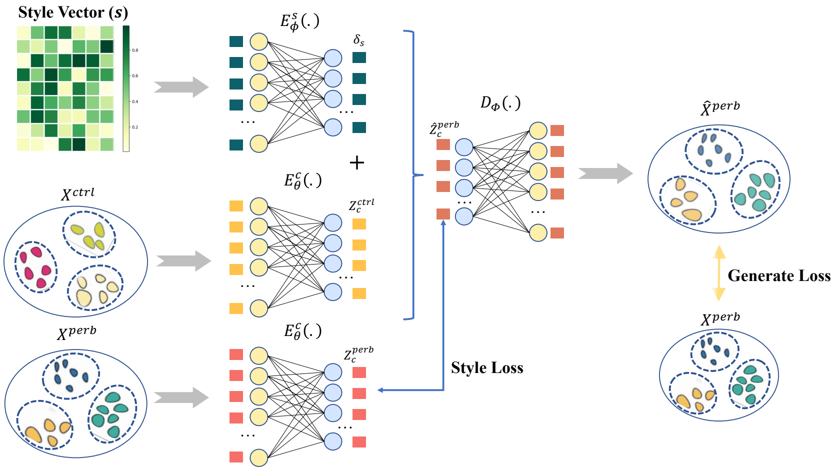

Fig. 1. scPerb predicts gene expressions of perturbed cells. scPerb was designed to predict gene expressions in perturbed cells and combines the principles of both style transfer and VAE. With the perturbed and control dataset as inputs, the content encoder projected the data into latent space. Differences between the latent representations of the perturbed dataset and the control dataset were captured by a Style Vector (s), which enabled transferring from the perturbed style to the control style. Such Style Vector was initiated with a random vector and updated via a style encoder, which learned the style of the perturbed dataset and transferred it to the control dataset by adding it to the latent representation of the control dataset. By minimizing the differences between both latent representations and gene expressions between predicted perturbed data and real perturbed data, scPerb transferred the control style to the perturbed style and predicted the gene expression of perturbed cells.

2.2 Decoder

In the decoder part, scPerb reparametrized the latent variable from the estimated posterior distribution $Z_c^{ctrl} \sim N(\mu^{ctrl}, \sigma^{ctrl})$ and $Z_c^{perb} \sim N(\mu^{perb}, \sigma^{perb})$. Unlike the standard VAE, which directly reconstructed the output $\hat{X}^{perb}$ from the latent variable Z_c^{ctrl} and Z_c^{perb} , scPerb converted the representation of the control data Z_c^{ctrl} to the latent representation $\hat{Z}_c^{perb}$, and generated the predicted perturbed data from decoder D_ϕ:

$$\hat{X}^{perb} = D_\phi(\hat{Z}_c^{perb})$$

Note that our task was to predict the perturbation of the cell types using the control dataset, instead of generating the samples from Z_c^{perb} and Z_c^{ctrl} as the original VAE, we only used $\hat{Z}_c^{perb}$ to generate $\hat{X}^{perb}$. Therefore, our $GeneratedLoss$ was:

$$GeneratedLoss = SmoothL1loss(X^{perb}, \hat{X}^{perb})$$

2.3 Loss function

The final objective function consisted of the $Generatedloss$, $StyleLoss$, and the KL regulation terms.

$$Loss = w_1 StyleLoss + w_2 KLLoss^{ctrl} + w_3 KLLoss^{perb} + w_4 Generatedloss$$

3 DATASETS AND PREPROCESS

We obtained the PBMC-Zheng dataset from Zheng et al. [31]. After removing the megakaryocyte cells that had uncertainly assigned labels, we log-transformed and normalized the data and selected the top 7,000 highly variable genes.

Kang et al. published a dataset from PBMCs including both control and perturbed cell types [25]. Among these data, we extracted the average of the top 20 cluster genes, which has 6,998 genes in total, from seven cell types,

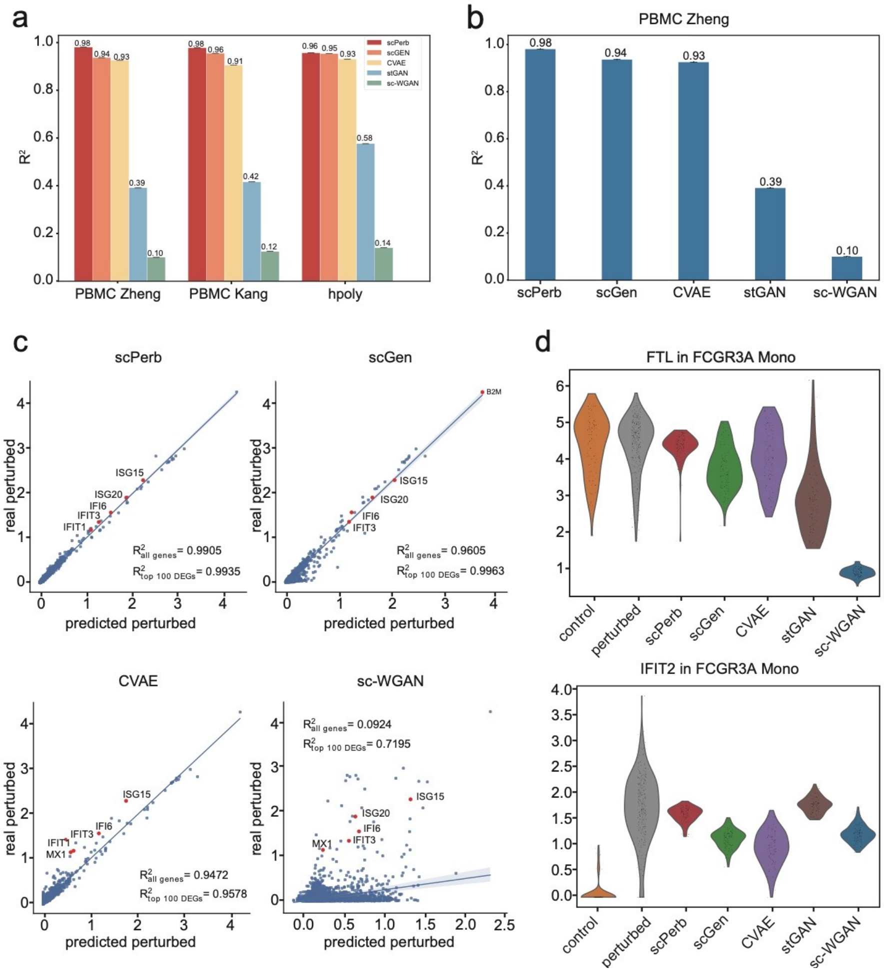

Fig. 2. The Overall result of scPerb. **a**: Comparison of R^2 values across all benchmarking methods; **b**: Bar plots showed the R^2 value of all methods in the PBMC-Zheng dataset [31]; **c**: Scatter plot showed the correlation between real and predicted gene expression of 7,000 genes by scPerb and other three benchmarking methods in CD4-T cells, and the five red dots represented the top five DEGs; **d**: The distribution of the control dataset, perturbed dataset, and the prediction of all methods in one of the least DEGs (*FTL*), and one of the top DEGs (*IFIT2*).

0.91. Similarly, the stGAN and sc-wGAN only had an average R^2 score of 0.42 and 0.12, respectively, in this dataset. Finally, we applied scPerb to the H.poly dataset and still got a 0.96 average R^2 score, followed by the scGen, CVAE, stGAN, and sc-wGAN with the average R^2 score of 0.95, 0.93, 0.58, 0.14. When comparing their results in a specific cell type, scPerb consistently outperformed other benchmarking methods (**Fig. 2b**). For example, in CD4-T cell type, one of the most numerous cell types in the PBMC-Zheng dataset, scPerb achieved a superior R^2 score of 0.99, which was much better than scGen, CVAE, stGAN, and sc-WGAN (R^2 score: 0.96, 0.95, 0.16, and 0.09) respectively.

In particular, we evaluated the performance of the proposed scPerb and the other benchmarking methods at the gene level. In **Fig. 2c**, we illustrated the prediction of our scPerb and the performance of the other three benchmarking methods in CD4-T cells from the PBMC-Zheng dataset. The scatter plot demonstrated that scPerb got the average R^2 score of 0.9905 when we used all the genes in this cell type. The performance could go up to 0.9935 when we only consider the top 100 DEGs. In comparison under the same setting, scGen achieved the average R^2 score of 0.9605 over all genes and 0.9963 on the top 100 DEGs. Our scPerb could outperform CVAE (average R^2 score of all genes = 0.9472, average R^2 score of top 100 DEGs = 0.9578) and sc-WGAN (average R^2 score = 0.0924, average R^2 score = 0.9578) on both the evaluation criteria. Specifically, DEGs including *IFIT1*, *IFIT3*, *IFI6*, *ISG20*, and *ISG15*, showed the best performance.

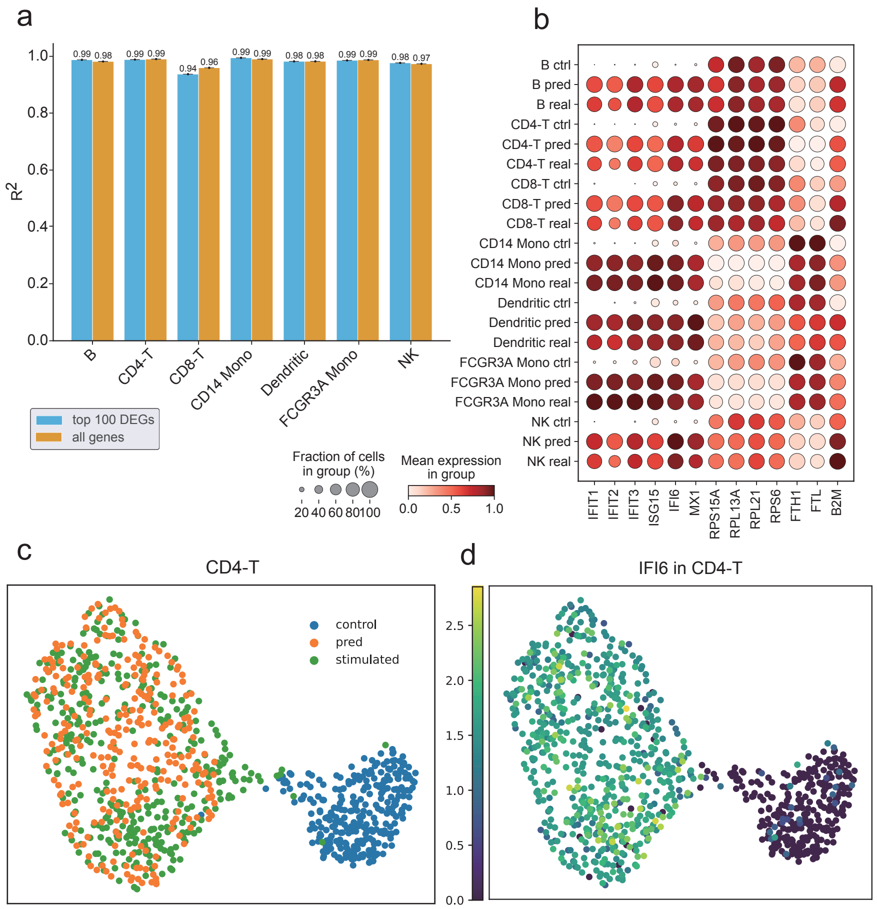

Fig. 3. Result of scPerb in PBMC-Zheng dataset. **a**: Grouped boxplot showed the result of scPerb in R^2 values in all genes and the top 100 DEGs in every cell type in the PBMC-Zheng dataset; **b**: Dot plot illustrating the mean gene expression in each cell type of control, real perturbed, and predicted perturbed condition, including the top DEGs and the least DEGs; **c-d**: UMAP [35] visualizations depicted the condition distribution of the overall CD4-T cell type in the PBMC-Zheng dataset and the expression pattern of *IFI6*, one of the top DEGs in the CD4-T cells.

zero values, scPerb made a better prediction than any other method, capturing the mean of the ground truth. In this case, the prediction of other methods barely captured the mean of the real perturbed data. (**Fig. 4b**) The Wilcoxon test can further explain the difference between the prediction and the real perturbed cells in the *MT2A* genes: only scPerb achieved a P value of 0.8785, meaning that the difference between the prediction of scPerb and the real perturbed data was not statistically different; however, all other methods including scGen, CVAE, and both GAN-based methods resulted in an adjusted P value far less than 0.0001, showing a significant difference between their predictions and the real perturbed data (**Fig. 4c**). Besides, the dot plot (**Fig. 4d**) showed that scPerb could get robust prediction no matter whether the original control gene expression was lower (for example the *IFIT1* gene), approximately the same (for example the *RPL13A* gene), or higher than (for example the *FTH1* gene) the ground truth. Moreover, it is worth noting that the prediction of scPerb correlated better with the real perturbed data, especially the top 5 DEGs (the red dots shown in **Fig. 4e**); and the R^2 values of scPerb (0.9950 and 0.9956 for all genes and the top 100 DEGs) were also higher than all the other benchmarks including scGen, CVAE, and sc-WGAN.

5.4 scPerb has robust results across different datasets

In the H.poly dataset [26], scPerb maintained superior performance with robust predictive capacity. For the cell types in the H.poly dataset, scPerb gained an average of R^2 as 0.96, which was better than the scGen and CVAE

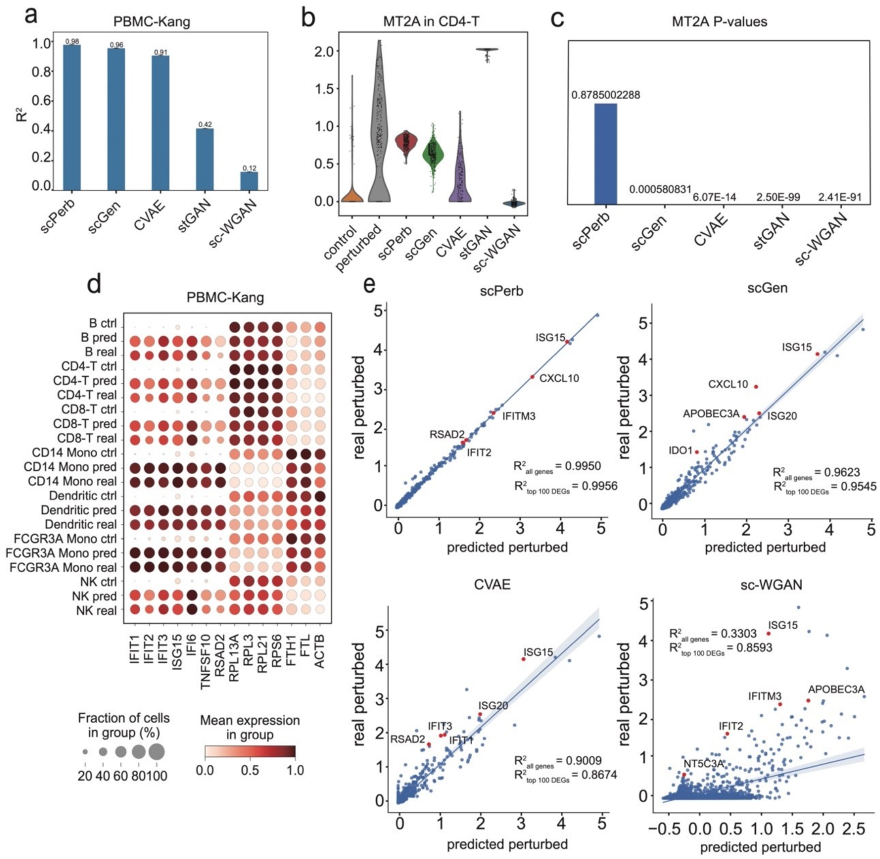

Fig. 4. Result of scPerb in PBMC-Kang dataset. **a**: This bar plot compared the R^2 values of all the methods within the PBMC-Kang dataset, while central values represented the mean R^2 values across all 7 cell types in the dataset; **b-c**: Comparing the distribution of all the methods in the *MT2A* gene in CD4-T cells in the PBMC-Kang dataset. Center values in **Fig. 4c** were the adjusted P values comparing the prediction of each method to the ground truth by using the Wilcoxon test [34]; **d**: A dot plot comparing the mean gene expression of all 7 cell types and all three conditions in the most and least DEGs in the PBMC-Kang dataset; **e**: The correlation of the mean expression of all 6,998 genes in FCGR3A Mono cells. It compared predictions from three of the best benchmark methods and scPerb against the ground truth, with shaded lines representing the 95% confidence interval of the regression estimate.

(scGen = 0.95, CVAE = 0.93), much better than stGAN and sc-WGAN (stGAN = 0.38, sc-WGAN = 0.14). The line plot in **Fig. 5a** also illustrated that scPerb maximized its difference in R^2 compared with other methods in Tuft cells, having R^2 = 0.94. While other VAE-based benchmarks had worse performance (scGen = 0.91, CVAE = 0.84). **Fig. 5a** also showed that all VAE-based methods (scPerb, scGen, CVAE) had a much better result than GAN-based methods (sc-WGAN, stGAN). In 7 out of 8 other cell types, scPerb showed superior performance than the benchmarking methods. Even for the endocrine cell type, in which scPerb presented less outperformance, it still achieved R^2 as 0.87, which was comparable with scGen (R^2 = 0.89).

Moreover, scPerb made better predictions in this dataset, especially in the Enterocyte. Progenitor cells. In **Fig. 5b**, the distance between the prediction (green dot) and real perturbed data (orange dot) was closer than the distance between the perturbed dataset to the control dataset (blue dot). For the other benchmarks, the VAE-base methods, scGen (**Fig. 5c**) and CVAE (**Fig. 5d**) could not easily divide the control data samples from the prediction and perturbed data, so their prediction resulted somewhere in between the control data samples and the perturbed data samples. And for the GAN-based methods, as shown in **Fig. 5e** for stGAN and **Fig. 5f** for sc-WGAN, the predictions were notably distant from both the control and perturbed datasets.

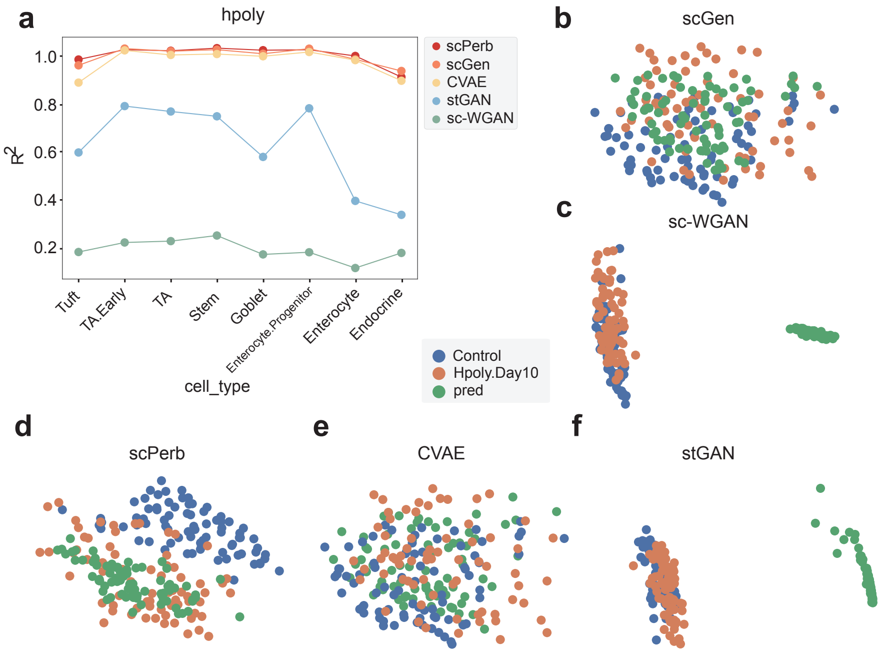

Fig. 5. The result of scPerb in the H.poly dataset a: Line plot using R^2 to compare the outcomes of all the methods; **b-f**: The UMAP visualization of the control (blue dots), perturbed (orange dots), and predicted (green dots) condition of all the methods.

6 DISCUSSION

scPerb is a generative model that dynamically transfers the gene expression in the control dataset into the reliable perturbed dataset. The encoder of scPerb projects the raw control gene data into the high-dimensional latent space. scPerb aggregates it with the dataset-specific styles to generate a high-quality representation for the perturbed dataset. Based on the representation, the decoder from scPerb can reconstruct gene expressions that are correlated with the mean of the perturbed dataset. The experiments demonstrate that scPerb can capture the latent content features and generate stable dataset-specific styles across different cell types and data from multiple studies. Moreover, the quantitative evaluation indicated the performance of scPerb outperforms four representative benchmarks, having state-of-the-art results in three different datasets.

Compared with traditional works [21-24], scPerb is a data-driven algorithm that can fully explore the gene expression in the raw dataset and does not rely on solid domain priors. On the opposite, the traditional works extract the principal components and build up a graph-based model in the low-dimensional manifold. Such methods rely heavily on the experienced domain knowledge, and lack of generalization abilities. Compared with other data-driven algorithms, scPerb incorporates the stableness from the VAE settings and exploits the advantage of the GAN to generate high-quality samples.

However, minor problems still exist. In Endocrine cells in the H.poly dataset, one of the cell types containing the fewest cells in the H.poly dataset (163 in 5,059), scPerb makes predictions slightly worse than scGen [23]. Using R^2 values as a criterion, scGen results in 0.89 while scPerb only results in 0.87. Note that scGen only calculates a fixed liner vector while scPerb uses style transfer, in this case, the problem of “overfitting” exists. However, such cases are very rare and scPerb can still outcompete “simple” methods like scGen in other cases when the data is small. In Tuft cells, also one of the cell types containing the fewest cells in the H.poly dataset (248 in 5,059), scPerb achieves a R^2 value of 0.94 while scGen only gets 0.91.