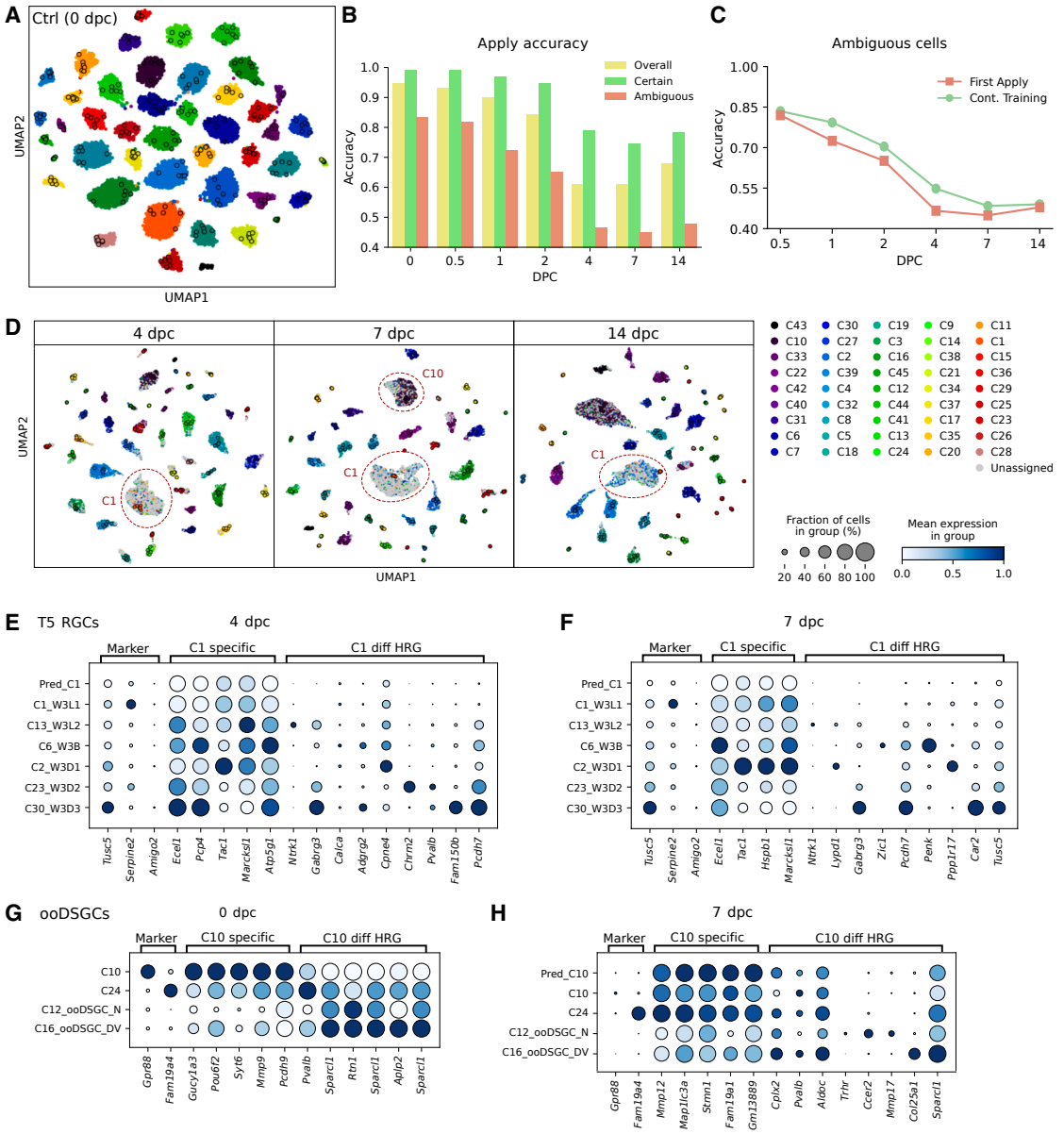


Figure 5. Continue training on the time-course RGC ONC dataset

(A) UMAP visualization of the control group (0 dpc) RGCs.

(B) Initial prediction accuracy at different time points when applying models trained on the previous time point data. For each time point, accuracies are shown for overall predictions (yellow), certain predictions (green), and ambiguous predictions (orange).

(C) Comparison of prediction accuracy for ambiguous cells only, before and after continued training. "First Apply" shows the accuracy of ambiguous predictions from the initial model application (orange bars in [B]). "Cont. Training" shows improved accuracy for these same ambiguous cells after training the model for an additional 30 epochs on cells with certain predictions.

(D) UMAP visualization of later time points at 4 dpc (left), 7 dpc (middle), and 14 dpc (right). Cells with certain predictions from both the initial application and after continued training are plotted to visualize the results. The prototypes are represented by dots with black edges. Cell types are ordered by injury survival, from the most resilient (C43) to the most susceptible (C28), with an increasing proportion of unassigned cells after 4 dpc.

(E and F) (E) HRG expression of T5-RGCs at 4 dpc and (F) 7 dpc. Comparison of top-ranked HRGs and differentially ranked HRGs between C1 RGCs and other T5 RGCs. "C1 specific" are highly ranked HRGs within C1 (rank < 30) with low relevance in other types (mean rank > 200). "C1 diff HRG" represents genes with the opposite pattern. Re-assigned "unassigned" cells predicted as C1 are included with "Pred_C1." "Marker" refers to known canonical markers for each cell type based on established literature.[53] Genes include known markers (*Tusc5* shared by all T5 RGCs, *Serpine2* and *Amigo2*, and C1 RGC marker genes), HRGs shared by all T5 RGCs, and differentially ranked HRGs between C1 and other T5 RGCs.

(G and H) (G) HRG expression of ooDSGCs at 0 dpc and (H) 7 dpc, including predicted C10 cells (Pred_C10). Shared and distinct HRGs are compared between C10 and other ooDSGCs, with known markers *Gpr88* (a C10 RGC marker) and *Fam19a4* (a C24 RGC marker).

ProtoCloud transfers labels from the esophageal mucosal cell atlas

We applied ProtoCloud to a recently published esophageal mucosal cell atlas (AtlasEoE[25]), which comprises cells from 37 patient biopsies taken from 22 donors across three conditions: healthy, remission eosinophilic esophagitis (EoE), and active EoE. ProtoCloud achieved an overall validation accuracy of 94.5% and a macro F1 score of 0.877 on this large dataset, which comprises 60 prevalent cell types (n = 393,763) and 12 additional rare cell types exhibiting patient-specific bias[25] (n = 1,215). In particular, the model achieved 97% accuracy specifically for the 12 rare cell states (Figure S5A). In contrast, scANVI and CellTypist exhibited frequent misclassifications, particularly confusing cells related to epithelial populations (Figures S5B and S5C).

As expected, ProtoCloud identified cell-type-specific HRGs, with known cell type marker genes ranked highly in terms of their prototypical relevance scores (Figure S5D). Notably, for *ALOX15*$^+$ macrophages, *PRDM16*$^+$ dendritic cells (cDC2Cs), and group 2 innate lymphoid cells (ILC2s)—three rare cell types of potential importance for EoE pathogenesis—their top HRGs included their respective known marker genes. ProtoCloud accurately identified *MMP12* and *ALOX15* as key relevant genes for *ALOX15*$^+$ macrophages (with relevance scores of 0.615 and 0.267, respectively) but not for other macrophages (relevance score < 0.03) (Figure 6A). *PRDM16*$^+$ cDC2Cs can be distinguished from cDC2Bs based on *PRDM16*, *TGM2*, and *PIGR* (Figure 6B). ILC2, a rare population enriched in active EoE patients,[77] can be distinguished from other IL-13 and IL-5 expressing immune cells (e.g., T helper 2 cells [Th2]) by its upregulation of prostaglandin-related genes *PTGDR2*, *PTGS2*, and *HPGD*[25] (Figure 6C; Table S6). We also identified shared HRGs associated with cell cycle activity (*TUBA1B*, *HMGB2*, *TUBB*, *H2AFZ*, and *KIAA0101*) across proliferating cell populations (Figure 6D).

We transferred this granular annotation (72 cell states) to two independent EoE datasets[46,78] (Figure S6A), produced by the 10× Chromium[3] and the SeqWell[6] platforms, respectively. In the Clevenger et al. dataset,[46] ProtoCloud classified 88.7% of cells as epithelial populations (including quiescent basal, cycling basal, suprabasal, and apical cells), closely aligning with the published proportion of 86.83%. The eight major cell populations identified by Morgan et al.[78] provided broader categories under which our more granular subcategory annotations were mapped (Figure 6E).

Importantly, the three rare but potentially important cell types in the esophagus of EoE patients were clearly identified in the two test datasets, despite their absence in the original annotations.[46,78] As expected, *ALOX15*$^+$ macrophages expressed the marker genes *ALOX15* and *MMP12*; *PRDM16*$^+$ dendritic cells expressed *PRDM16*, *PIGR*, and *TGM2*; and ILC2s expressed *IL13*, *PTGS2*, and *IL17RB* (Figures S6B and S6C). Moreover, we observed a notable increase in the proportion of these cells annotated by ProtoCloud in active EoE patients compared to healthy or remission individuals (Figures 6F and 6G), consistent with previous findings.[25]

We compared this complex label-transfer scenario to the top two benchmark methods: CellTypist and scANVI (Table S7). While CellTypist also identified major immune cell subsets, it underrepresented rare populations, such as ILC2s and cycling DCs. Although scANVI also identified these rare cell states and, in some cases, assigned more cells to these rare cell states, the cells specifically identified by scANVI had lower marker gene signature scores compared to those identified by ProtoCloud (Figures S6D and S6E), indicating low-specificity predictions.

DISCUSSION

We present ProtoCloud, a self-explanatory model that augments the variational autoencoder architecture with cell-type-specific prototypes to annotate and analyze scRNA-seq data with enhanced transparency. While a trade-off between performance and explainability is often observed in machine learning,[79,80] ProtoCloud demonstrates that these objectives can be achieved simultaneously, matching or exceeding the performance of state-of-the-art methods. To achieve both explainability and high accuracy, ProtoCloud incorporates a carefully designed architecture, including a decomposed latent space to disentangle cell type information from cell-type-irrelevant nuisance factors, specialized loss functions (e.g., the atomic loss), and a two-stage training strategy tailored to this architecture. ProtoCloud's latent space effectively captures both inter-cell type and intra-cell type variations while disentangling additional biological or technical variations without requiring prior knowledge of these cell-type-irrelevant factors.

While current deep learning methods achieve impressive accuracy in cell type annotation, ProtoCloud addresses their intrinsic lack of transparency by incorporating cell-type-specific prototypes and PRP. To the best of our knowledge, ProtoCloud is the first approach to achieve both cell-level and gene-level interpretability in this context. While the granularity of PRP[54] is often considered a limitation in computer vision applications,[55] this fine-grained gene-level resolution is particularly advantageous for scRNA-seq analysis. The input (gene)-level resolution of PRP aligns with the functions of individual genes and directly reveals genes critical for model decisions. By identifying the genes that strongly activate specific prototypes, ProtoCloud implements a genomic version of the "this looks like that" decision process.[81]

ProtoCloud achieves explainability in terms of three essential criteria: explicitness, faithfulness, and stability.[42,82] The model demonstrates explicitness through its ability to capture HRGs that correspond to both established and novel cell type markers, enabling the distinction between closely related cell types. Faithfulness is evidenced by the confidence score assigned to each prediction, which enables the identification of annotation discrepancies, further supported by the HRG expression patterns in disputed cases. Finally, the stability of the learned prototypes is validated across different sequencing techniques, organs, and disease states. Independent analyses identified highly overlapping sets of HRGs for similar cell types in diverse datasets, spanning different technologies and tissues. These capabilities facilitate practical applications in complex biological contexts, helping uncover key molecular features of cell identities.

The practical utility of ProtoCloud was validated through two challenging disease applications. In a time-course analysis of RGCs, ProtoCloud successfully tracked cellular transitions

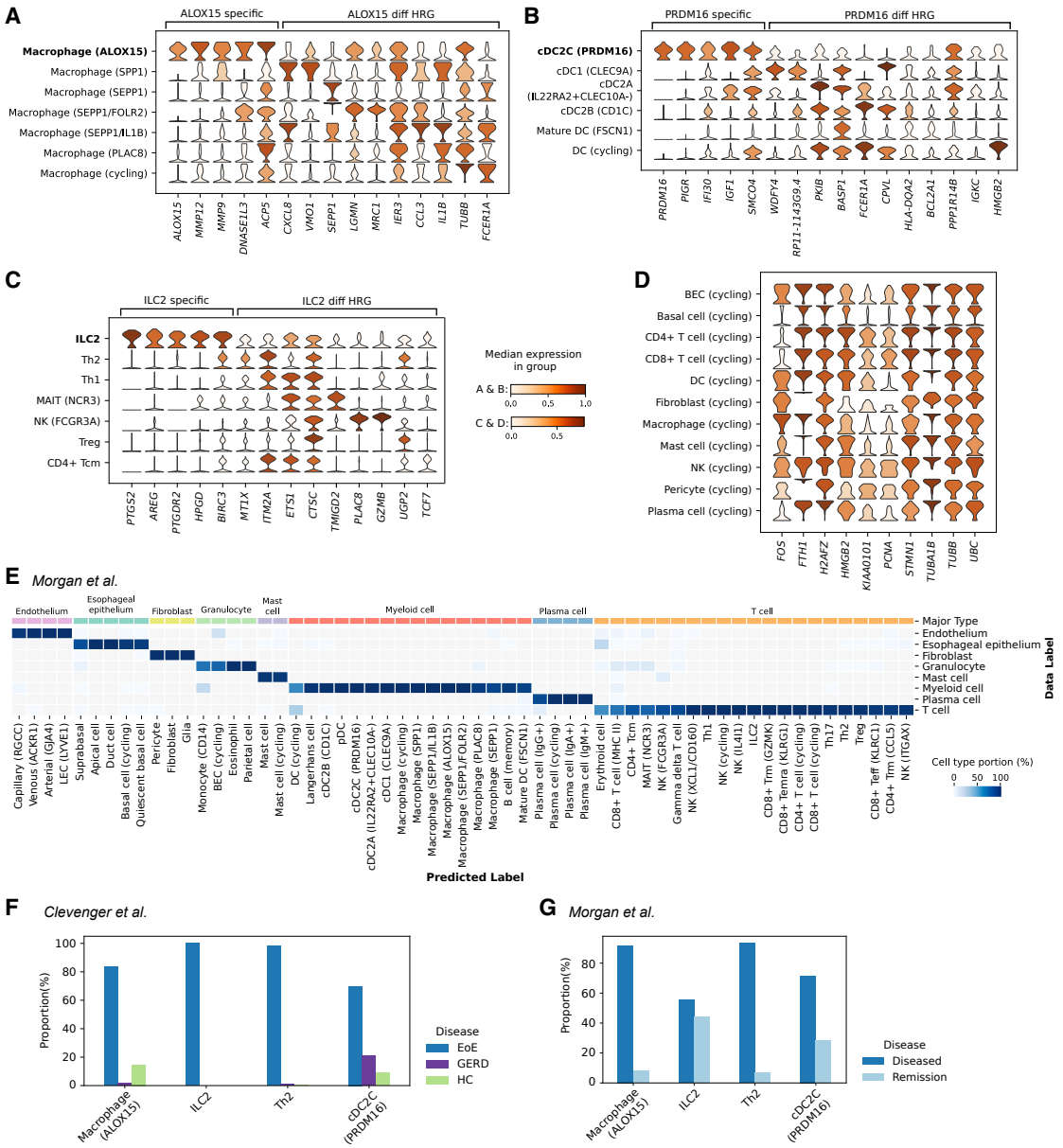


Figure 6. Annotation transfer across EoE datasets and disease conditions

(A) Expression profiles of differentially ranked HRGs that distinguish *ALOX15*$^+$ from *SPP1*$^+$ macrophages. "*ALOX15* specific" denotes highly ranked HRGs in *ALOX15*$^+$ macrophages (rank < 30) and low relevance in others (mean rank > 200). "*ALOX15* diff HRG" represents HRGs that are highly ranked in contrasting cell types but not *ALOX15*$^+$ macrophages.

(B) Expression profiles of differentially ranked HRGs between *PRDM16*$^+$ dendritic cells and *CD1C* + cDC2B cells.

(C) Expression patterns of top ILC2-specific HRGs across immune cell populations. Genes were compared between ILC2s and Th2 cells.

(D) Expression patterns of cycling-associated HRGs across proliferating cell populations.

(E) Annotation transfer correspondence between the AtlasEoE[25] dataset (72 cell types) and the Morgan et al. dataset[78] (eight cell types). Color bar: column-wise proportion.

(F and G) (F) Cell type composition across disease conditions in the Clevenger et al. dataset[46] and (G) the Morgan et al. dataset. EoE, eosinophilic esophagitis; GERD, gastroesophageal reflux disease; and HC, healthy.

despite limited reference data. In the EoE analysis, it accurately identified and tracked rare disease-specific cell populations across three datasets with varying disease conditions. In both cases, ProtoCloud's explainability provided molecular-level evidence for disease-specific cellular dynamics, demonstrating its utility in complex, real-world scenarios.

In summary, ProtoCloud transforms single-cell annotation from a black-box prediction process into an evidence-based analytical

framework, providing researchers with actionable insights into cellular identity. The model's ability to identify cell-type-specific genes helps bridge the gap between computational predictions and biological understanding. While ProtoCloud effectively identifies genes crucial for cell type classification, it does not directly elucidate the underlying biological mechanisms or environmental interactions. The causal relationships between HRGs and cell types warrant further systematic investigation. By advancing explainable approaches to cell type annotation, ProtoCloud represents a significant step toward more comprehensive and transparent single-cell analysis.

Limitations of the study

Currently, ProtoCloud requires an annotated reference dataset for training. This means that in the absence of such data, ProtoCloud cannot be directly applied; however, as technology advances, the availability of high-quality reference datasets continues to increase.[11] Second, in this study, we focused on scRNA-seq data analysis. Extending ProtoCloud to other modalities, such as single-cell assay for transposase-accessible chromatin using sequencing (scATAC-seq) data,[83] should be straightforward. Applications of ProtoCloud to multimodal data for learning gene regulatory relationships have not yet been explored and represent a promising direction for future research. Finally, although ProtoCloud can identify novel cell types not present in the training data through its built-in uncertainty estimation, how to incrementally update a pretrained ProtoCloud model to incorporate new cell types remains nontrivial and warrants further investigation.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Jiarui Ding (jiarui.ding@ubc.ca).

Materials availability

This study did not generate new, unique reagents.

Data and code availability

- All data used in this study are publicly available. The datasets used are available through their sources in the key resources table.
- All original code has been deposited at https://github.com/Ding-Group/ProtoCloud and achieved in Zenodo (https://doi.org/10.5281/zenodo.18882740).
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

ACKNOWLEDGMENTS

We thank Dr. Anne Condon, Jeffrey Niu, Minuk Ma, and Carlos Vasquez for their constructive criticism and valuable feedback on this work and the members of the Ding group for insightful discussions and support. This work was supported by a Discovery grant from the Natural Sciences and Engineering Research Council of Canada (NSERC) and a department startup fund from the University of British Columbia (to J.D.). J.D. is a Canada Research Chair and is supported by the Canadian Institutes of Health Research through the Canada Research Chair Program. The computational resource is partially supported by the Canada Foundation for Innovation & John. R. Evans Leader Fund (to J.D.). This research was supported in part through the computational resources and services provided by Advanced Research Computing at the University of British Columbia.

AUTHOR CONTRIBUTIONS

J.D. conceived the project. J.D. and K.G. developed the model. K.G. conducted experimental analyses with guidance from J.D. K.G. and J.D. interpreted the results and wrote the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

STAR★METHODS

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

- KEY RESOURCES TABLE
- METHOD DETAILS
  - The ProtoCloud model
  - The orthogonal loss to help learn diverse prototypes for each class
  - The atomic loss to help learn diverse and accurate prototypes
  - Shaping latent spaces through a two-stage curriculum and latent decomposition
  - Prototypical relevance propagation for gene-level interpretation
  - Data augmentation for training robust ProtoCloud models
  - Model architecture and training
  - Ablation study
  - Reliability and uncertainty estimation
  - Differentially ranked highly relevant genes
  - Baselines for comparison
  - Evaluation metrics
  - Data collection and processing

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.xgen.2026.101217.

Received: May 8, 2025
Revised: February 2, 2026
Accepted: March 17, 2026
Published: April 16, 2026

REFERENCES

1. Klein, A.M., Mazutis, L., Akartuna, I., Tallapragada, N., Veres, A., Li, V., Peshkin, L., Weitz, D.A., and Kirschner, M.W. (2015). Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. Cell *161*, 1187–1201. https://doi.org/10.1016/j.cell.2015.04.044.
2. Macosko, E.Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., Tirosh, I., Bialas, A.R., Kamitaki, N., Martersteck, E.M., et al. (2015). Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. Cell *161*, 1202–1214. https://doi.org/10.1016/j.cell.2015.05.002.
3. Zheng, G.X.Y., Terry, J.M., Belgrader, P., Ryvkin, P., Bent, Z.W., Wilson, R., Ziraldo, S.B., Wheeler, T.D., McDermott, G.P., Zhu, J., et al. (2017). Massively parallel digital transcriptional profiling of single cells. Nat. Commun. *8*, 14049. https://doi.org/10.1038/ncomms14049.
4. Cao, J., Packer, J.S., Ramani, V., Cusanovich, D.A., Huynh, C., Daza, R., Qiu, X., Lee, C., Furlan, S.N., Steemers, F.J., et al. (2017). Comprehensive single-cell transcriptional profiling of a multicellular organism. Science *357*, 661–667. https://doi.org/10.1126/science.aam8940.
5. Rosenberg, A.B., Roco, C.M., Muscat, R.A., Kuchina, A., Sample, P., Yao, Z., Graybuck, L.T., Peeler, D.J., Mukherjee, S., Chen, W., et al. (2018). Single-cell profiling of the developing mouse brain and spinal cord with split-pool barcoding. Science *360*, 176–182. https://doi.org/10.1126/science.aam8999.

6. Gierahn, T.M., Wadsworth, M.H., Hughes, T.K., Bryson, B.D., Butler, A., Satija, R., Fortune, S., Love, J.C., and Shalek, A.K. (2017). Seq-Well: portable, low-cost RNA sequencing of single cells at high throughput. Nat. Methods *14*, 395–398. https://doi.org/10.1038/nmeth.4179.

7. Han, X., Wang, R., Zhou, Y., Fei, L., Sun, H., Lai, S., Saadatpour, A., Zhou, Z., Chen, H., Ye, F., et al. (2018). Mapping the mouse cell atlas by microwell-seq. Cell *172*, 1091–1107.e17. https://doi.org/10.1016/j.cell.2018.02.001.

8. Domínguez Conde, C., Xu, C., Jarvis, L.B., Rainbow, D.B., Wells, S.B., Gomes, T., Howlett, S.K., Suchanek, O., Polanski, K., King, H.W., et al. (2022). Cross-tissue immune cell analysis reveals tissue-specific features in humans. Science *376*, eabl5197. https://doi.org/10.1126/science.abl5197.

9. Travaglini, K.J., Nabhan, A.N., Penland, L., Sinha, R., Gillich, A., Sit, R.V., Chang, S., Conley, S.D., Mori, Y., Seita, J., et al. (2020). A molecular cell atlas of the human lung from single-cell RNA sequencing. Nature *587*, 619–625. https://doi.org/10.1038/s41586-020-2922-4.

10. Wagner, J., Rapsomaniki, M.A., Chevrier, S., Anzeneder, T., Langwieder, C., Dykgers, A., Rees, M., Ramaswamy, A., Muenst, S., Soysal, S.D., et al. (2019). A single-cell atlas of the tumor and immune ecosystem of human breast cancer. Cell *177*, 1330–1345.e18. https://doi.org/10.1016/j.cell.2019.03.005.

11. Regev, A., Teichmann, S.A., Lander, E.S., Amit, I., Benoist, C., Birney, E., Bodenmiller, B., Campbell, P., Carninci, P., Clatworthy, M., et al. (2017). The human cell atlas. eLife *6*, e27041. https://doi.org/10.7554/eLife.27041.

12. Smillie, C.S., Biton, M., Ordovas-Montanes, J., Sullivan, K.M., Burgin, G., Graham, D.B., Herbst, R.H., Rogel, N., Slyper, M., Waldman, J., et al. (2019). Intra-and inter-cellular rewiring of the human colon during ulcerative colitis. Cell *178*, 714–730.e22. https://doi.org/10.1016/j.cell.2019.06.029.

13. Xu, H., Ding, J., Porter, C.B.M., Wallrapp, A., Tabaka, M., Ma, S., Fu, S., Guo, X., Riesenfeld, S.J., Su, C., et al. (2019). Transcriptional atlas of intestinal immune cells reveals that neuropeptide α-CGRP modulates group 2 innate lymphoid cell responses. Immunity *51*, 696–708.e9. https://doi.org/10.1016/j.immuni.2019.09.004.

14. Efremova, M., Vento-Tormo, M., Teichmann, S.A., and Vento-Tormo, R. (2020). CellPhoneDB: inferring cell–cell communication from combined expression of multi-subunit ligand–receptor complexes. Nat. Protoc. *15*, 1484–1506. https://doi.org/10.1038/s41596-020-0292-x.

15. Aizarani, N., Saviano, A., Sagar, null, Mailly, L., Durand, S., Herman, J.S., Pessaux, P., Baumert, T.F., and Grün, D. (2019). A human liver cell atlas reveals heterogeneity and epithelial progenitors. Nature *572*, 199–204. https://doi.org/10.1038/s41586-019-1373-2.

16. Plasschaert, L.W., Žilionis, R., Choo-Wing, R., Savova, V., Knehr, J., Roma, G., Klein, A.M., and Jaffe, A.B. (2018). A single-cell atlas of the airway epithelium reveals the CFTR-rich pulmonary ionocyte. Nature *560*, 377–381. https://doi.org/10.1038/s41586-018-0394-6.

17. Gavish, A., Tyler, M., Greenwald, A.C., Hoefflin, R., Simkin, D., Tschernichovsky, R., Galili Darnell, N., Somech, E., Barbolin, C., Antman, T., et al. (2023). Hallmarks of transcriptional intratumour heterogeneity across a thousand tumours. Nature *618*, 598–606. https://doi.org/10.1038/s41586-023-06130-4.

18. Karaiskos, N., Wahle, P., Alles, J., Boltengagen, A., Ayoub, S., Kipar, C., Kocks, C., Rajewsky, N., and Zinzen, R.P. (2017). The Drosophila embryo at single-cell transcriptome resolution. Science *358*, 194–199. https://doi.org/10.1126/science.aan3235.

19. Wagner, D.E., Weinreb, C., Collins, Z.M., Briggs, J.A., Megason, S.G., and Klein, A.M. (2018). Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo. Science *360*, 981–987. https://doi.org/10.1126/science.aar4362.

20. Tabula Muris Consortium (2020). A single-cell transcriptomic atlas characterizes ageing tissues in the mouse. Nature *583*, 590–595. https://doi.org/10.1038/s41586-020-2496-1.

21. He, S., Wang, L.-H., Liu, Y., Li, Y.-Q., Chen, H.-T., Xu, J.-H., Peng, W., Lin, G.-W., Wei, P.-P., Li, B., et al. (2020). Single-cell transcriptome profiling of an adult human cell atlas of 15 major organs. Genome Biol. *21*, 294. https://doi.org/10.1186/s13059-020-02210-0.

22. Jin, K., Yao, Z., van Velthoven, C.T.J., Kaplan, E.S., Glattfelder, K., Barlow, S.T., Boyer, G., Carey, D., Casper, T., Chakka, A.B., et al. (2025). Brain-wide cell-type-specific transcriptomic signatures of healthy ageing in mice. Nature *638*, 182–196. https://doi.org/10.1038/s41586-024-08350-8.

23. Vázquez-García, I., Uhlitz, F., Ceglia, N., Lim, J.L., Wu, M., Mohibullah, N., Niyazov, J., Ruiz, A.E.B., Boehm, K.M., Bojilova, V., et al. (2022). Ovarian cancer mutational processes drive site-specific immune evasion. Nature *612*, 778–786. https://doi.org/10.1038/s41586-022-05496-1.

24. Alladina, J., Smith, N.P., Kooistra, T., Slowikowski, K., Kernin, I.J., Deguine, J., Keen, H.L., Manakongtreecheep, K., Tantivit, J., Rahimi, R.A., et al. (2023). A human model of asthma exacerbation reveals transcriptional programs and cell circuits specific to allergic asthma. Sci. Immunol. *8*, eabq6352. https://doi.org/10.1126/sciimmunol.abq6352.

25. Ding, J., Garber, J.J., Uchida, A., Lefkovith, A., Carter, G.T., Vimalathas, P., Canha, L., Dougan, M., Staller, K., Yarze, J., et al. (2024). An esophagus cell atlas reveals dynamic rewiring during active eosinophilic esophagitis and remission. Nat. Commun. *15*, 3344. https://doi.org/10.1038/s41467-024-47647-0.

26. Xu, C., Lopez, R., Mehlman, E., Regier, J., Jordan, M.I., and Yosef, N. (2021). Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. Mol. Syst. Biol. *17*, e9620. https://doi.org/10.15252/msb.20209620.

27. 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.

28. Ergen, C., Xing, G., Xu, C., Kim, M., Jayasuriya, M., McGeever, E., Oliveira Pisco, A., Streets, A., and Yosef, N. (2024). Consensus prediction of cell type labels in single-cell data with popV. Nat. Genet. *56*, 2731–2738. https://doi.org/10.1038/s41588-024-01993-3.

29. Satija, R., Farrell, J.A., Gennert, D., Schier, A.F., and Regev, A. (2015). Spatial reconstruction of single-cell gene expression data. Nat. Biotechnol. *33*, 495–502. https://doi.org/10.1038/nbt.3192.

30. Ianevski, A., Giri, A.K., and Aittokallio, T. (2022). Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data. Nat. Commun. *13*, 1246. https://doi.org/10.1038/s41467-022-28803-w.

31. Ding, J., Condon, A., and Shah, S.P. (2018). Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. Nat. Commun. *9*, 2002. https://doi.org/10.1038/s41467-018-04368-5.

32. Lopez, R., Regier, J., Cole, M.B., Jordan, M.I., and Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. Nat. Methods *15*, 1053–1058. https://doi.org/10.1038/s41592-018-0229-2.

33. Amodio, M., Van Dijk, D., Srinivasan, K., Chen, W.S., Mohsen, H., Moon, K.R., Campbell, A., Zhao, Y., Wang, X., Venkataswamy, M., et al. (2019). Exploring single-cell data with deep multitasking neural networks. Nat. Methods *16*, 1139–1145. https://doi.org/10.1038/s41592-019-0576-7.

34. Eraslan, G., Simon, L.M., Mircea, M., Mueller, N.S., and Theis, F.J. (2019). Single-cell RNA-seq denoising using a deep count autoencoder. Nat. Commun. *10*, 390. https://doi.org/10.1038/s41467-018-07931-2.

35. Li, X., Wang, K., Lyu, Y., Pan, H., Zhang, J., Stambolian, D., Susztak, K., Reilly, M.P., Hu, G., and Li, M. (2020). Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. Nat. Commun. *11*, 2338. https://doi.org/10.1038/s41467-020-15851-3.

36. Ding, J., and Regev, A. (2021). Deep generative model embedding of single-cell RNA-Seq profiles on hyperspheres and hyperbolic spaces. Nat. Commun. *12*, 2554. https://doi.org/10.1038/s41467-021-22851-4.

37. Yang, F., Wang, W., Wang, F., Fang, Y., Tang, D., Huang, J., Lu, H., and Yao, J. (2022). scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data. Nat. Mach. Intell. *4*, 852–866. https://doi.org/10.1038/s42256-022-00534-z.

38. Wen, H., Tang, W., Dai, X., Ding, J., Jin, W., Xie, Y., and Tang, J. (2024). CellPLM: pre-training of cell language model beyond single cells. International Conference on Learning Representations. https://doi.org/10.1101/2023.10.03.560734. https://openreview.net/forum?id=BKXvPDekud.

39. Cui, H., Wang, C., Maan, H., Pang, K., Luo, F., Duan, N., and Wang, B. (2024). scGPT: toward building a foundation model for single-cell multi-omics using generative AI. Nat. Methods *21*, 1470–1480. https://doi.org/10.1038/s41592-024-02201-0.

40. Karin, J., Mintz, R., Raveh, B., and Nitzan, M. (2024). Interpreting single-cell and spatial omics data using deep neural network training dynamics. Nat. Comput. Sci. *4*, 941–954. https://doi.org/10.1038/s43588-024-00721-5.

41. Arrieta, A.B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Inf. Fusion *58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012.

42. Alvarez-Melis, D., and Jaakkola, T.S. (2018). Towards robust interpretability with self-explaining neural networks. In Proceedings of the 32nd international conference on neural information processing systems NIPS'18 (Curran Associates Inc.), pp. 7786–7795.

43. Johansen, N., and Quon, G. (2019). scAlign: a tool for alignment, integration, and rare cell identification from scRNA-seq data. Genome Biol. *20*, 166. https://doi.org/10.1186/s13059-019-1766-4.

44. Falcone, F.H., Haas, H., and Gibbs, B.F. (2000). The human basophil: a new appreciation of its role in immune responses. Blood *96*, 4028–4038. https://doi.org/10.1182/blood.V96.13.4028.

45. Rochman, M., Wen, T., Kotliar, M., Dexheimer, P.J., Ben-Baruch Morgenstern, N., Caldwell, J.M., Lim, H.-W., and Rothenberg, M.E. (2022). Single-cell RNA-Seq of human esophageal epithelium in homeostasis and allergic inflammation. JCI Insight *7*, e159093. https://doi.org/10.1172/jci.insight.159093.

46. Clevenger, M.H., Karami, A.L., Carlson, D.A., Kahrilas, P.J., Gonsalves, N., Pandolfino, J.E., Winter, D.R., Whelan, K.A., and Tétreault, M.-P. (2023). Suprabasal cells retain progenitor cell identity programs in eosinophilic esophagitis–driven basal cell hyperplasia. JCI Insight *8*, e171765. https://doi.org/10.1172/jci.insight.171765.

47. Ben-Baruch Morgenstern, N., Rochman, M., Kotliar, M., Dunn, J.L.M., Mack, L., Besse, J., Natale, M.A., Klingler, A.M., Felton, J.M., Caldwell, J.M., et al. (2024). Single-cell RNA-sequencing of human eosinophils in allergic inflammation in the esophagus. J. Allergy Clin. Immunol. *154*, 974–987. https://doi.org/10.1016/j.jaci.2024.05.029.

48. Mereu, E., Lafzi, A., Moutinho, C., Ziegenhain, C., McCarthy, D.J., Álvarez-Varela, A., Batlle, E., Sagar, n, Grün, D., Lau, J.K., et al. (2020). Benchmarking single-cell RNA-sequencing protocols for cell atlas projects. Nat. Biotechnol. *38*, 747–755. https://doi.org/10.1038/s41587-020-0469-4.

49. Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D.J., Hicks, S.C., Robinson, M.D., Vallejos, C.A., Campbell, K.R., Beerenwinkel, N., Mahfouz, A., et al. (2020). Eleven grand challenges in single-cell data science. Genome Biol. *21*, 31. https://doi.org/10.1186/s13059-020-1926-6.

50. Korsunsky, I., Millard, N., Fan, J., Slowikowski, K., Zhang, F., Wei, K., Baglaenko, Y., Brenner, M., Loh, P.R., and Raychaudhuri, S. (2019). Fast, sensitive and accurate integration of single-cell data with Harmony. Nat. Methods *16*, 1289–1296. https://doi.org/10.1038/s41592-019-0619-0.

51. Chen, S., Regev, A., Condon, A., and Ding, J. (2026). CellUntangler: separating distinct biological signals in single-cell data with deep generative models. Cell Genom. *6*, 101073. https://doi.org/10.1016/j.xgen.2025.101073.

52. Wang, Z., Yeo, G.H., Sherwood, R., and Gifford, D. (2019). Disentangled representations of cellular identity. In Research in computational molecular biology (Springer International Publishing), pp. 256–271.

53. Tran, N.M., Shekhar, K., Whitney, I.E., Jacobi, A., Benhar, I., Hong, G., Yan, W., Adiconis, X., Arnold, M.E., Lee, J.M., et al. (2019). Single-cell profiles of retinal ganglion cells differing in resilience to injury reveal neuroprotective genes. Neuron *104*, 1039–1055.e12. https://doi.org/10.1016/j.neuron.2019.11.006.

54. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PLoS One *10*, e0130140. https://doi.org/10.1371/journal.pone.0130140.

55. Gautam, S., Höhne, M.M.-C., Hansen, S., Jenssen, R., and Kampffmeyer, M. (2023). This looks more like that: Enhancing self-explaining models by prototypical relevance propagation. Pattern Recogn. *136*, 109172. https://doi.org/10.1016/j.patcog.2022.109172.

56. Gautam, S., Boubekki, A., Hansen, S., Salahuddin, S., Jenssen, R., Höhne, M., and Kampffmeyer, M. (2022). ProtoVAE: A trustworthy self-explainable prototypical variational model. Adv. Neural Inf. Process. Syst. *35*, 17940–17952. https://doi.org/10.48550/arXiv.2210.08151.

57. Lundberg, S.M., and Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In Advances in Neural Information Processing Systems (Curran Associates, Inc.).

58. Kingma, D.P. (2014). Auto-encoding variational bayes. In 2nd international conference on learning representations.

59. Rezende, D.J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In International conference on machine learning (PMLR)), pp. 1278–1286.

60. Chen, J., Xu, H., Tao, W., Chen, Z., Zhao, Y., and Han, J.-D.J. (2023). Transformer for one stop interpretable cell type annotation. Nat. Commun. *14*, 223. https://doi.org/10.1038/s41467-023-35923-4.

61. De Donno, C., Hediyeh-Zadeh, S., Moinfar, A.A., Wagenstetter, M., Zappia, L., Lotfollahi, M., and Theis, F.J. (2023). Population-level integration of single-cell datasets enables multi-scale analysis across samples. Nat. Methods *20*, 1683–1692. https://doi.org/10.1038/s41592-023-02035-2.

62. Gonzalez-Ferrer, J., Lehrer, J., O'Farrell, A., Paten, B., Teodorescu, M., Haussler, D., Jonsson, V.D., and Mostajo-Radji, M.A. (2024). SIMS: A deep-learning label transfer tool for single-cell RNA sequencing analysis. Cell Genom. *4*, 100581. https://doi.org/10.1016/j.xgen.2024.100581.

63. Ding, J., Adiconis, X., Simmons, S.K., Kowalczyk, M.S., Hession, C.C., Marjanovic, N.D., Hughes, T.K., Wadsworth, M.H., Burks, T., Nguyen, L.T., et al. (2020). Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. Nat. Biotechnol. *38*, 737–746. https://doi.org/10.1038/s41587-020-0465-8.

64. Madissoon, E., Wilbrey-Clark, A., Miragaia, R.J., Saeb-Parsy, K., Mahbubani, K.T., Georgakopoulos, N., Harding, P., Polanski, K., Huang, N., Nowicki-Osuch, K., et al. (2019). scRNA-seq assessment of the human lung, spleen, and esophagus tissue stability after cold preservation. Genome Biol. *21*, 1. https://doi.org/10.1186/s13059-019-1906-x.

65. McHugh, M.L. (2012). Interrater reliability: the kappa statistic. Biochem. Med. *22*, 276–282. https://doi.org/10.11613/BM.2012.031.

66. Rautenstrauch, P., and Ohler, U. (2025). Shortcomings of silhouette in single-cell integration benchmarking. Nat. Biotechnol. https://doi.org/10.1038/s41587-025-02743-4.

67. McInnes, L., Healy, J., and Melville, J. (2018). UMAP: Uniform manifold approximation and projection for dimension reduction. Preprint at arXiv. https://doi.org/10.21105/joss.00861.

68. Zhang, X., Lan, Y., Xu, J., Quan, F., Zhao, E., Deng, C., Luo, T., Xu, L., Liao, G., Yan, M., et al. (2019). CellMarker: a manually curated resource of cell markers in human and mouse. Nucleic Acids Res. *47*, D721–D728. https://doi.org/10.1093/nar/gky900.

69. Zhang, Z., Luo, D., Zhong, X., Choi, J.H., Ma, Y., Wang, S., Mahrt, E., Guo, W., Stawiski, E.W., Modrusan, Z., et al. (2019). SCINA: a semi-supervised

subtyping algorithm of single cells and bulk samples. Genes *10*, 531. https://doi.org/10.3390/genes10070531.

70. Zhang, A.W., O'Flanagan, C., Chavez, E.A., Lim, J.L.P., Ceglia, N., McPherson, A., Wiens, M., Walters, P., Chan, T., Hewitson, B., et al. (2019). Probabilistic cell-type assignment of single-cell RNA-seq for tumor microenvironment profiling. Nat. Methods *16*, 1007–1015. https://doi.org/10.1038/s41592-019-0529-1.
71. Tirosh, I., Izar, B., Prakadan, S.M., Wadsworth, M.H., Treacy, D., Trombetta, J.J., Rotem, A., Rodman, C., Lian, C., Murphy, G., et al. (2016). Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. Science *352*, 189–196. https://doi.org/10.1126/science.aad0501.
72. Yanai, I., Benjamin, H., Shmoish, M., Chalifa-Caspi, V., Shklar, M., Ophir, R., Bar-Even, A., Horn-Saban, S., Safran, M., Domany, E., et al. (2005). Genome-wide midrange transcription profiles reveal expression level relationships in human tissue specification. Bioinforma. Oxf. Engl. *21*, 650–659. https://doi.org/10.1093/bioinformatics/bti042.
73. Glenn, W.B.; others (1950). Verification of forecasts expressed in terms of probability. Mon. Weather Rev. *78*, 1–3. https://doi.org/10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.
74. Nixon, J., Dusenberry, M.W., Zhang, L., Jerfel, G., and Tran, D. (2019). Measuring calibration in deep learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR) workshops.
75. Zadrozny, B., and Elkan, C. (2002). Transforming classifier scores into accurate multiclass probability estimates. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 694–699. https://doi.org/10.1145/775047.775151.
76. Huang, W., Xu, Q., Su, J., Tang, L., Hao, Z.-Z., Xu, C., Liu, R., Shen, Y., Sang, X., Xu, N., et al. (2022). Linking transcriptomes with morphological and functional phenotypes in mammalian retinal ganglion cells. Cell Rep. *40*, 111322. https://doi.org/10.1016/j.celrep.2022.111322.
77. Doherty, T.A., Baum, R., Newbury, R.O., Yang, T., Dohil, R., Aquino, M., Doshi, A., Walford, H.H., Kurten, R.C., Broide, D.H., and Aceves, S. (2015). Group 2 innate lymphocytes (ILC2) are enriched in active eosinophilic esophagitis. J. Allergy Clin. Immunol. *136*, 792–794.e3. https://doi.org/10.1016/j.jaci.2015.05.048.
78. Morgan, D.M., Ruiter, B., Smith, N.P., Tu, A.A., Monian, B., Stone, B.E., Virk-Hundal, N., Yuan, Q., Shreffler, W.G., and Love, J.C. (2021). Clonally expanded, GPR15-expressing pathogenic effector TH2 cells are associated with eosinophilic esophagitis. Sci. Immunol. *6*, eabi5586. https://doi.org/10.1126/sciimmunol.abi5586.
79. Yang, Y., Tresp, V., Wunderle, M., and Fasching, P.A. (2018). Explaining therapy predictions with layer-wise relevance propagation in neural networks. In ICHI, pp. 152–162. https://doi.org/10.1109/ICHI.2018.00025.
80. Minh, D., Wang, H.X., Li, Y.F., and Nguyen, T.N. (2022). Explainable artificial intelligence: a comprehensive review. Artif. Intell. Rev. *55*, 3503–3568. https://doi.org/10.1007/s10462-021-10088-y.
81. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., and Su, J.K. (2019). This looks like that: deep learning for interpretable image recognition. Adv. Neural Inf. Process. Syst. *32*. https://doi.org/10.48550/arXiv.1806.10574.
82. Ali, S., Abuhmed, T., El-Sappagh, S., Muhammad, K., Alonso-Moral, J.M., Confalonieri, R., Guidotti, R., Del Ser, J., Díaz-Rodríguez, N., and Herrera, F. (2023). Explainable artificial intelligence (XAI): What we know and what is left to attain trustworthy artificial intelligence. Inf. Fusion *99*, 101805. https://doi.org/10.1016/j.inffus.2023.101805.
83. Buenrostro, J.D., Wu, B., Litzenburger, U.M., Ruff, D., Gonzales, M.L., Snyder, M.P., Chang, H.Y., and Greenleaf, W.J. (2015). Single-cell chromatin accessibility reveals principles of regulatory variation. Nature *523*, 486–490. https://doi.org/10.1038/nature14590.
84. Gene Ontology Consortium (2026). The gene ontology knowledgebase in 2026. Nucleic Acids Res. *54*, D1779–D1792. https://doi.org/10.1093/nar/gkaf1292.
85. Wolf, F.A., Angerer, P., and Theis, F.J. (2018). SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. *19*, 15. https://doi.org/10.1186/s13059-017-1382-0.
86. Luecken, M.D., Büttner, M., Chaichoompu, K., Danese, A., Interlandi, M., Müller, M.F., Strobl, D.C., Zappia, L., Dugas, M., Colomé-Tatché, M., and Theis, F.J. (2022). Benchmarking atlas-level data integration in single-cell genomics. Nat. Methods *19*, 41–50. https://doi.org/10.1038/s41592-021-01336-8.
87. Svensson, V. (2020). Droplet scRNA-seq is not zero-inflated. Nat. Biotechnol. *38*, 147–150. https://doi.org/10.1038/s41587-019-0379-5.
88. Maan, H., Zhang, L., Yu, C., Geuenich, M.J., Campbell, K.R., and Wang, B. (2024). Characterizing the impacts of dataset imbalance on single-cell data integration. Nat. Biotechnol. *42*, 1899–1908. https://doi.org/10.1038/s41587-023-02097-9.
89. Ioffe, S., and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. International conference on machine learning. (PMLR) *37*, 448–456. https://proceedings.mlr.press/v37/ioffe15.html.
90. Nair, V., and Hinton, G.E. (2010). Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th international conference on machine learning (ICML-10), pp. 807–814.
91. Loshchilov, I., and Hutter, F. (2019). Decoupled weight decay regularization. In International conference on learning representations. https://doi.org/10.48550/arXiv.1711.05101.
92. Guo, C., Pleiss, G., Sun, Y., and Weinberger, K.Q. (2017). On calibration of modern neural networks. In Proceedings of the 34th international conference on machine learning - volume 70 ICML'17 (JMLR.org), pp. 1321–1330.
93. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. J. Mach. Learn. Res. *12*, 2825–2830.
94. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., et al. (2000). Gene ontology: tool for the unification of biology. Nat. Genet. *25*, 25–29. https://doi.org/10.1038/75556.
95. Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. R. Stat. Soc. Ser. B Methodol. *57*, 289–300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.
96. Gayoso, A., Lopez, R., Xing, G., Boyeau, P., Valiollah Pour Amiri, V., Hong, J., Wu, K., Jayasuriya, M., Mehlman, E., Langevin, M., et al. (2022). A Python library for probabilistic analysis of single-cell omics data. Nat. Biotechnol. *40*, 163–166. https://doi.org/10.1038/s41587-021-01206-w.

CellPress
OPEN ACCESS

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
GO_Biological_Process_2025	Gene Ontology Consortium[84]	http://www.geneontology.org/
ScType markers	Ianevski et al.[30]	https://github.com/IanevskiAleksandr/sc-type
PBMC10K	Zheng et al.[3]	https://scvi-tools.org/
PBMC30K	Ding et al.[63]	https://singlecell.broadinstitute.org/single_cell/study/SCP424/
AtlasRGC/ONC	Tran et al.[53]	https://singlecell.broadinstitute.org/single_cell/study/SCP509
Patch-seq RGC	Huang et al.[76]	GSE211038
TSCA Datasets	Madissoon et al.[64]	https://www.tissuestabilitycellatlas.org/
ICA	Domínguez et al.[8]	https://www.tissueimmunecellatlas.org/
AtlasEoE	Ding et al.[25]	https://singlecell.broadinstitute.org/single_cell/study/SCP1242/
Morgan EoE	Morgan et al.[78]	GSE175930
Clevenger EoE	Clevenger et al.[46]	GSE218607
Software and algorithms		
ProtoCloud	This paper	https://github.com/Ding-Group/ProtoCloud
Scanpy (v1.10.2)	Wolf et al.[85]	https://github.com/scverse/scanpy
scib-metrics (v1.1.6)	Luecken et al.[86]	https://github.com/theislab/scib
Seurat (v4.4.0)	Hao et al.[27]	https://satijalab.org/seurat/
CellTypist (v1.6.3)	Dominguez et al.[8]	https://github.com/Teichlab/celltypist
scANVI (v1.1.5)	Xu et al.[26]	https://scvi-tools.org/
TOSICA (v1.0.0)	Chen et al.[60]	https://github.com/JackieHanLab/TOSICA
SIMS (v3.0.6)	Gonzalez-Ferrer et al.[62]	https://github.com/braingeneers/SIMS
scPoli (v0.6.1)	De Donno et al.[61]	https://github.com/theislab/scPoli_reproduce
scGPT (v0.2.4)	Cui et al.[39]	https://github.com/bowang-lab/scGPT
scBERT (v1.0.0)	Yang et al.[37]	https://github.com/TencentAILabHealthcare/scBERT

METHOD DETAILS

The ProtoCloud model

We present the implementation details of ProtoCloud, a self-explaining generative model for cell type annotation. ProtoCloud embeds high-dimensional gene expression profiles of individual cells into a low-dimensional latent space and subsequently classifies cells based on their positions in that space. The model comprises four main components: a probabilistic encoder $\boldsymbol{g} : \mathbb{R}^G \to \mathbb{R}^d$, a probabilistic decoder $\boldsymbol{f} : \mathbb{R}^d \to \mathbb{R}^G$, a prototype matrix $\mathbf{P} \in \mathbb{R}^{(K \cdot M) \times d}$ and a single-layer linear classifier without the bias term $\boldsymbol{h} : \mathbb{R}^{(K \cdot M)} \to \mathbb{R}^K$. Here, G denotes the number of genes, d the dimensionality of the latent space, K the number of cell types, and M the number of prototypes per type ($M \geq 1$). The prototype matrix $\mathbf{P}$ contains $K \cdot M$ prototypes. For each cell type k, $\mathbf{P}_k \in \mathbb{R}^{M \times d}$ represents its set of M prototypes, with its jth prototype vector denoted by $\boldsymbol{p}_{k,j}$, where j indexes the prototypes within cell type k. The prototypes reside in the same latent space as the cell embeddings, and are randomly initialized and optimized during training to serve as learnable representations of cell types.

ProtoCloud takes raw Unique Molecular Identifier (UMI) counts of cells and cell type annotations as input, in the form of $\mathcal{D} = \{(\mathbf{x}_i, y_i)_{i=1}^N\}$ consisting of N cell-label pairs, during model training. Each cell i is represented by a vector $\mathbf{x}_i \in \mathbb{R}^G$ of G genes assigned to cell type $y_i \in \{1, \ldots, K\}$. We use $\mathrm{y}_\mathrm{i} \in \{0,1\}^\mathrm{K}$ to denote the one-hot encoding of the cell type label y_i. The probabilistic encoder $\boldsymbol{g}$ receives log-transformed $\mathbf{x}_i$ (specifically, $log(1+\mathbf{x}_i)$) as input, and outputs the parameters $\boldsymbol{\mu}_\mathbf{i} \in \mathbb{R}^d$ and $\log(\boldsymbol{\sigma}_\mathbf{i}) \in \mathbb{R}^d$ for a normal distribution. The low-dimensional representation $\mathbf{z}_i$ for cell i is then sampled from this distribution:

$$\mathbf{z}_i \sim q_{\boldsymbol{g}}(\mathbf{z}_i|\mathbf{x}_i) := \mathcal{N}(\mathbf{z}_i|\boldsymbol{\mu}_\mathbf{i}, \mathrm{diag}(\boldsymbol{\sigma}_\mathbf{i})) \quad \text{(Equation 1)}$$

Each low-dimensional embedding $\mathbf{z}_\mathbf{i}$ serves two purposes: it is used for cell type classification by comparing it with the prototype matrix $\mathbf{P}$ and feed into classifier $\boldsymbol{h}$, and it is also passed to the probabilistic decoder $\boldsymbol{f}$ to reconstruct the UMI counts $\widehat{\mathbf{x}}_\mathbf{i}$.

Single-cell data are noisy, containing both cell type information and frequently capturing cell type-irrelevant nuisance factors such as experimental batches. We thus partition the latent embedding $\mathbf{z}_i$ into two parts: the first half, $\mathbf{z}_i^1 = \left[z_{i,1}, z_{i,2}, \cdots, z_{i,d/2}\right]^\mathsf{T}$, is designed to specifically capture cell type information, while the second half, $\mathbf{z}_i^2 = \left[z_{i,d/2+1}, z_{i,d/2+2}, \cdots, z_{i,d}\right]^\mathsf{T}$, is designed to capture cell type-irrelevant factors. Consequently, only $\mathbf{z}_i^1$ is used for cell type classification, while both $\mathbf{z}_i^1$ and $\mathbf{z}_i^2$ are used by the decoder $\boldsymbol{f}$ to reconstruct the gene expression profile $\widehat{\mathbf{x}}_i$.

The classifier $\boldsymbol{h}$ takes a vector of similarities between cell i and the $K{\cdot}M$ prototypes in $\mathbf{P}$, and produces a probability distribution over the K cell types. We compute the similarity between cell i and the j-th prototype of cell type k using a scaled Cauchy distribution density function:

$$s_i(k,j) = \mathrm{sim}\left(\mathbf{z}_i^1, \boldsymbol{p}_{kj}^1\right) = \frac{1}{\sqrt{\left(\left\|\mathbf{z}_i^1 - \boldsymbol{p}_{kj}^1\right\|_2 \cdot \kappa\right)^2 + 1}}$$

where κ is a learnable, positive scalar parameter and $\boldsymbol{p}_{kj}^1 \in \mathbb{R}^{d/2}$ is the first half of $\boldsymbol{p}_{kj}$. This formulation assigns a maximum similarity score $s_i(k,j)$ of 1 when the embedding $\mathbf{z}_i^1$ exactly matches prototype $\boldsymbol{p}_{kj}^1$, and the similarity smoothly decreases as the Euclidean distance increases. The similarity $s_i(k,j)$ indicates the influence of prototype j of cell type k on the classifier prediction, and the maximum similarity can serve as a prediction certainty measurement (see section "Reliability and uncertainty estimation"). The classifier $\boldsymbol{h}$ takes the vector of similarity scores

$$\boldsymbol{s}_i = [s_i(1,1), s_i(1,2), \ldots, s_i(K,M)]^\mathsf{T} \in \mathbb{R}^{(K \cdot M)}$$

as input and is trained via cross-entropy (CE) loss:

$$\mathcal{L}_{CE} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{CE}(\boldsymbol{h}(\boldsymbol{s}_i), \mathbf{y}_i) \qquad \text{(Equation 2)}$$

The probabilistic decoder $\boldsymbol{f}$ is essential for providing cell-level interpretations by reconstructing the gene expression profiles from the latent embeddings. For scRNA-seq data, the UMI count distribution for a gene g in cell i is generally assumed to follow a negative binomial (NB) distribution.[87] Accordingly, the decoder outputs the mean $m_{i,g} \geq 0$ and the dispersion $r_{k,g} > 0$ parameters. Given the training data $\mathcal{D}$ with cell type labels, we thus use a cell type-specific dispersion parameter $r_{k,g}$ for gene g, to capture cell type-specific variability in gene expression. The mean parameter $m_{i,g}$ for gene g remains cell-specific. Specifically, the probabilistic decoder outputs a G-dimensional probability vector (non-negative and summing to 1), which is multiplied by the total UMI count n_i for cell i to obtain the mean vector $\boldsymbol{m}_i$. The cell type-specific dispersion $r_{k,g}$ is learned for each gene g and cell type k. Thus, the ProtoCloud model likelihood can be written as

$$p_{\boldsymbol{f}}(\mathbf{x}_i | \mathbf{z}_i, y_i) = \prod_{j=1}^{G} \mathrm{NB}\left(x_{ij} \middle| m_{ij}, r_{y_i j}\right)$$

ProtoCloud has a variational autoencoder (VAE) architecture backbone, and we need to calculate the evidence lower bound (ELBO) to optimize the probabilistic encoder ($\boldsymbol{g}$) and decoder ($\boldsymbol{f}$). The ELBO includes a Kullback–Leibler (KL) divergence term between the variational distribution $q_{\boldsymbol{g}}(\mathbf{z}_i|\mathbf{x}_i)$ (Equation 1) and a latent prior $p(\mathbf{z}_i)$. Conventionally, $p(\mathbf{z}_i)$ is a standard multivariate normal distribution. In our case, it becomes complicated because we want cells to be close to their cell-type prototypes. Moreover, we decompose the latent embedding into two components: $\mathbf{z}_i^1$ for cell type-specific information and $\mathbf{z}_i^2$ for cell type-irrelevant factors. To achieve these goals, we consider a mixture of VAEs with shared encoder and decoder networks. Each VAE mixture component has a prior centered on one of the M prototypes for each cell type k. Only the training cells with label k are used to calculate the KL-divergences for these M VAEs. The KL divergence for each prototype is weighted by its relative similarity within that class, $\tilde{s}_i(y_i,j)$.[56]

$$\tilde{s}_i(y_i,j) = \frac{s_i(y_i,j)}{\sum_{l=1}^{M} s_i(y_i,l)}$$

The overall KL divergence is then calculated as the weighted sum of the KL divergences for the cell type-specific and cell type-irrelevant components:

$$D_{KL,i} = \sum_{j=1}^{M} \tilde{s}_i(y_i,j) \cdot KL\left(\mathcal{N}\left(\boldsymbol{\mu}_i^1, \mathrm{diag}\left(\boldsymbol{\sigma}_i^1\right)\right) \,\|\, \mathcal{N}\left(\boldsymbol{p}_{y_i j}^1, \mathbf{I}_{d/2}\right)\right) + KL\left(\mathcal{N}\left(\boldsymbol{\mu}_i^{(2)}, \mathrm{diag}\left(\boldsymbol{\sigma}_i^{(2)}\right)\right) \,\|\, \mathcal{N}\left(\mathbf{0}, \mathbf{I}_{d/2}\right)\right)$$

where $\mathrm{KL}(q\|p)$ denotes the KL divergence between probability distributions q and p, $\boldsymbol{\mu}_i^{(2)}$ and $\boldsymbol{\sigma}_i^{(2)}$ are the second halves of the mean and standard deviation parameters of the latent normal distribution for $\mathbf{z}_i$, and $\mathbf{I}_{d/2}$ is the identity matrix with a dimensionality of $d/2$.

Combining these elements, we formulate the VAE objective function to maximize the ELBO:

$$\mathcal{L}_{ELBO} = \frac{1}{N}\sum_{i=1}^{N}\left[\mathbb{E}_{q_{\boldsymbol{g}}(\mathbf{z}_i|\mathbf{x}_i)}[\log p_{\boldsymbol{f}}(\mathbf{x}_i|\mathbf{z}_i, y_i)] - D_{KL,i}\right] \quad \text{(Equation 3)}$$

As both the encoder ($\boldsymbol{g}$) and the decoder ($\boldsymbol{f}$) networks are shared across all cell types and prototypes, ProtoCloud remains computationally efficient, comparable to conventional VAEs.

The orthogonal loss to help learn diverse prototypes for each class

Training ProtoCloud using the sum of the cross-entropy loss (Equation 2) and the negative ELBO (Equation 3) frequently encounters the prototype collapse problem, in which the M prototypes of cell type k converge to a single point, often near the center of the embeddings of class k cells. This collapse prevents the model from learning diverse representative cell prototypes that capture the intra-class diversity. To help mitigate this problem, we introduce an extra orthogonal loss

$$\mathcal{L}_{\text{orth}} = \frac{1}{K}\sum_{k=1}^{K}\left\|\overline{\mathbf{P}}_k^1\left(\overline{\mathbf{P}}_k^1\right)^{\mathsf{T}} - \mathbf{I}_M\right\|_F^2$$

here $\|\cdot\|_F$ denotes the Frobenius norm and $\overline{\mathbf{P}}_k^1 \in \mathbb{R}^{M\times(d/2)}$ is the first half of the centered prototype matrix for class k. Specifically, we compute the mean of the M prototypes for class k

$$\overline{\boldsymbol{p}}_k^1 = \frac{1}{M}\sum_{j=1}^{M}\boldsymbol{p}_{kj}^1$$

and subtract this mean from each prototype $\boldsymbol{p}_{kj}^1$ within that class to obtain a centered matrix. By encouraging the rows of $\overline{\mathbf{P}}_k^1$ to be orthogonal, this loss helps maintain multiple, distinct prototype directions within each class.

The atomic loss to help learn diverse and accurate prototypes

To further encourage cell embedding to be close to cell type-specific prototypes, we introduce an attractive force. Let mask_i be a binary vector of length $K{\cdot}M$ where $\text{mask}_i[M{\cdot}(k\text{-}1)+j] = 1$ if the prototype j of cell type k belongs to the same cell type as cell i, and 0 otherwise. The attractive force is defined as:

$$F_{attractive} = -\frac{1}{N}\sum_{i=1}^{N}\max(\boldsymbol{s}_i \odot \text{mask}_i)$$

where $\odot$ denotes element-wise multiplication. This force encourages each cell embedding $\mathbf{z}_i^1$ to be close to at least one of its M prototypes in its own class. However, a trivial solution could minimize this force by collapsing all cells to the same point, overlapping with all $K{\cdot}M$ prototypes.

To counteract such collapse and address potential issues arising from multiple prototypes per class and rare cell types, we add a repulsive force. As ProtoCloud uses multiple prototypes for each class, it's possible that some prototypes of cell type k are close to the embeddings of cells from other classes. Furthermore, in the presence of rare cell types, both their prototypes and the encoder may not be well trained to embed these cells effectively,[31] leading to misplaced prototypes. Such misplacements make it challenging to draw reliable cell-level interpretations. The repulsive force pushes away prototypes of other classes from the cell embeddings of class k cells

$$F_{repulsive} = \frac{1}{N}\sum_{i=1}^{N}\max(\boldsymbol{s}_i \odot \neg\text{mask}_i)$$

where $\neg\text{mask}_i$ is the negation of the binary mask vector mask_i. Together, the dynamics of attractive force and repulsive force—analogous to inter-molecular forces between atoms—are combined into the atomic loss

$$\mathcal{L}_{atomic} = F_{attractive} + F_{repulsive}$$

Shaping latent spaces through a two-stage curriculum and latent decomposition

Latent embeddings learned by variational autoencoders can also fluctuate, similar to the latent embeddings learned by conventional autoencoders, as observed by pioneers in deep learning such as Geoffrey Hinton, resulting in latent space fragmentation, where cell embeddings of the same type are widely separated and partitioned into disjoint regions by embeddings of other cell types. This fragmentation is primarily driven by technical factors such as batch effects, donor-specific variation, and sequencing depth, which can artificially separate otherwise biologically homogeneous cell populations. To learn more compact embeddings in which cells of the same cell type form a unified and cohesive region, we employ a two-stage training strategy, along with latent space decomposition, in ProtoCloud.

In the first stage, we train ProtoCloud by minimizing the loss function

$$\min_{g,f,P,h}\left(-\mathcal{L}_{ELBO}+\mathcal{L}_{CE}\right)$$

During this stage of training, the prototypes of the same class are encouraged to collapse toward a single vector, thereby drawing the corresponding cell embeddings together to form a unified, compact region around their prototypes. Notably, as classification utilizes only the first half of the latent dimensions, batch-related information is encouraged to be captured within the second half of the dimensions, effectively disentangling batch effects from cell type information.

In the second stage of training, we reintroduce prototype diversity through the orthogonality and atomic loss terms. The orthogonal and atomic losses are designed to generate repulsive forces among embeddings, promoting separation in the latent space and increasing the diversity of learned prototypes. The updated loss function in this stage then becomes

$$\min_{g,f,P,h}\left(-\mathcal{L}_{ELBO}+\mathcal{L}_{CE}+\mathcal{L}_{orth}+\mathcal{L}_{atomic}\right)$$

To facilitate feature (gene)-level interpretation of cell type annotation in the second stage, we add a penalty term L_1 to the weights of the first layer of the encoder. This penalty encourages sparsity by driving the weights of unimportant genes for cell type annotation toward zero. The strength of the L_1 regularization is set to the inverse of the number of elements in the first layer weight matrix.

By decoupling the training process into two stages, ProtoCloud progressively refines its representations: the first stage prioritizes inter-class diversity, while the second stage enhances intra-class diversity. After a ProtoCloud model has been trained, it can be used to annotate cells, taking only raw UMI count data of cells as input.

Prototypical relevance propagation for gene-level interpretation

To obtain gene (feature)-level interpretation and identify which genes drive ProtoCloud's predictions, we use prototypical relevance propagation (PRP).[55] For each cell type k and one of its associated prototypes $\boldsymbol{p}_{k,j}$, we first calculate the cell-prototype similarity for a given cell n assigned to cell type k:

$$\boldsymbol{\gamma}_{n,k,j}=\frac{1}{\sqrt{\left(\|\boldsymbol{p}^1_{k,j}-\boldsymbol{\mu}^1_n\|_2\cdot\kappa\right)^2+1.0}}$$

where $\boldsymbol{\mu}^1_n$ is the first half of the mean vector of the variational normal latent distribution $q_{\boldsymbol{g}}(\mathbf{z}_n|\mathbf{x}_n)$. The similarity vector $\boldsymbol{\gamma}_n\in\mathbb{R}^{(K\cdot M)}$ is backpropagated through the encoder following the layer-wise relevance propagation (LRP) rules. The initial relevance $\boldsymbol{R}^{(L+1)}$ is a $(K\cdot M)$-dimensional vector, with each element set to -1, except for the M elements corresponding to the prototypes of cell type k, which are set to either 1 (the prototype closest to cell n) or $1/M$ (the other prototypes). The number of hidden layers in the encoder network is L.

More specifically, let o_j represent the jth output of a fully connected layer l, before the layer's rectified linear unit (ReLU) activation, then

$$o_j=\sum_i a_i\cdot w_{i,j}$$

where $w_{i,j}$ is the weight connecting the ith input neuron with activation a_i at layer l, to the output neuron j at layer l+1. Thus, the relevance from the layer's output $R_j^{(l+1)}$ is propagated to its input $R_i^{(l)}$, according to the following equation:

$$R_i^{(l)}=\begin{cases}\sum_j\frac{a_i\cdot w_{i,j}}{\sum_t a_t\cdot w_{t,j}+\epsilon}R_j^{(l+1)}, & l=L\\ \sum_j\frac{(a_i\cdot w_{i,j})^+}{\sum_t(a_t\cdot w_{t,j})^+}R_j^{(l+1)}, & l<L\end{cases}\qquad\text{(Equation 4)}$$

The parameter $\epsilon=1e^{-6}$ is used by default. For convenience, we use the notation $(\cdot)^+=max(0,\cdot)$. We only consider positive contributions to the relevance for the encoder layers because the input gene expression values are non-negative, and we want to find genes that positively contribute to the prediction of cell types. The rules in Equation 4 are called the ϵ-rule (when $l=L$) and z^+-rule (when $l<L$), or equivalently the $\alpha\beta$-rule with $\alpha=1$ and $\beta=0$.[54]

The final relevance scores $R^{(1)}_{g=1:G}$ for all the G genes are normalized to the range [0,1] to facilitate direct comparison of relevance scores across different prototypes. To obtain a gene-level relevance score for cell type k, we weighted the relevance scores across all M prototypes associated with cell type k and all training cells with label k.

Data augmentation for training robust ProtoCloud models

Single-cell RNA-Seq data often exhibit class imbalance, with some cell types being extremely rare (e.g., accounting for <0.005% of the recovered cells) and dominant cell types.[25] This disparity is prevalent in data from various biological contexts, such as development, homeostasis, and cancer, posing significant challenges for machine learning algorithms.[88] A common approach to address this imbalance is oversampling, where rare class populations are duplicated to create more balanced training data. However, this simple approach can lead to overfitting of the rare populations, as it does not capture the underlying data distribution of a rare class.

To mitigate this class imbalance problem, we generate artificial cells for rare populations using statistical modeling of the gene count distribution for each gene in each rare cell type. Specifically, for each rare cell type (defined as having a proportion less than $0.5/K$, where K is the number of cell types), we generate artificial samples to ensure that the total number of cells from each rare population reaches $0.5\times N/K$, where N is the total number of cells in the original dataset. We sample an artificial cell whose UMI count vector $\tilde{\mathbf{x}} \in \mathbb{R}^G$ is drawn from a multinomial distribution. The distribution's rate probability vector parameter $\boldsymbol{r}$ is estimated from a randomly drawn UMI count vector $\mathbf{x}$ from a cell of the rare population:

$$\mathbf{r} = \frac{(\mathbf{x}+0.1)}{\sum_{g=1}^{G}(x_g+0.1)}$$

Adding 0.1 to each element of $\mathbf{x}$ ensures that $\boldsymbol{r}$ is a dense probability vector that sums to 1. By sampling from this multinomial distribution with probability vector $\boldsymbol{r}$, we can generate non-zero counts for genes that had zero counts in the original sample $\mathbf{x}$. The size parameter (number of trials) for the multinomial distribution is sampled uniformly from the interval:

$$\left[0.5 \times \sum_{g=1}^{G}(x_g+0.1), \sum_{g=1}^{G}(x_g+0.1)\right]$$

This data augmentation step ensures a more balanced representation of all cell types in the training data, thereby reducing the risk of overfitting to rare cell types.

Model architecture and training

By default, ProtoCloud employs an encoder with three hidden layers (1,024, 512, and 256 neurons, respectively) and a decoder with two layers (512 and 1,024 neurons, respectively). Each layer consists of a bias-free linear transformation followed by batch normalization[89] and ReLU activation.[90] The model uses a 20-dimensional latent space and maintains six prototypes per class. Training is performed for 100 epochs using the AdamW optimizer[91] with a learning rate of 1×10^{-3}. The mini-batch size is 128 for datasets with fewer than 100,000 cells and 1,024 for larger datasets. These hyperparameters remain constant across experiments unless otherwise specified.

To estimate the time required to train ProtoCloud for 100 epochs, we down-sampled one dataset to various numbers of cells (10,000, 50,000, 100,000, 200,000, and 300,000 cells). All experiments were conducted on a server with an NVIDIA Tesla V100 16GB GPU on a single node. Training ProtoCloud is efficient, even for relatively large datasets with 300,000 cells when using adaptive batch sizes (128 for smaller datasets with fewer than 100,000 cells and 1,024 for datasets exceeding 100,000 cells). The training times across the different dataset sizes are shown in Figure S1A. For large datasets, the training time can be further decreased by reducing the training epochs (default: 100 epochs).

Ablation study

To systematically assess the contribution of each component in ProtoCloud, we conducted an ablation study examining three key design choices: the disentangled latent space, the two-stage curriculum, and the inclusion of two auxiliary losses (orthogonal and atomic losses). These studies evaluate model performance through multiple complementary perspectives, including batch effect separation, prototype embedding distributions, and similarity score ranges.

We evaluate the quality of prototype embeddings using Euclidean distances in the first half of the embedding space. Our evaluation differs from conventional clustering metrics such as the silhouette coefficient, which prefer large inter-class distances and small intra-class distances. Instead, we require both distances to be substantial to effectively capture cell-type distinctions while preserving within-type biological variation.

Disentangled vs. unified latent space

Our standard architecture models the first half of the latent dimensions of a cell with a multivariate normal distribution that is constrained to be close to its prototypes and uses these dimensions for similarity score computation, while the remaining dimensions are encouraged to follow a standard multivariate normal distribution. The unified variant removes this architectural separation in the KL divergence, encouraging all latent dimensions of a cell to follow a multivariate normal distribution that is constrained to be close to its prototypes and using all latent dimensions for the similarity calculation.

Two-stage vs. single-stage curriculum

In the standard two-stage curriculum, ProtoCloud is initialized and trained with only the baseline VAE losses, $\mathcal{L}_{ELBO}$ and $\mathcal{L}_{CE}$, for the first 30 epochs. The model then enters the second stage after 30 epochs, where the atomic loss and orthogonal loss are added. The single-stage training baseline incorporates all loss components from initialization, training end-to-end for 100 epochs.

Loss function component ablation

We evaluated three loss configurations against our standard formulation. The baseline VAE configuration uses only $\mathcal{L}_{ELBO}$ and $\mathcal{L}_{CE}$; the atomic-only version adds the atomic loss, $\mathcal{L}_{atomic}$, to the baseline VAE losses; and the orthogonal-only version adds the atomic loss, $\mathcal{L}_{orth}$, instead.

Reliability and uncertainty estimation

ProtoCloud uses similarity scores as a more interpretable and robust measure of uncertainty than softmax probabilities, which often saturate toward extreme values.[92] We quantify prediction confidence using the maximum similarity score between each cell's first-half latent embedding and the prototypes.

Dichotomous certainty thresholds are determined at the end of training based on the training data. These thresholds are computed separately for each cell type, set at the 10th percentile of similarity scores between training cells and their corresponding class prototypes. Predictions with similarity scores exceeding their respective cell type thresholds are marked as "certain", indicating strong confidence in the assigned annotation. Certain predictions that disagree with the original labels are categorized as Type 1 classifications. Conversely, predictions below this threshold are marked as "ambiguous". Ambiguous predictions that disagree with the original labels are categorized as Type 2 classifications. As we adopt conservative thresholds, "ambiguous" reflects insufficient confidence for re-annotation rather than incorrect predictions.

To further improve reliability and interpretability, we transformed similarity scores into calibrated certainty scores through an adaptive calibration approach. The test data were randomly partitioned into two subsets: 25% were used to fit a calibrator using ProtoCloud predictions, similarity scores, and corresponding cell type labels, while the remaining 75% were used to validate the resulting calibrated scores. First, we trained a global isotonic regression[75] calibrator on the entire dataset to establish a baseline certainty mapping, which outputs a calibrated probability for each prediction. For rare cell types (see the "Data augmentation for training robust ProtoCloud models" section), the global calibration model was employed to prevent overfitting and ensure stable certainty estimates. For cell types with sufficient samples, we further trained cell type-specific calibrators to map similarity scores to calibrated certainty values.

Differentially ranked highly relevant genes

The genes for each cell type are ranked according to their relevance scores from high to low. Differentially ranked HRGs between two cell types are identified using two criteria: (1) ranked top t (default 20) in exactly one of the interested cell types, or (2) showing a ranking difference greater than u (default 200) between the two cell types.

Baselines for comparison

Seurat v4.[27] We used its reference-based label transfer functionality, following the workflow described in the Seurat integration tutorial (https://satijalab.org/seurat/articles/integration_mapping). Specifically, we employed the *FindTransferAnchors()* function with the projected PCA option to identify correspondences between reference and query datasets, followed by the *TransferData()* function to propagate cell type labels to the query dataset.

scANVI.[26] We utilized the *scvi.model.scANVI* API from the Python package scvi-tools (v1.1.5). Following best practices, we first trained an scVI model on the dataset for a maximum of 400 epochs. The resulting pre-trained scVI model was then used to initialize the scANVI model, which was subsequently trained for an additional 20 epochs to refine the annotations. For the time-course RGC ONC data analysis, the initial model was trained on labeled control data and used to predict labels for the cells from the subsequent time point. These predictions were then incorporated into the previous training set to train the next version of the model. All predictions were added to the training set as the model lacks a confidence output. This process was repeated sequentially through all time point data.

CellTypist.[8] CellTypist models were trained using the SGD algorithm with a maximum of 500 iterations. For the time-course RGC ONC data analysis, we used CellTypist with an iterative training approach: the initial model was trained on labeled control data and used to predict labels for the cells from the subsequent time point. Predictions with corresponding confidence scores greater than 0.5 were then incorporated into the previous training set to train the next version of the model (the fraction of confident predictions was small; Table S4). This process was repeated until the cells from every time point had been predicted.

TOSICA.[60] We implemented TOSICA using its standard workflow. For all datasets involving human cells, the pre-prepared human GO ('human_gobp') mask was used. For the AtlasRGC dataset, the corresponding mouse GO mask ('mouse_gobp') was used.

SIMS.[62] Following the recommendation for typical convergence, the model was trained for 10 epochs on each dataset. All other parameters were maintained at their default values.

scPoli.[61] The method requires batch information as input data. The model was trained for a total of 100 epochs, which included an initial 40 epochs of pre-training, as specified.

scGPT.[39] Due to the significant computational complexity associated with fine-tuning large foundation models, we employed scGPT in a zero-shot inference setting. This approach leverages the pre-trained whole-human scGPT model to generate cell embeddings and annotations for our query datasets without any additional training. For the AtlasRGC dataset, genes were first mapped to human orthologs prior to analysis.

scBERT.[37] We utilized the pre-trained scBERT model and fine-tuned it on our labeled reference data for 10 epochs, following the standard pre-train and fine-tune paradigm of the method. Due to scBERT's computational complexity and gene constraints from its pretrained model, we obtained results only for the PBMC10K, PBMC30K, AtlasRGC, TSCA lung, and TSCA esophagus datasets.

Evaluation metrics

We evaluated the performance of cell type classification using three standard metrics: accuracy, macro F1 score, and Cohen's Kappa coefficient.[65] Accuracy offers an overall view of model performance, macro F1 score is more sensitive to minority classes by treating each class equally, and Cohen's Kappa coefficient (κ) quantifies the agreement between predicted and true labels. More specifically, κ corrects for chance agreement:

$$\kappa = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

where p_o is the classification accuracy, or the empirical probability of agreement on the label assigned to a cell, and p_e is the expected classification accuracy when both true labels and predicted labels are randomly assigned to cells. Thus, Cohen's Kappa coefficient is a function of the classification accuracy. The advantage of κ over accuracy is that κ is sensitive to rare classes. We used scikit-learn (v1.5.1)[93] to calculate κ. Together, these three metrics (with higher values indicating better performance) ensure reliable and fair assessment across all cell types, including rare populations.

To assess batch separation in the latent space, we used the batch entropy score (BES) and the scib-metrics framework (v0.5.1).[86] BES quantifies batch mixing based on the composition of k-nearest neighbors (k-NN). Let B be the total number of batches, $p_{i,b}$ represent the proportion of the neighbors of cell i belonging to batch b. We then averaged the entropy across all cells to obtain the overall metric:

$$\mathrm{BES} = -\frac{1}{N}\sum_{i=1}^{N}\left(\sum_{b=1}^{B} p_{i,b}\log\left(p_{i,b}\right)\right)$$

where N is the total number of cells. A BES value close to $log(B)$ indicates greater batch mixing and weaker batch effects, whereas a lower value (with a minimum of 0) suggests stronger batch separation within the latent space.

Additionally, we applied the scIB metrics to evaluate biological conservation and batch correction in the latent space. Biological conservation was evaluated using five complementary metrics: Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI) (computed using K-means clustering) to measure cluster agreement; the isolated label F1 score to assess the preservation of rare cell types; and cell-type Adjusted Silhouette Width (ASW) along with cell-type LISI (cLISI) to quantify neighborhood purity. Batch correction performance was assessed using batch ASW, integration LISI (iLISI), and the k-nearest neighbor batch effect test (kBET) to measure local mixing; graph connectivity to evaluate the structural integrity of the neighbor graph; and principal component regression (PCR) comparison to quantify the variance explained by batch covariates.

To assess the biological relevance of the top-ranked HRGs, we employed three complementary metrics: gene signature scoring, cell type specificity quantification, and pathway enrichment analysis. To show whether the HRGs identified by ProtoCloud capture cell type-specific transcriptional patterns, we calculated the gene signature score for each cell type's top HRGs across all cells. This analysis was performed using the *sc.tl.score_genes* function from the Scanpy Python library.[85] To further investigate the gene-level specificity of HRGs, we compared the top-ranked HRGs against a curated set of known markers from scType.[30] We used the Tau index (τ), a widely adopted metric for measuring tissue or cell type specificity in transcriptomic studies.[72] The Tau index is defined as:

$$\tau = \frac{\sum_{k=1}^{K}\left(1 - \hat{x}_k\right)}{K - 1}$$

where K denotes the number of cell types, and $\hat{x}_k$ represents the average expression level of a gene in cell type k, normalized by the maximum average expression across all cell types. The Tau index ranges from 0 to 1, where $\tau = 0$ indicates ubiquitous expression across all cell types, and $\tau = 1$ indicates highly specific expression that the gene is strongly expressed in only one cell type and is nearly absent from all others. To show the biological functions associated with HRGs, we performed Gene Ontology (GO) enrichment analysis[94] targeting biological processes (BP). Specifically, we utilized the *GO_Biological_Process_2025* library[84] to query the HRGs identified for each cell type. Enrichment significance was assessed using Fisher's exact test, followed by the Benjamini-Hochberg[95] procedure to adjust the p-values for multiple comparisons (False Discovery Rate, FDR). We reported the top enriched biological processes for each cell type to characterize the functional relevance of identified HRGs.

To evaluate the extent to which the certainty estimates are calibrated, we employed two metrics: the Brier score[73] and the expected calibration error (ECE).[74] The Brier score quantifies the overall probabilistic accuracy by calculating the mean squared error between the predicted certainties and the actual outcomes:

$$\mathrm{BS} = \frac{1}{n}\sum_{i=1}^{n}\left(o_i - p_i\right)^2$$

where o_i is the outcome of the prediction at instance i (1 if the prediction is correct, otherwise 0), and p_i is the predicted probability. To specifically measure the discrepancy between the predicted confidence and empirical accuracy, we utilized the ECE. This metric partitions the predictions into a number of bins ($B = 20$ by default) based on their confidence scores. The goal is to calculate the weighted average of the absolute difference between the mean predicted confidence and the fraction of positive outcomes within each bin:

$$\mathrm{ECE} = \sum_{b=1}^{B} \frac{n_b}{N} |\mathrm{acc}(b) - \mathrm{conf}(b)|$$

where B is the number of bins, n_b is the number of samples in bin b, acc(b) is the accuracy, and conf(b) is the average model confidence (e.g., predicted similarities) within the bin. While both metrics assess calibration, they capture distinct aspects: the Brier score emphasizes the accuracy of probability estimates, whereas the ECE emphasizes the alignment between confidence and correctness. Both metrics range from 0 to 1, where a value of 0 indicates perfect calibration.

Data collection and processing

For data preprocessing, mitochondrial and ribosomal protein-coding genes are removed from all datasets. When applying the trained ProtoCloud models to unseen data (EoE studies; see below) or continuing training with a new dataset (RGC ONC studies; see below), only the genes shared with the data used for training the pre-trained ProtoCloud models are utilized. Missing genes in the test data are filled with zeros.

PBMC10K

We used the preprocessed PBMC dataset[3] downloaded from scvi-tools.[96] This dataset includes 12,039 cells, 3,346 genes, and two batch groups, spanning nine annotated cell types. No filtering was applied except for removing the cells labeled as ''unknown''.

PBMC30K

This PBMC dataset was derived from two experiments, each employing multiple scRNA-seq methods.[63] The original dataset consisted of 30,495 cells across ten cell types. We filtered out cells with fewer than 1,000 counts and genes with fewer than 200 total counts. The ''unassigned'' cells were removed from the data, resulting in 14,978 cells that passed filtering, with the top 3,000 HVGs selected for analysis.[85]

AtlasRGC

An atlas of mouse retinal ganglion cells (RGCs) in the adult retina.[53] The dataset comprises three batches and a total of 45 RGC subtypes, with cell proportions ranging from 8.4% (C1) to 0.15% (C45). We excluded cells with fewer than 1,000 total counts and cells labeled as ''unknown'', resulting in 35,699 cells. Genes with a total count below 500 were removed, and the top 3,000 HVGs were selected for analysis.

Time-course RGC datasets

The time-course RGC dataset, profiling retinal ganglion cells following optic nerve crush, was obtained from the same study[53] as AtlasRGC. This dataset includes varying numbers of cells at each time point (Table S4), with ''unassigned'' cells after 0 days post-crush (0 dpc). In our experiment, only cells from 0 dpc were used to train the initial model, based on the top 3,000 HVGs identified from the 0 dpc data. Cells from subsequent time points were then incorporated using the same set of 3,000 HVGs.

Patch-seq RGC

The raw single cell RNA-Seq data collected from Patch-seq technology, featuring 472 cells and 55,542 genes.[76] We applied the ProtoCloud model trained on AtlasRGC to this dataset. The top 3,000 HVGs from AtlasRGC were used, of which 2,924 were present in the Patch-seq RGC dataset.

TSCA datasets—Lung, esophagus, and spleen

These datasets[64] are from three human primary tissues, each exhibiting distinct levels of sensitivity to ischemia. The spleen, considered the most stable, contains 94,257 immune cells spanning 30 cell types. The esophageal mucosa includes 87,947 cells from 19 cell types, while the lung, identified as the least stable, comprises 57,020 cells across 28 cell types. For all three datasets, we filtered out cells with fewer than 1,000 counts and removed genes with fewer than 500 total counts. The top 3,000 HVGs were selected for analysis for each dataset. For comparative models requiring batch information, patient identity was used as the batch variable, resulting in five, six, and five batches for the spleen, esophageal mucosa, and lung datasets, respectively.

ICA

The cross-tissue Immune Cell Atlas dataset is part of the Human Cell Atlas project.[8] It includes over 300,000 immune cells, isolated from 16 different tissues obtained from 12 deceased adult donors. Organ origins are used when batch information is needed by comparative models. Cell identities were annotated using CellTypist, resulting in 43 cell subtypes. The top 3,000 HVGs were selected for analysis.

AtlasEoE

The esophageal cell atlas[25] consists of 421,312 scRNA-seq profiles. These samples were taken from 15 EoE patients (eight in active disease state and seven in remission) and seven healthy participants. The dataset includes 60 prevalent cell types grouped into major categories (e.g., epithelial cells, stromal cells, monocytes/mac/DC cells, B cells, T/NK/ILC cells). Additionally, the atlas has 12 rare

cell subsets, comprising 1,215 cells in total. For methods requiring batch information, donor identifiers were treated as batch labels, resulting in 22 batches. For this dataset, we used the highly variable genes (G = 2,956) provided by the authors.

Clevenger et al. dataset

This dataset[46] comprises 151,519 esophageal cells from six healthy donors, six patients with EoE, and four patients with gastroesophageal reflux disease (GERD). The authors identified eight major cell populations and focused on epithelial cells in their study. We applied the ProtoCloud model, trained on the AtlasEoE data, to this dataset.

Morgan et al. dataset

This dataset[78] comprises 14,242 esophageal cells collected from pediatric EoE patients in active or remission EoE, encompassing eight major cell types. We applied the ProtoCloud model, trained on the AtlasEoE data, to this dataset.