# SpaPerb

SpaPerb, equipped with the variational auto-encoders to estimate the latent content variable, and the dataset-related style, generate new samples from the control and stimulus dataset:

Then we applied the decoder to generate the new data sample based on the extracted feature: SpaPerb aims to transfer the gene expressions from the control dataset to the stimulus dataset.

There are two ways to achieve this aim.

The first way is to make the latent representation of the control and stimulus data as close as possible. Inspired by the style transfer GAN, we style-transferred the latent representation of the control data into stimulus data. We add a delta vector to the to achieve the goal. The delta vector is generated by projecting a delta vector from a delta encoder . By using a delta encoder, is able to learn the difference between and much better than using a constant vector. Finally, to decrease the bias of the delta vector, is generated by randomly choosing values that ranged instead of using a vector with constant values.

Therefore, the process is shown below:

Another way is to make the generated image similar to the stimulus dataset.

This process differs from the original VAE and other methods. As these methods apply a reconstruction loss, that is, to compute a loss between and .

In this case, there are two losses:

However, in this case, we want to emphasize the style transfer from to , the reconstructed data should be similar to the original data , so the reconstruction loss will be:

In our model, SpaPerb, we combine these two methods above and generated a better result than only using one of these models. In our model, we first randomly chose a vector with a size of input from ; then project it through a delta encoder to get the latent representation of the difference between control and stimulus datasets; finally, we compute a reconstruction loss between the generated and the real stimulus images.

Then by using VAE, we need to know the possibility curve of the graph; that is, we need to maximize , or in other words, .

Therefore, we have to transform this equation to solve this problem. Here, we create a new function to help us better achieve the goal:

In this case, as , .

Also, as , is independent to , so when optimizing to minimize , the will gradually become closer to , or in other words, the lower bound of .

Therefore, in order to maximize , we need to maximize the lower bound .

To maximize the lower bound, we need further processing:

Therefore, .

By transforming the equation above, we can get the final equation to compute the loss for the model.

And our goal is to construct a neural network to compute a function f to maximize ELBO.

By setting a prior distribution as a normal distribution , we can use a neuro network to generate and of and maximizing the KLD part of the ELBO by using a Kullback-Leibler divergence ; Maximizing the reconstruction loss will maximize the rest part of the ELBO.

As a result, the total loss of SpaPerb will be:

In SpaPerb the weight sets up as the following: .

Overall, the process works as the following:

# Datasets and preprocessing:

Mohammad et al. included three groups of control and stimulated cells: two groups of PBMC cells, and a group of HPOLY cells. Mohammad et al. preprocessed the data by removing megakaryocytic cells, filtering the cells with a minimum of 500 expressing cells, extracting the top 6998 cells, and log-transforming the original data. All the data are available on https://github.com/theislab/scgen-reproducibility.

In our model, we performed further data preprocessing to ensure consistency between control and stimulus cells within each cell type. Specifically, for each cell type, we randomly selected an equal number of cells from both the control and stimulated groups and used them to balance the dataset. This data preprocessing step helped us create a more robust and unbiased dataset, enabling accurate and fair comparisons between the control and stimulus conditions within each cell type during subsequent analyses. By doing such data processing, we guarantee that each pair of and have the same cell type, so the following style transfer process would be valid.

# Results and statistics

In SpaPerb, we evaluated the performance of our model using the square of the , which is calculated by the function. This metric measures the correlation between the generated images and the ground truth data. We computed the values for both the mean and variance of all genes, as well as for the top 100 Differentially Expressed Genes (DEGs).

To gain a visual understanding of the model’s results, we created scatter plots （注：就是左下角的图片，名字存疑） comparing the generated images to the corresponding ground truth data. This graph allowed us to observe how well the model’s predictions aligned with the actual values.

Additionally, we investigated the differences between the generated images and the ground truth data for the top DEG using a violin plot. The DEGs were identified using the function, employing the Wilcoxon method.

Through these analyses, we aimed to assess the accuracy and performance of our SpaPerb model in generating realistic images based on the input gene expression data. The evaluation of values and the visualization of the scatter and violin plots provided valuable insights into the model’s capabilities and highlighted any discrepancies between the generated and true data for further investigation.