bioRχiv

Deep learning applications in single-cell omics data analysis

Nafiseh Erfanian [1†], **A. Ali Heydari** [2,3†], **Pablo Iañez** [4], **Afshin Derakhshani** [5], **Mohammad Ghasemigol** [6], **Mohsen Farahpour** [7], **Saeed Nasseri** [8], **Hossein Safarpour** [8], and **Amirhossein Sahebkar** [9*]

[1] Student Research Committee, Birjand University of Medical Sciences, Birjand, Iran
[2] Department of Applied Mathematics, University of California, Merced, CA, USA
[3] Health Sciences Research Institute, University of California, Merced, CA, USA
[4] Cellular Systems Genomics Group, Josep Carreras Research Institute, Barcelona, Spain
[5] IRCCS Istituto Tumori "Giovanni Paolo II" of Bari, Bari, Italy
[6] Department of Computer Engineering, University of Birjand, Birjand, Iran
[7] Faculty of Electrical Engineering and Computer, University of Birjand, Birjand, Iran
[8] Cellular and Molecular Research Center, Birjand University of Medical Sciences, Birjand, Iran
[9] Biotechnology Research Center, Pharmaceutical Technology Institute, Mashhad University of Medical Sciences, Mashhad, Iran

[†] These authors contributed equally
[*] Correspondence should be addressed to A.S. (sahebkara@mums.ac.ir)

Abstract

Deep learning (DL) is a branch of machine learning (ML) capable of extracting high-level features from raw inputs in multiple stages. Compared to traditional ML, DL models have provided significant improvements across a range of domains and applications. Single-cell (SC) omics are often high-dimensional, sparse, and complex, making DL techniques ideal for analyzing and processing such data. We examine DL applications in a variety of single-cell omics (genomics, transcriptomics, proteomics, metabolomics and multi-omics integration) and address whether DL techniques will prove to be advantageous or if the SC omics domain poses unique challenges. Through a systematic literature review, we have found that DL has not yet revolutionized or addressed the most pressing challenges of the SC omics field. However, using DL models for single-cell omics has shown promising results (in many cases outperforming the previous state-of-the-art models) but lacking the needed biological interpretability in many cases. Although such developments have generally been gradual, recent advances reveal that DL methods can offer valuable resources in fast-tracking and advancing research in SC.
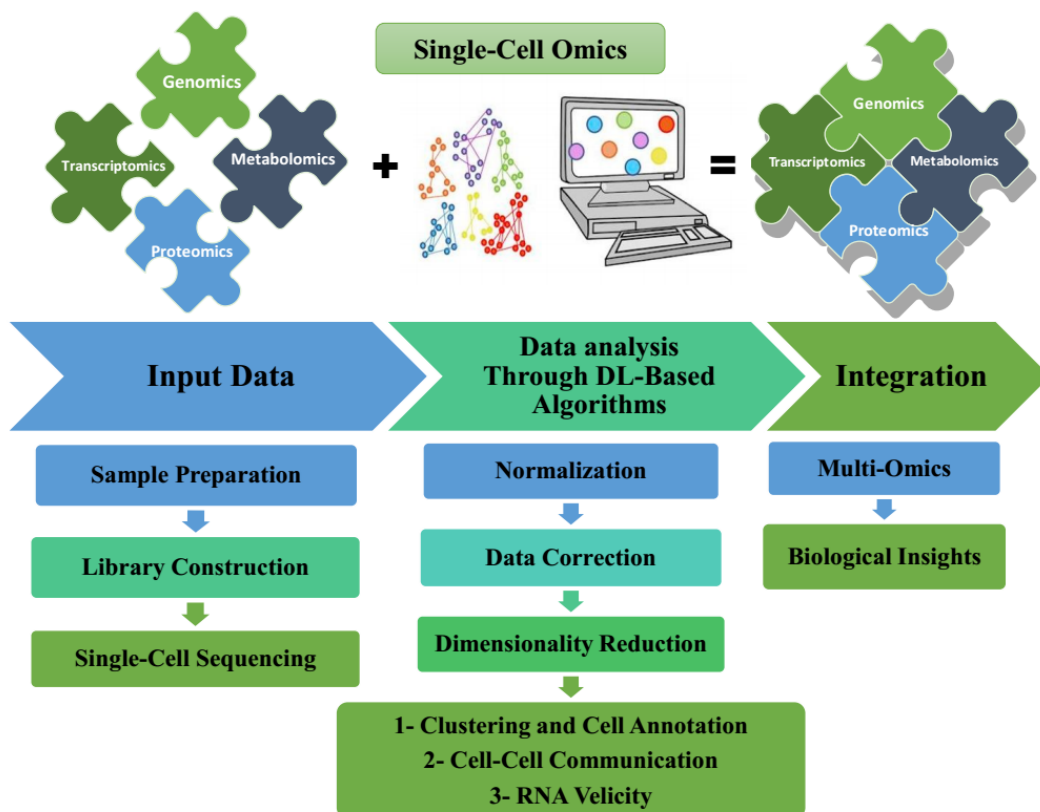
1. Introduction

Since it was highlighted as the "Method of the Year" in 2013 (Fritzsch, Dusny et al. 2012), sequencing individual cells at single-cell resolution has become the norm for studying cell-to-cell heterogeneity. RNA and DNA single-cell measurements, and recently epigenetic and protein levels, stratify cells at the highest possible resolution. That is, single-cell RNA sequencing (scRNA-seq) makes it possible to measure transcriptome-wide gene expression at the single-cell level. Such resolution enables researchers to distinguish different cell types based on their characteristics [see (Anchang, Hart et al. 2016, Haber, Biton et al. 2017, Tabula Muris, Overall et al. 2018, Han, Zhou et al. 2020)], organize cell populations, and identify cells transitioning between states. Such analyses provide a much better picture of tissues and the underlying dynamics in the development of an organism, which in turn allow for delineating intra-population heterogeneities that had previously been perceived as homogeneous by bulk RNA sequencing. Similarly, single-cell DNA sequencing (scDNA-seq) studies can reveal somatic clonal structures [e.g., in cancer, see (Roth, Khattra et al. 2014, Zafar, Navin et al. 2019)], thereby helping to monitor cell lineage development and providing insight into evolutionary mechanisms that function on somatic mutations.

The prospects of single-cell sequencing (SC-seq) are tremendous: it is now possible to re-evaluate hypotheses on differences between predefined sample groups at a single-cell level regardless of sample groups being disease subtypes, treatment groups, or morphologically distinct cell types. As a result, enthusiasm about the possibility of screening the basic units of life's genetic material has continued to expand in the recent years. Human Cell Atlas (Regev, Teichmann et al. 2017) is a prominent example: an effort to sequence the various cell types and cellular states that make up a human being. Encouraged by the great potential of single-cell investigation of DNA and RNA, there has been a substantial growth in the development of related experimental technologies. In particular, the advent of microfluidics techniques and combinatorial indexing strategies (Hosokawa, Nishikawa et al. 2017, Zilionis, Nainys et al. 2017) has resulted in routinely sequencing hundreds of thousands of cells in a single experiment. This growth has also allowed a recent publication to analyze millions of cells at once (Cao, Spielmann et al. 2019). More large-scale SC-seq datasets are becoming accessible worldwide (with tens of thousands of cells), constituting a data explosion in single-cell analysis platforms.

The continuous growth of scale and quantity of available SC-seq data has raised significant questions: 1) How do we correctly interpret and analyze the increasing complexity of SC-seq datasets? 2) How can different types of data sets (mentioned above) provide a deeper understanding of the underlying biological dynamics for a specific condition? and 3) how can the acquired information be changed into practical applications in medicine, ranging from rapid and precise diagnoses to accurate medicine and targeted preventions. As we face the challenges of rising chronic diseases, aging populations, and limited resources, a transformation towards intelligent analysis, interpretation, and understanding of complex data is essential. In this paradigm shift, the rapidly emerging field of ML is central (Zheng and Wang 2019).

ML is the study of algorithms that can automatically learn from data without the need for an explicit set of instructions. Simultaneously with biomedical advancements in the past decade, there has been a surge in the development and application of ML models, spearheaded by advances in DL. The recent improvements in computational hardware have made training DL models feasible, resulting in successful and revolutionary applications of such models across many fields. Some of the earliest DL algorithms developed were intended to be computational models of our brains' learning process, therefore being called "Artificial Neural Networks" (ANN). DL models often consist of many processing layers (with many nodes in each layer), which enable them to learn a representation of data with several levels of abstraction.

This review aims to discuss DL applications in SC-seq analysis and elaborate on their instrumental role in improving SC-seq data processing and analysis. We first introduce some key concepts in DL, which enable us to discuss their applications in the SC-seq field later in this manuscript. In Section 3, we review the existing literature on the applications of DL in SC-seq, particularly in transcriptomics, proteomics, metabolomics, and multi-omics integration. In the same section, we also provide an overview of techniques for handling the high technical noise and complexity of SC-seq data. Finally, we discuss possible future developments, highlighting both emerging obstacles and opportunities to advance the field.

2. Essential Concepts in Deep Learning

2.1 Feed Forward Neural Network

Learning algorithms use statistics and analytics to improve a given task's performance ("learn") utilizing only data, and without a specific set of instructions (Akay and Hess 2019). These algorithms use supervised and unsupervised learning tasks to match and identify inputs. Supervised learning, the most common form of ML (LeCun, Bengio et al. 2015), uses labeled data as inputs, which then is mapped to an output. An example of supervised learning in SC-seq is classifying cell subpopulations: this requires a labeled set of cell-types for training, an objective function for calculating learning statisitcs ("teaching" the model), and testing data for measuring how well the model can predict the cell-type (label) on data it has not seen before. Regression is another common supervised learning task where the model outputs continuous values instead of labels or categorical values. For all supervised tasks, the model is trained on a majority of the data (known as "training set") and then evaluated on held-out data ("test set"). Depending on the size of available data, there could also be a third data split known as a "validation set" used to measure the performance of the model throughout training. The evaluation of the model on the validation set is used for "Early Stopping", i.e. when the model is being overfitted on the training set and therefore generalizing worse if the learning continues (Prechelt 1998).
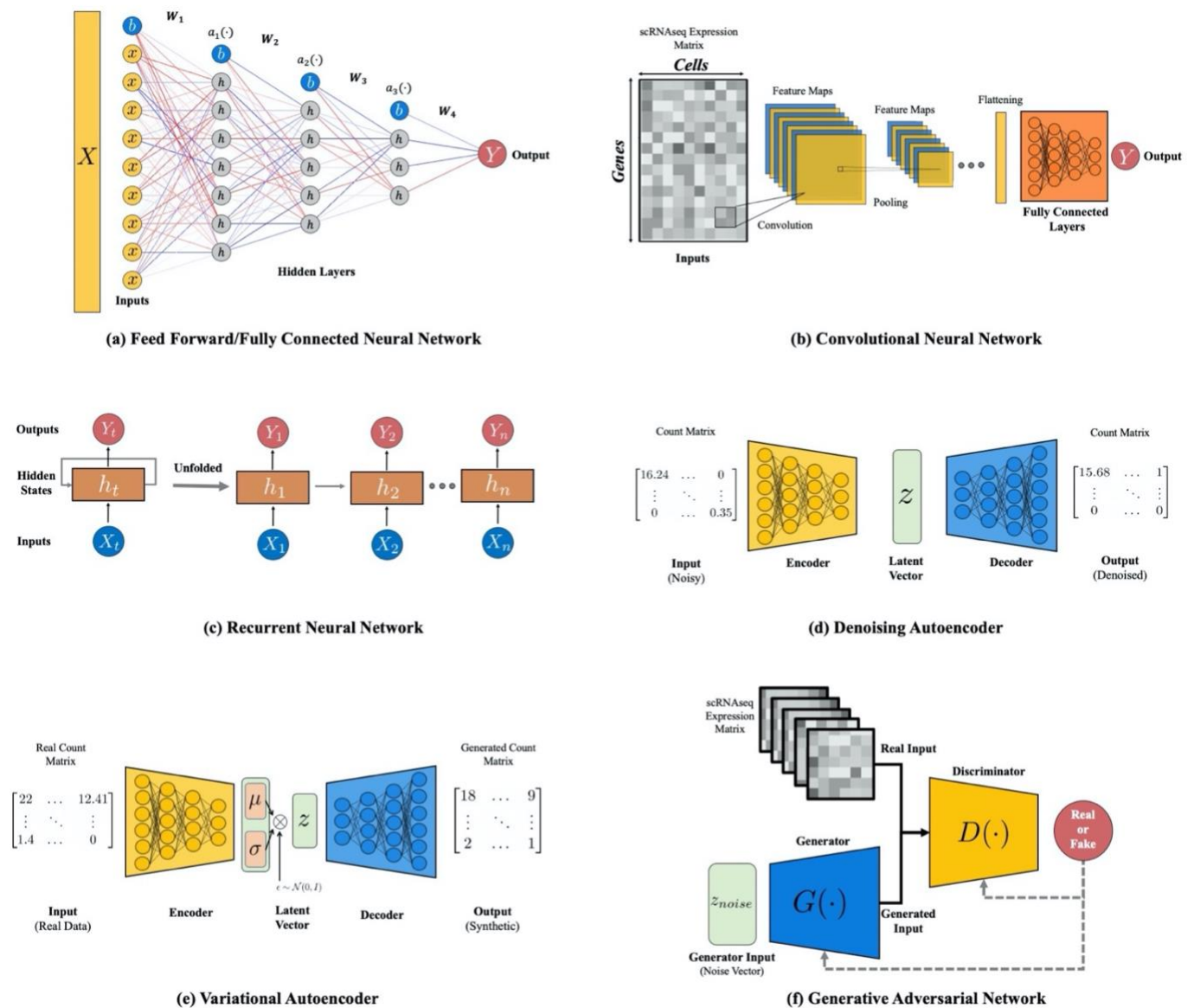
(a) Feed Forward/Fully Connected Neural Network

(b) Convolutional Neural Network

(c) Recurrent Neural Network

(d) Denoising Autoencoder

(e) Variational Autoencoder

(f) Generative Adversarial Network

Figure 1. (a) An example of a FFNN architecture. In this network, the raw input is shown as *X*, the weight tensors are denoted W_i, activation layers are α_i and the bias nodes are shown in blue nodes marked by *b*. The positive and negative weights are shown as blue and red edges, respectively, where a smaller numerical value corresponds to more transparent edges and vice versa. **(b)** An example of a CNN architecture for classification. In this hypothetical architecture, the model feeds the inputs through the three stages of a CNN (with non-linear activation not depicted) to extract various features. As is often the case, the output of the CNN is inputted to a fully connected neural network for classification. **(c)** An RNN depicted with its training flow, and an unrolled version showing the timestep-dependent inputs, hidden state, and outputs. **(d)** An illustration of a denoising autoencoder. Autoencoders (AE) have an encoder network, which tries to extract the most critical features and map them to a latent space (often of a much smaller dimension than the input data). The decoder network takes in the latent vector produced by the encoder (denoted as *z*) and maps this representation back to the original input dimension. **(e)** The architecture of a Variational AE (VAE) that aims to generate synthetic scRNA-seq data using the gene expression values. **(f)** A depiction of generative adversarial network for generating in-silico scRNAseq data. In this illustration, the scRNA-seq count matrix is converted to images that are used as real samples for the descriminator.

Due to the nature of SC-seq data, Feed Forward Neural Networks (FFNNs) are the most common architecture used as the core of many existing models for single-cell omics. FFNNs, the quintessential example of an ANN, are composed of interconnected nodes ("artificial neurons") that resemble and mimic the brain's neuronal functions.

The connections between the nodes ("edges") strengthen or weaken as the learning process progresses. Figure 1(a) illustrates an FFNN with an input layer (which senses and detects signals within the environment), three hidden layers (that process the signal sent by the input layer), and an output layer (the response to a signal or stimulus). FNNN can be viewed as a function that maps inputs from SC-seq space to an output space, often with fewer dimensions than the input space. This function is composed of simpler functions, each of which provides a different mathematical representation of the input. The true success of FFNN was realized when the capacity of such models was increased through the use of non-linear activation layers (such as Rectified Linear Units (Nair and Hinton 2010) and Hyperbolic Tangent) and adding more layers (depth).

These architectural choices allow neural networks to approximate highly non-linear and complex functions (theoretically being "universal approximators").

The of ANNs is to learn in the same manner as humans: by interacting with, and responding to, various stimuli within a local environment. FFNNs consist of numerous internal adjustable parameters (often on the order of hundreds of millions) that determine the input-output function of the machine. The objective of the model is to minimize a loss function that teaches the network to perform specific tasks. This optimization is done via the machine learning an optimal set of parameters (weights) that minimize the objective function. In FFNN, all neurons are fully connected; thus, the model must learn the optimal contribution of each node, which determines the final output of the model. That is, the direct connection between any two nodes can have positive or negative weights (refer to Fig. 1(a) for a visualization of this concept). The output values are sequentially computed within the network's hidden layers, with each input vector taking in the previous layer's output vector until reaching the final output layer (this process is known as the forward propagation). This multi-stage propagation of features through the layers results in an abstract data representation attentive to small details while ignoring irrelevant information (Hua, Zhao et al. 2019).

Initially assigned random values, the model weights are optimized throughout the training based on the loss function. Conventionally, the model optimization is done through gradient descent performed backward in the network, which consists of two components: a numerical algorithm for efficient computation of the chain rule for derivatives (backpropagation) and an optimizer (e.g., Stochastic Gradient Descent or Adam [Kingma and Ba 2015]). The optimizer is an algorithm which perform gradient descent, while backpropagation is an algorithm for computing the expression for gradients during the backward pass of the model. Deep neural networks (DNNs) are a subset of ANNs with more hidden layers between the input and output layers (usually greater than or equal to three hidden layers). This depth enables the model to learn complex nonlinear functions and allows the machine to learn a multi-step computer program (Goodfellow, Bengio et al. 2016). Several studies have shown that an increase in the network's capacity (e.g., through depth or non-linearity) has resulted in better performance in recognition and prediction tasks (Nair and Hinton 2010, Sun, Liang et al. 2015, Bahdanau, Chorowski et al. 2016).

Large-scale training of DNNs have been made possible by an increase in available data, algorithmic advances, and substantial increases in computational capacity, mainly through the use of general-purpose graphical processing units (GPUs) (Shi, Wang et al. 2016). At first, GPU hardware was so specialized that it could be used only for graphical tasks. However, once GPUs became more flexible for running custom subroutines, DL researchers were able to train simpler neural networks on these devices, which reduced the training time compared to CPUs (Steinkraus, Buck et al. 2005, Chellapilla, Puri et al. 2006). NVIDIA's C-like programming language for GPUs, called CUDA, provided researchers with even more flexibility in executing custom code. GPUs' convenient programming model combined with their high memory bandwidth and massive parallelism now stand as the ideal platform for developing deep neural networks (Goodfellow, Bengio et al. 2016).

Over the past seven years, DNNs have led to performance advancements in computer vision (Krizhevsky, Sutskever et al. 2012, Girshick, Donahue et al. 2014, Long, Shelhamer et al. 2015), speech identification (Hannun, Case et al. 2014), and natural language processing (Wu, Schuster et al. 2016), and the value of DL models in DNA sequence analysis was first demonstrated by a Alipanahi et al. in 2015 (Alipanahi, Delong et al. 2015, Zhou and Troyanskaya 2015). Since then, there has been a significant increase in the number of studies reporting the use of DNNs in genomics. Meanwhile, the DL community has substantially enhanced existing algorithms and has extended its repertoire of modeling methods, with some already impacting computational genomics.

2.2 Common DL Architectures

2.2.1 Recurrent Neural Network

RNNs (Rumelhart, Hinton et al. 1986) are often used for processing sequential data, including natural language and time series. RNNs process sequential inputs one at a time and implicitly maintain a history of previous elements of the input. We present a typical architecture for RNNs in Fig 1(c). The learning in RNNs occurs by propagating the gradients of each hidden state's inputs at discrete times. This process becomes more intuitive if we consider the outputs of the hidden units at various time iterations as if they were the outputs of different neurons in a deep multilayer network.

Due to the sequential nature of RNNs, the backpropagation of gradients would shrink or grow at each time step, causing the gradients to vanish or blow up for longer input sequences, makeing RNNs notoriously hard to train. However, when these issues are averted (via gradient clipping or other techniques), RNNs are powerful models and gain state-of-the-art capabilities in many applications across different fields (e.g. machine translation in natural language processing). The training challenges combined with the nature of SC-seq data have resulted in fewer developments of RNNs for single-cell analysis. However, recently some studies have used RNNs and Long Short-Term Memory (Hochreiter and Schmidhuber 1997), a variant of RNNs, used for predicting cell types and cell motility (Kimmel, Brack et al. 2019).

2.2.2 Convolutional Neural Network

CNNs (LeCun, Bengio et al. 1995) are specialized types of networks that use convolution (the mathematical operation) instead of tensor multiplication (which is done in FFNN) in at least one of their layers. This convolution operation makes CNNs ideal for processing data with a grid-like structure (images are a standard example of such datasets). CNNs have been effectively used in many applications in computer vision and time-series analysis. Still, they have not been used as

frequently in SC-seq data (due to not having a grid-like structure). However, some studies, such as Xu et al. (Xu, Zhang et al. 2020) have utilized CNNs after converting SC-seq data to images, which have shown promising results. CNNs take advantage of natural signals in three ways: (i) sparse interactions, (ii) shared weights, and (iii) equivariant representations (LeCun, Bengio et al. 1995).

The architecture of CNNs was inspired by the organization of the visual cortex (LGN–V1–V2–V4–IT hierarchy in the visual cortex ventral pathway [Hubel and Wiesel 1962]), with the connectivity pattern trying to resemble the neural connections of our brains. A typical CNN architectural block is composed of a sequence of layers (usually three) which include a convolutional layer (affine transform), a detector stage (non-linear transformation), and a pooling layer. The learning unit of a convolutional layer is called a filter or kernel. Each convolutional filter is a matrix, typically of small dimensions (e.g. 3x3), composed of a set of weights that acts as an object detector and is continuously calibrated during the learning process; that is, the CNN learns to find the filters it needs for the task at hand. A weighted sum of these local patches is inputted to a non-linear activation layer (second stage). The result of the convolution between the input data and the filter's weights is named a feature map. The output of a convolutional layer is made of as many stacked feature maps as filters are present within the layer. There are two key ideas behind this design: first, in data with grid-like structure (such as images), local neighbors have highly correlated information. Second, equivariance to translation can be obtained if units at different locations share weights. In other words, parameter sharing in CNNs allows for the detection of features regardless of the locations that they appear in. An example of this would be detecting a car. In a dataset, a car could appear at any position in a 2D image, but the network should be able to detect it regardless of the specific coordinates (Zhao, Zheng et al. 2019).

The resulting feature maps are then fed to a pooling layer. Here, a summary statistic of a rectangular neighborhood of nearby values is extracted from the detector stage's output at a certain location. There are many pooling operations, with the common ones being max-pooling (taking the maximum value of a rectangular neighborhood), mean-pooling (taking the average), and L_2 norm (taking the norm). In all cases, pooling takes input from rectangular patches from one or several feature maps and merges semantically similar features into one, decreasing the representation's dimension and making it insensitive to small shifts and distortions (LeCun, Bengio et al. 2015). CNNs typically have an ensemble of stacked convolution layers, non-linearity stage, and pooling layers, followed by completely connected layers that produce the final output of the network. Fig. 1(b) illustrates an example of a typical CNN architecture. The backpropagation of gradients through CNNs is analogous to DNNs, which enables learning the optimal weights for the filters in the model.

2.2.3 Autoencoders

AEs are neural networks that aim to reconstruct (or copy) the original input via a non-trivial mapping. Conventional AEs have an "hour-glass" architecture with two mirroring networks: an encoder and a decoder. The encoder's task is to map the input data to a latent space, often of a much smaller dimension than the original input space. The encoder is responsible for data compression and feature extraction, forming the narrowing part of the hourglass architecture (see Fig. 1(d)). The output of the encoder network (latent vector) will contain the most important features present in the data in a compressed form. Conversely, the decoder is tasked with mapping the latent vector back to the original input dimension and reconstructing the original data. In the ideal case, the decoder's output will be an exact copy of the training sample.

AEs were traditionally used for dimensionality reduction and denoising, aiming to minimize a mean squared error (MSE) objective among the input data and the reconstructed sample (output of the decoder). Fig. 1(d) depicts an example of a denoising AE. Over time, the AE framework has been generalized to stochastic mappings of encoder and decoder distributions. A well-known example of such generalization is the Variational AEs (Kingma and Ba 2015), where using the same hour-glass architecture, one can generate new samples that are similar to the training data. Both traditional AEs and the generalized have practical applications in many biological fields.

2.2.4 Deep Belief Network

Although Deep Belief Networks (DBNs) (Hinton et al. 2006) have fallen out of favor in recent years, they had an essential role in starting the DL revolution. DBNs are generative models with several layers of binary latent variables, which use probabilities and unsupervised learning to produce outputs. Unlike other models where each layer extracts a specific feature of its input, individual layers of DBNs learn the input in its entirety. DBNs are composed of stacked undirected energy-based models (known as Restricted Boltzmann Machines) with two layers of visible and hidden units, respectively. The nodes in each layer of DBNs are fully connected to the subsequent and previous layers, with a network of symmetrical weights connecting each layer. The connection of these nodes helps in extracting features and identifying the correlation in the input data. The training of the model is done layer-by-layer, with each layer ensuring a layer-based optimal choice (greedy learning algorithms). To reach a global optimum, the input data is fed to the first layer, and each subsequent layer takes the output of the predecessor layer as its input.

Despite DBNs being generative models, they were primarily used as a pre-training core that could be fine-tuned as an ANN for other tasks. This provided one of the first examples of transfer learning, where the unsupervised learned weights would be fine-tuned on a specific supervised task, with classification being the primary application. In addition to improving classification, DBNs also showed promising results in image and video recognition in the early days.

2.2.5 Variational Autoencoders

VAEs (Kingma and Ba 2015, Kingma and Welling 2019) are generative networks that learn deep latent-variable and inference models simultaneously, i.e., they are made up of generative and inference models. VAEs are AEs that employ variational inference to recreate the original data, allowing them to produce new (or random) data that is "similar" to that which already exists in a dataset (illustrated in Fig. 1(e)). Traditional VAEs have stronger mathematical properties and training stability than Generative Adversarial Networks (GANs), and thus have been used extensively in SC omics application. However, VAEs could suffer from two major weaknesses: conventional VAEs create "blurry" samples (those that adhere to an average of the data points), rather than the sharp samples that GANs generate due to adversarial training. Introspective VAEs (IntroVAEs) Huang et al. have generally solved this issue by specifying an adversarial training between the encoder and the decoder (Huang, Li et al. 2018). IntroVAEs are single-stream generative algorithms that assess the quality of the images they generate. IntroVAEs have largely been employed in computer vision, where they have outperformed their GAN counterparts in applications like synthetic image generation (Huang, Li et al. 2018) and single-image super-resolution (Heydari and Mehmood 2020).

The other major issue with VAEs occurs when the variational posterior and actual posterior are nearly identical to the prior (or collapse to the prior). Because the generative model excludes a subset of latent variables which may have important latent features for inputs, this occurrence frequently creates data creation concerns (He, Spokoyny et al. 2019). Some studies have suggested that this happens because of the regularization term in the VAE's objective purpose (Lucas, Tucker et al. 2019), i.e. when the prior and posterior divergence is close to zero. Studies aimed at reducing posterior collapse can be divided into two categories: (i) solutions aimed at weakening the generative model (Semeniuta, Severyn et al. 2017, Yang, Hu et al. 2017), (ii) alterations to the training purpose (Tolstikhin, Bousquet et al. 2017, Yang, Hu et al. 2017, Zhao, Song et al. 2017), and (iii) alterations to the training procedure (He, Spokoyny et al. 2019, Heydari et al. 2019). If both issues mentioned above can be addressed, VAEs have shown to perform comparably (or similar) to GANs while training faster due to the simpler training procedure, which can be tremendously advantageous for SC omics applications.

2.2.6 Generative Adversarial Networks

GANs (Goodfellow, Pouget-Abadie et al. 2014) can generate realistic synthetic data and have been effectively utilized in a variety of computer vision tasks (Dziugaite, Roy et al. 2015, Vondrick, Pirsiavash et al. 2016, Zhu, Krähenbühl et al. 2016), natural language processing (Yang, Chen et al. 2017, Fedus, Goodfellow et al. 2018), time series synthesis (Esteban, Hyland et al. 2017, Engel, Agrawal et al. 2019), and bioinformatics (Marouf, Machart et al. 2020). GANs are made up of a generator network (G) and a discriminator network (D) that compete in a zero-sum game; we present the architecture of GANs in Fig. 1(f). The goal of the G network is to produce fake samples that resembles the distribution of the real data, to "fool" the D network into believing that these fake samples are real. Conversely, D trains to learn the difference between real and synthetic samples (hence the name *discriminator*) . In each GAN training iteration, the entire system is re-adjusted to update both G and D parameters. In this process, the generator learns to generate more realistic samples as well as deceiving the discriminator with higher success. At the same time, the discriminator learns to develop its classification ability over the enhanced samples from the generator. GANs' ability to produce realistic samples is attributed to the adversarial training between G and D. GANs have several advantages to other generative models, including the flexibility to generate any sort of probability density, no prior assumptions for training the generator network, and no limitations on the size of the latent space.

Despite these benefits, GANs are notoriously difficult to train since achieving Nash equilibrium for G and D is extremely difficult (Wang, She et al. 2021). Another drawback of GANs is vanishing gradients, which occur when an optimal D is insufficient for G to learn and progress. As demonstrated by Arjovsky et al., if D learns the distinction between real and generated data well enough, G will be unable to train (Arjovsky, Chintala et al. 2017). Another problem with GANs is "mode collapse," which occurs when G produces only a small number of outputs that potentially trick D. This happens when G has trained to map many noise vectors z to the same output that D recognizes as real data. In this case, G is over-optimized, resulting in a lack of diversity in the generated samples. Quantifying how much GANs have learned the distribution of real data is often difficult, hence one of the most common methods of assessing GANs is to evaluate the output directly (Larsen, Sønderby et al. 2016), which could be laborious.

Even though certain GAN variations have been proposed to reduce vanishing gradients and mode collapse, (e.g., Wasserstein-GANs (WGANs) [Arjovsky, Chintala et al. 2017] and Unrolled-GANs [Metz, Poole et al. 2016]), the convergence of GANs is still a big issue. In the case of mode collapse, the feedback from D to G becomes pointless as the training progresses. If GANs are trained past this stage, the quality of synthetic samples may be affected, and the system may eventually collapse. It is also worth noting that most GANs can not be trained as single-stream networks, thus defining a training schedule for G and D separately is required, which adds an additional layer of intricacy. Although all DL models are susceptible to hyperparameter settings, Lucic et al. (Lucic, Kurach et al. 2017) show that GANs (including WGANs) are substantially more sensitive than VAEs. Such sensitivity could be a disadvantage of utilizing GANs for scRNA-seq generation since the hyperparameters may require to be re-tuned for each new dataset.
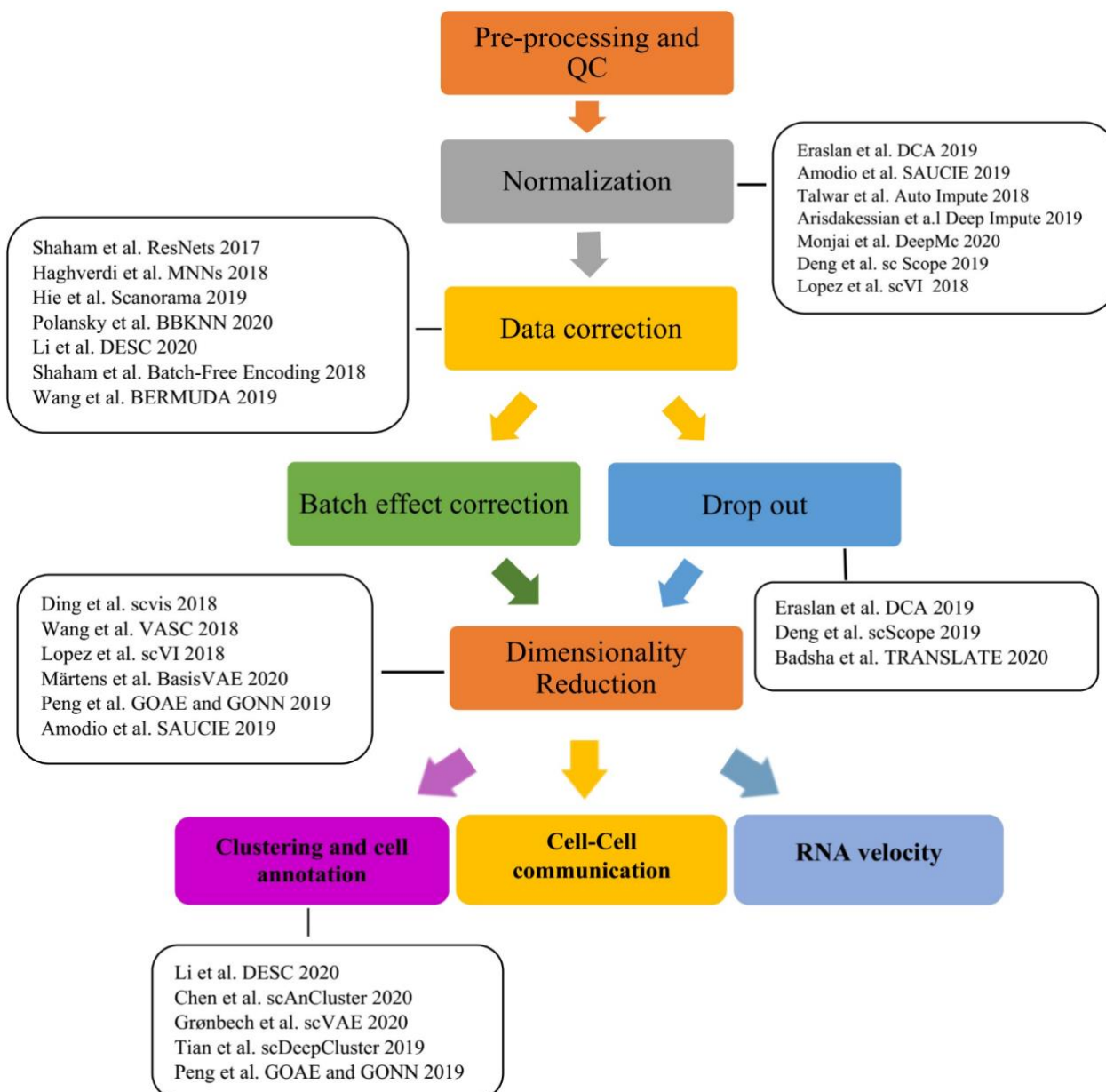
3. DL Applications in SC Omics

Figure 2. Workflow for RNA-seq data analysis

3.1 DL in SC Transcriptomics

ScRNA-seq has improved our understanding of biological processes substantially in recent years. In addition to studying cellular heterogeneity in humans and mice (Tabula Muris, Overall et al. 2018, Han, Zhou et al. 2020), we can now analyze the transcriptome of other organisms such as zebrafish, frogs, and planaria (Briggs, Weinreb et al. 2018, Plass, Solana et al. 2018, Wagner, Weinreb et al. 2018), as well as uncover previously unknown cell populations (Montoro, Haber et al. 2018, Plass, Solana et al. 2018). Given the vast potentials of SC transcriptomics, computational biologists have developed a variety of research methods (Rostom, Svensson et al. 2017) (Fig. 2).

3.1.1 Pre-processing and quality control

Based on tag-based unique molecular identifiers (UMIs), raw data provided through sequencing machines is utilized to generate matrices of molecular counts (count matrices) or read counts (read matrices). Reading quality control (QC), mapping reads to their cellular barcodes and mRNA molecules of sources (also known as "demultiplexing"), genome alignment, and quantification is all handled by raw data processing pipelines, including Cell Ranger (Zheng, Terry et al. 2017), indrops (Klein, Mazutis et al. 2015), SEQC (Azizi, Carr et al. 2018), or zUMIs (Parekh, Ziegenhain et al. 2018). The dimension of the read (or count matrix) generated is the number of barcodes times the number of transcripts. Since all reads allocated to the same barcode may not correspond to reads from the same cell, the word "barcode" is used instead of

"cell." A barcode can tag several cells by mistake (doublet) or not tag any cells at all (empty droplet/well). Although the rate of noise in measurements differs between read and count data, the processing methods in a standard research pipeline are all the same (Lafzi, Moutinho et al. 2018).

Despite scRNA-seq data's richness which can offer significant and deeper insights, the data's complexity and noise are far higher than in traditional bulk RNA-seq, making it challenging to prepare raw sequence data for further study. Unwanted variations such as biases, artifacts, etc., require extensive quality control (QC) and normalization efforts (Jiang, Thomson et al. 2016). The number of counts per barcode (count depth), the number of genes per barcode, and the fraction of counts from mitochondrial genes per barcode are three QC covariates widely used in cell QC (Griffiths, Scialdone et al. 2018). These variants may result in low-quality libraries in scRNA-seq data due to various factors, including cell damage during dissociation or library preparation failure. As a result, low-quality libraries are a concern because they can lead to erroneous findings in downstream analyses. Therefore, the cells at the beginning of the study should be excluded to prevent - or at least minimize – such issues. Currently, there is an unmet need for developing more efficient and accurate methods for filtering low-quality cells.

Given the limited number of studies regarding pre-processing and quality control, in the following, we focus only on the DL applications that are most related to normalization, data correction, and downstream analysis.

3.1.2 Normalization

Normalization is a crucial first step in pre-processing scRNA-seq expression data to address the constraints caused by low input content, or the different types of systematic measuring biases (Bacher, Chu et al. 2017). Normalization aims to detect and remove changes in measurements between samples and features (e.g., genes) caused by technical artifacts or unintended biological effects (e.g., batch effects) (Hogan, Courtier et al. 2019). Methods designed for normalizing bulk RNA-seq and microarray data are often used to normalize scRNA-seq data. However, these techniques often ignore essential aspects of scRNA-seq results (Hogan, Courtier et al. 2019). For scRNA-seq data, a few families of normalization techniques have been developed, such as scaling techniques (Lun, Bach et al. 2016), regression-based techniques for identified nuisance factors (Buettner, Natarajan et al. 2015, Bacher, Chu et al. 2017), and techniques based on spike-in sequences from the External RNA Controls Consortium (ERCC) (Ding, Zheng et al. 2015, Vallejos, Marioni et al. 2015). While these techniques address some issues caused by utilizing bulk normalization methods for scRNA-seq, they all have limitations in their applicability across different research designs and experimental protocols. Accounting for technical noise of scRNA-seq data remains a challenge and an active area of research within the field (Zheng and Wang 2019). Recently, DL-based methods were used to alleviate some of these challenges.

3.1.3 Data correction

Although normalization aims to address the noise and bias in the data, normalized data can still contain unexpected variability. These additional technical and biological variables, such as batch, dropout, cell cycle effects are accounted for during the "data correction" stage, which depends on the downstream analysis (Luecken and Theis 2019). In addition, it is a recommended practice to address biological and technical covariates separately (Luecken and Theis 2019), since they serve different purposes. Given the mentioned subtleties, designing DL models that can address most of these challenges is difficult. Therefore, there are currently no DL models that are widely used for data correction within the field.

3.1.3.1 Dropout

Compared to bulk RNA-seq, scRNA-seq datasets are noisy and sparse and pose unique nuance such as "dropout" which is one of the most significant issues in this field (Kharchenko, Silberstein et al. 2014, Gong, Kwak et al. 2018). A dropout is when a gene is observed at a low to a moderate level in one cell but not in another cell of the same type (Qiu 2020). Dropout events can occur during library preparation (e.g. extremely low levels of mRNA in single cells) or due to biological properties (the stochastic aspect of gene expression in multiple cells)(Ran, Zhang et al. 2019). Shorter genes have lower counts and a greater dropout rate (Zappia, Phipson et al. 2017). Overall, a low RNA capture rate results in the inability of detecting an expressed gene, leading to a "false" zero, known as a dropout event. Furthermore, it has been suggested that sometimes near-zero expression measurements can also be dropouts (Lin, Troup et al. 2017). Dropout events will introduce technical variability and noise, adding an extra layer of difficulty in analyzing scRNA-seq (Sengupta, Rayan et al. 2016), and downstream analyses, such as clustering and pseudo-time reconstruction, can be affected (Arisdakessian, Poirion et al. 2019).

It is essential to understand the difference between "false" and "true" zero counts. True zero counts mean that a gene is not expressed in a particular cell type, indicating true cell-type-specific expression (Eraslan, Simon et al. 2019). Hence, it is important to note that zeros in scRNA-seq data do not necessarily translate to missing values and must remain in the data. However, the false zeros (missing values) must be imputed to further improve the analysis. The missing values are replaced with either random values or by an imputation method (Eraslan, Simon et al. 2019). Imputation approaches designed for bulk RNA-seq data may not be suitable for scRNA-seq data for multiple reasons, mainly due to scRNA-seq's heterogeneity and dropouts. ScRNA-seq has much higher cell-level heterogeneity than bulk RNA-seq data; scRNA-seq has cell-level gene expression data while bulk RNA-seq data represents the averaged gene expression of the cell population. Dropouts in scRNA-seq are not necessarily missing values; they have zero expression and are combined with real zeros. Additionally, the number of missing values in bulk RNA-seq data is much lower compared to scRNA-seq (Gong, Kwak et al. 2018). Given these factors and the non-trivial difference between true and false zero counts, classic imputation approaches with specified

missing values are often not be appropriate for scRNA-seq data, and scRNA-seq-specific dropout imputation methods are required.

Current scRNA-seq imputation methods can be divided into two groups: those that change all gene expression levels like high non-zero values, such as MAGIC (van Dijk, Nainys et al. 2017) and SAVER (Huang, Wang et al. 2018), and those that impute drop-out events (zero or near-zero counts) alone, such as scImpute (Li and Li 2018), DrImpute (Gong, Kwak et al. 2018), and LSImpute (Moussa and Măndoiu 2019). These techniques can fail to account for the non-linearity of the data's count structure. Moreover, as larger scRNA-seq datasets become available and common, imputation methods should scale to millions of cells, however, many of the earlier models are either incapable of or very slow at processing datasets of this size (Eraslan, Simon et al. 2019). As a result, many have resorted to designing DL-based approaches to combat these challenges, both on the technical and efficiency fronts.

Most DL algorithms for imputing drop-out events are based on AEs. For example, in 2018, Talwar et al. proposed AutoImpute, a technique for retrieving the whole gene expression matrix using overcomplete AEs to impute the dropouts. AutoImpute learns the underlying distribution of the input scRNA-seq data and imputes missing values dependent on that distribution, with minor modifications to the biologically silent gene expression values. Through expanding the expression profiles into a high-dimensional latent space, AutoImpute learns the underlying distribution and patterns of gene expression in single cells and reconstructs an imputed model of the expression matrix. In that year, they claim that their system is the only one that can conduct imputation on the largest of the nine datasets in question, PBMC (which contains ~68,000 cells), without running out of memory (Talwar, Mongia et al. 2018).

In another study, Eraslan et al. proposed a deep count AE network (DCA). DCA uses a negative binomial noise model both with and without zero-inflation to account for the count distribution, overdispersion, and sparsity of the results, and nonlinear gene-gene dependencies are captured. Since their approach scales linearly with the number of cells, it can be used on millions of cell datasets. It also depends on gene similarities; using simulated and true datasets, DCA denoising enhances several traditional scRNA-seq data analyses. One of the key benefits of DCA is that it only requires the user to define the noise model. Current scRNA-seq approaches depend on a variety of hypotheses, including zero-inflated negative binomial models. DCA increases biological exploration by outperforming current data imputation approaches in terms of quality and time. Overall, DCA calculates the "dropout probability" of a zero-expression value due to scRNA-seq dropout and imputes the zeros only when the probability is high. Consequently, while DCA effectively detects true zeros, it can be biased when dealing with nonzero values (Eraslan, Simon et al. 2019).

Badsha et al. propose TRANSLATE (TRANSfer learning with LATE) (Badsha, Li et al. 2020), a DL model for computing zeros in scRNA-seq datasets which are extremely sparse. Their nonparametric approach is based on AEs and builds on their previous method, LATE (Learning with AuToEncoder). The key presumption in LATE and TRANSLATE is that all zeros in the scRNA-seq data are missing values. In most cases, their approach achieves lower mean squared error, restores nonlinear gene-gene interactions, and allows for improved cell type separation. Both models are also very scalable, and on a GPU, they can train on over a million cells in a few hours. TRANSLATE has shown better performance on inferring technical zeros than other techniques, while DCA is better at inferring biological zeros than TRANSLATE.

SAUCIE (Amodio, Van Dijk et al. 2019) is a regularized AE that denoises and imputes data using the reconstructed signal from the AE. Despite the noise in the input data, SAUCIE can restore the significant relationships across genes, leading to better expression profiles which can improve downstream analyses such as differential gene expression (Amodio, Van Dijk et al. 2019).

ScScope (Deng, Bao et al. 2019) is a recurrent AE network that iteratively handles imputation by employing a recurrent network layer; taking the time recurrence of ScScope to one (i.e. *T=1)* will reduce the model to a traditional AE. Given that ScScope is a modification of AEs, its runtime is similar to other AE-based models (Deng, Bao et al. 2019).

There also have been a few non-AE-based models developed for imputation and denoising of scRNA-seq data. DeepImpute (Arisdakessian, Poirion et al. 2019) uses several sub-neural networks to impute groups of target genes using signals (genes) that are strongly associated with the target genes. Arisdakessian et al. demonstrate that DeepImpute has a better performance than DCA, contributing the advantages to their divide-and-conquer approach (Arisdakessian, Poirion et al. 2019).

Mongia et al. (Mongia, Sengupta et al. 2020) introduced deepMc, an imputation method based on deep matrix factorization for missing values in scRNA-seq data that utilizes a feed backward neural network. In most of their experiments, deepMc outperformed other existing imputation methods while not requiring any assumption on the prior distribution for the gene expression. We predict that deepMc will be the preferred initial approach for imputing scRNA-seq data, given the superior performance and simplicity of the model.

scVI is another DNN algorithm introduced by Lopez et al (Lopez, Regier et al. 2018). scVI is based on a hierarchical Bayesian model and uses DNN to define the conditional probabilities , assuming either a negative binomial or a zero-inflated negative binomial distribution (Lopez, Regier et al. 2018). Lopez et al. show that scVI can accurately recover gene expression signals and impute the zero-valued entries, potentially enhancing the downstream analyses without adding any artifacts or false signals.

Recently, Patruno et al. (Patruno, Maspero et al. 2020) compared 19 denoising and imputation methods, based on numerous experimental scenarios like recovery of true expression profiles, characterization of cell similarity, identification of differentially expressed genes, and computation time. Their results showed that ENHANCE, MAGIC, SAVER, and SAVER-X offer the best overall results when considering efficiency, accuracy and robustness for the measured tasks (Patruno,

Maspero et al. 2020). However, the more traditional methods are not well-suited for large-scale scRNA-seq, and therefore making DL-based models as favorable candidates instead. Given this fact, more work is required to build upon the previous DL methods for imputing dropout effects and better managing biological zeros while retaining technical zero precision.

3.1.3.2 Batch Effect Correction

When samples are conducted in separate batches, the term 'batch effect" is used to describe the variation caused by technical effects. Different types of sequencing machines or experimental platforms, laboratory environments, different sample sources, and even the technician who performed the experiment can cause batch effects (Fei and Yu 2020). Removing and accounting for batch effects is often helpful and it is often recommended, however, the success varies significantly across different datasets. For example, batch effect removal on bulk RNA-seq data from ENCODE (Encyclopedia of DNA Elements) human and mouse tissues (Lin, Lin et al. 2014) is a recommended standard data preparation step. Batch effect correction has been an active area of research since the microarray time. Johnson et al. suggested parametric and non-parametric empirical Bayes frameworks for adjusting the data for batch effects removal (Johnson, Li et al. 2007). In the recent years and with an increase level of complexity in sequencing datasets, more involved batch effect correction methods have been proposed and used (Fei and Yu 2020). However, a majority of the existing approaches need biological group expertise for each observation and were originally designed for bulk or microarray RNA-seq data. Given the heterogeneity present within scRNAseq-date, these earlier techniques are not well suited for SC analysis in certain cases (Luo and Wei 2019). Alternative batch effect correction techniques for scRNA-seq data have been developed to address the specific needs of single-cell datasets. Batch effects in scRNA-seq data may have a substantial impact on downstream data analysis, impacting the accuracy of biological measurements and ultimately contributing to erroneous conclusions (Büttner, Miao et al. 2019).

Several statistical methods, including linear regression models like ComBat (Johnson, Li et al. 2007), and nonlinear models like Seurat's canonical correlation analysis (CCA) (Zhang, Wu et al. 2019) or scBatch (Fei and Yu 2020), have been designed to eliminate or minimize scRNA-seq batch effects while aiming to maintain biological heterogeneity of scRNAseq data. Additionally, some differential testing frameworks such as limma (Ritchie, Phipson et al. 2015), MAST (Finak, McDavid et al. 2015), and DESeq2 (Love, Huber et al. 2014) already integrate the batch effect as a covariate in model design.

Due to the stochasticity of gene expression, failures in RNA capture or amplification across sequencing, SC studies are prone to dropout events, as discussed previously. Consequently, many studies aim at improving the workflows of dealing with dataset that exhibit dropout events (Tran, Ang et al. 2020), resulting in a new family of batch effect correction using DL methods.

Recently, there has been a considerable progress in using DL for batch effect corrections. Residual Neural Networks (ResNets) and AEs are two of the most commonly used DL-based batch correction approaches in scRNA-seq analysis. Shaham et al. (Shaham, Stanton et al. 2017) suggested a non-linear batch effect correction approach based on distribution-matching ResNet. ResNets are a form of deep neural network that make a direct connection between the input of a layer (or network) and the outputs, often through an addition operation. Their approach focuses on reducing the Maximum Mean Discrepancy (MMD) between two multivariate replication distributions that were measured in separate batches. Shaham et al. applied their methodology to batch correction of scRNA-seq and mass cytometry datasets, finding that their model can overcome batch effects without altering the biological properties of each sample (Shaham, Stanton et al. 2017).

Haghverdi et al. (Haghverdi, Lun et al. 2018) developed a novel and efficient batch correction method for SC data analysis that detects cell mappings between datasets, and subsequently reconstructs the data in a shared space. To create relations between two datasets, the algorithm first recognizes mutual nearest neighbors (MNNs). The translation vector is computed from the resulting list of paired cells (or MNNs) to align the data sets into a shared space. The benefit of this method is that it produces a normalized gene expression matrix, which can be used in downstream analysis and offer an effective correction in the face of compositional variations between batches (Haghverdi, Lun et al. 2018). However, MNN is computationally demanding in terms of CPU time and memory since it computes the list of neighbors in a high-dimensional gene expression space, (Fig. 3).

Scanorama (Hie, Bryson et al. 2019) and BBKNN (Polański, Young et al. 2020) are two other approaches that look for MNNs in reduced-dimension spaces, and use them in a similarity-weighted way to direct batch integration. Hie et al. (Hie, Bryson et al. 2019) proposed Scanorama which can combine and eradicate batch effects from heterogeneous scRNA-seq studies by recognizing and merging common cell types across all pairs in a dataset. Using a variety of existing tools, Scanorama batch-corrected output can be used for downstream tasks, such as classify cluster-specific marker genes in differential expression analysis. Scanorama outperforms current methods for integrating heterogeneous datasets and it scales to millions of cells, eventually allowing the identification of rare or new cell states through a variety of diseases and biological processes (Hie, Bryson et al. 2019).

Polańsky et al. (Polański, Young et al. 2020) developed BBKNN (batch balanced *k*-nearest neighbors), a fast graph-based algorithm that removes batch effects through linking analogous cells in different batches. BBKNN is a simple, rapid, and lightweight batch alignment tool, and its output can be directly applied for dimensionality reduction. BBKNN's default approximate neighbor mode scales linearly with the size of datasets and remains consistently faster by one or two orders of magnitude when compared to other existing techniques (Polański, Young et al. 2020).
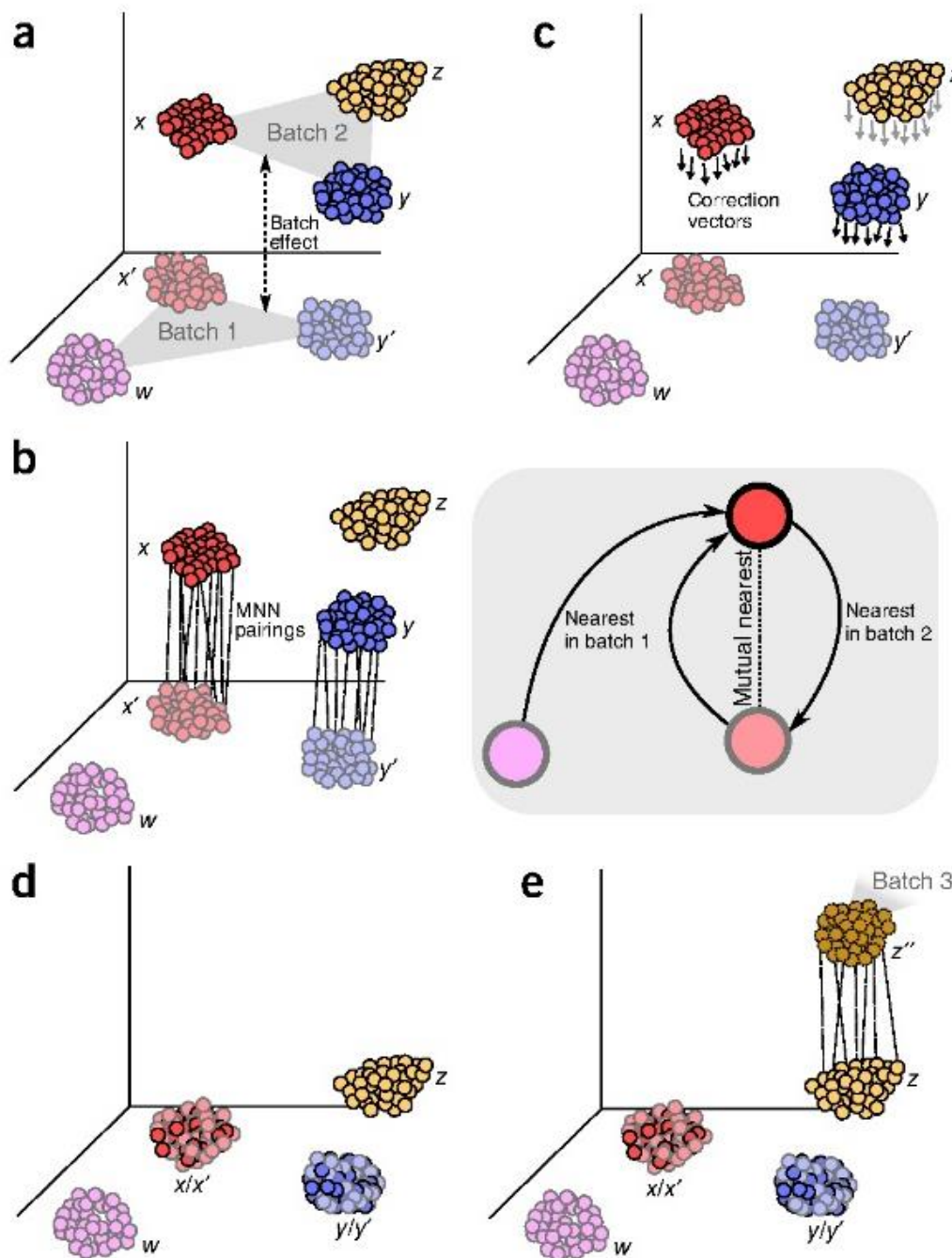
Figure 3: Batch-effect correction via MNN. **(a)** Batch 1 and batch 2 in high dimensions, with a batch effect variation that is almost orthogonal. **(b)** By identifying MNN pairs of cells, the algorithm recognizes matching cell types (gray box). **(c)** Between the MNN pairs, batch-correction vectors are measured. (d) Batch 1 is considered the reference, and batch 2 is combined into it by subtracting correction vectors. **(e)** The integrated data is used as a reference, and the process is repeated with each new batch of data. This figure has been reused with permission from authors (Haghverdi, Lun et al. 2018).

Moving onto the DL-based methods for batch correction, Li et al. (Liu and Di He 2019) presented DESC (deep embedding algorithm for single-cell clustering), an unsupervised DL algorithm for "soft" SC clustering which can also remove batch effects. DESC learns a non-linear mapping function from the initial scRNA-seq data space to a low-dimensional feature space using a DNN and iteratively improving a clustering objective function. This sequential process transfers each cell to the cluster closest to it and attemps to account for biological and technical variability across the clusters. Li et al. demonstrated that DESC can eliminated the technical batch effect more accurately than CCA and MNN while better preserving true biological differences within closely related immune cells (Li, Wang et al. 2020).

In a prior study, Shaham (Shaham 2018) proposes batch effect correction through batch-free encoding using an adversarial VAEs. Shaham utilizes an adversarial training to achieve data encoding that corresponded exclusively to a subject's intrinsic biological state, as well as to enforce accurate reconstruction of the input data. This approach results in maintaining the true biological patterns expressed in the data and minimizing the significant biological information loss (Shaham 2018).

Wang et al. introduced BERMUDA (Batch Effect ReMoval Using Deep Autoencoders) (Wang, Johnson et al. 2019), an unsupervised framework for correcting batch effect in scRNA-seq data across different batches. BERMUDA combines separate batches of scRNA-seq data with completely different cell population compositions and amplifies biological signals by passing information between batches. While MNN and BBKNN can manage variations in cell population composition between batches when such differences are significant, their efficiency is less optimal to BERMUDA. On the other hand, BERMUDA was developed with an emphasis on scRNA-seq data with distinct cell populations in mind, and it is focused on the similarities between cell clusters, even though scRNA-seq data can be constantly variable, such as data produced for cell differentiation (Amodio, Van Dijk et al. 2019). Altogether, a rapidly expanding number of general DL methods for batch effects correction in biological datasets represent new ways for eliminating batch effects in biological datasets.

3.1.3.3 Dimensionality reduction

A typical scRNA-seq data table consists of ~20,000 genes as columns/features and 50,000 to up to a million cells as rows/observations. Therefore some form of dimensionality reduction is necessary to visualize and interpret such high-dimensional data (Wang and Gu 2018). The most common dimensionality reduction techniques used for scRNA-seq are PCA (Sun, Zhu et al. 2019), t-SNE (Van der Maaten and Hinton 2008), diffusion map (Haghverdi, Buettner et al. 2015), GPLVM (Titsias and Lawrence 2010, Buettner and Theis 2012), SIMLR (Wang, Ramazzotti et al. 2017), and UMAP (Becht, McInnes et al. 2019), to name a few.

In low-dimensional spaces, linear projection methods like PCA traditionally cannot depict the complex structures of single-cell data. On the other hand, nonlinear dimension reduction techniques like t-SNE and UMAP, have been shown to be effective in a variety of applications and are commonly used in single-cell data processing (Ding, Condon et al. 2018). These methods also have some drawbacks, such as lacking robustness to random sampling, inability to capture global structures while concentrating on local data structures, parameter sensitivity, and high computational cost (Zheng and Wang 2019). Several DL techniques for reducing the dimensionality of scRNA-seq data have recently been developed. Here we focus on the ones that are based on VAEs or AEs.

Ding et al. (Ding, Condon et al. 2018) proposed a VAE-based model to learn a parametric transformation from a high-dimensional space to a low-dimensional embedding, which is used to learn the estimated posterior distributions of low-dimensional latent variables. Compared to common techniques (e.g. t-SNE), their approach (scvis) can obtain the global structure of the data, is more interpretable, and is more robust to cell type detection in the presence of noise or unclear measurements. Ding et al. demonstrated that scvis is a promising tool for regular study of large-scale, high-resolution cell population mapping (Ding, Condon et al. 2018). However, according to Becht et al. (Becht, McInnes et al. 2019), the runtime of scvis is high, particularly for dimensionality reduction, and it appears to be less effective at separating cell populations. In another work, Wang et al. (Wang and Gu 2018) proposed a method for unsupervised dimensionality reduction and visualization of scRNA-seq data named VAE for scRNA-seq data (VASC), which uses a deep VAE. VASC's architecture consists of the traditional encoder and decoder network of a VAE, with an addition of a zero-inflated layer that simulates dropout events. In comparison to current methods such as PCA, t-SNE, and ZIFA, VASC can identify nonlinear patterns present within the data, and has broader compatibility as well as better accuracy, particularly when sample sizes are larger (Wang and Gu 2018).

In 2020, Märtens et al. proposed BasisVAE (Märtens and Yau 2020) as a general-purpose approach for joint dimensionality reduction and clustering of features using a VAE. BasisVAE modified the traditional VAE decoder to incorporate a hierarchical Bayesian clustering prior and demonstrated how collapsed variational inference can identify sparse solutions when over-specifying K. Furthermore, translation invariance lets us handle scenarios where not all features are aligned (Märtens and Yau 2020).

Peng et al. (Amodio, Van Dijk et al. 2019) proposed an AE-based model that combines gene ontology (GO) and deep neural networks to achieve a low-dimensional representation of scRNA-seq data. Based on this idea, they proposed two innovative approaches for dimensionality reduction and clustering: an unsupervised technique called "GOAE" (Gene Ontology AutoEncoder) and a supervised technique called "GONN" (Gene Ontology Neural Network) for training their AE model and extracting the latent layer as low dimensional representation. Their findings show that by integrating prior information from GO, neural network clustering and interpretability can be enhanced and that they outperform the state-of-the-art dimensionality reduction approaches for scRNA-seq (Peng, Wang et al. 2019).

In a study by Armaki et al. (ARMACKI 2018), the dimensionality reduction capabilities of VAE- and AE-based models were evaluated and benchmarked against principal component analysis. They found that the best approach for reducing the dimensionality of single-cell data was the AE-based model, while the more efficient VAE performed worse in some respects than the linear PCA. One possible hypothesis could be that the prior used for modeling the latent space, which was Gaussian distribution, is not a good fit for single-cell data. A prior more befitting single-cell data (such as negative binomial distribution) could improve the performance of the VAE based model (ARMACKI 2018).

Finally, there is always the endeavor of optimizing algorithms. As mentioned earlier, representing an efficient DL method for dimensional reduction of data is a necessary next step, since it can potentially improve the quality of lower-dimensional representations.

3.1.3.4 In-Silico Generation and Augmentation

Given limitations on scRNA-seq data availability and the importance of adequate sample sizes, in-silico data generation and augmentation offer a fast, reliable, and cheap solution. Synthetic data augmentation is a standard practice in various

ML areas, such as text and image classification (Shorten and Khoshgoftaar 2019). With the advent of DL, traditional data augmentation techniques (such as geometric transformations or noise injection) are being replaced with more deep-learned generative models, primarily VAEs (Kingma and Welling 2013) and GANs (Goodfellow, Pouget-Abadie et al. 2014). In computational genomics, both GANs and VAEs have shown promising results in generating omics data. Here, we focus on the recent methods introduced for generating realistic in-silico scRNA-seq.

Marouf et al. (Marouf, Machart et al. 2020) introduced two GAN-based models for scRNA-seq generation and augmentation called single-cell GAN scGAN and conditional scGAN (cscGAN); we collectively refer to these models as scGAN. At the time, scGAN outperformed all other state-of-the-art methods for generating and augmenting scRNA-seq data (Marouf, Machart et al. 2020). The success of scGAN is attributed to the Wasserstein-GAN (Arjovsky, Chintala et al. 2017) that learns the underlying manifold of scRNA-seq data, which then can produce realistic never-seen-before samples. Marouf et al. showcase the power of scGAN by generating specific cell types that are almost indistinguishable from the real data, and augmenting the dataset with the synthetic samples improved the classification of rare cell populations.

In a related work, Heydari et al. (Heydari, Davalos et al. 2021) introduce a VAE-based in-silico scRNA-seq model that aimed at improving Marouf et al.'s training time, stability, and generation quality using only one framework (as opposed to two separate models). Heydari et al. propose ACTIVA (Automated Cell-Type-informed Introspective Variational Autoencoder), which employs a single-stream adversarial VAE conditioned with cell-type information. The cell-type conditioning encourages ACTIVA to learn the distribution of all cell types in the dataset, even if they may be rare, which allows the model to generate specific cell types on demand. Heydari et al. showed that ACTIVA performs better or comparably to scGAN while training up to 17 times faster due to the design choices. Data generation and augmentation with both ACTIVA and scGAN can enhance scRNA-seq pipelines and analysis, such as benchmarking new algorithms, studying the accuracy of classifiers, and detecting marker genes. Both generative models will facilitate the analysis of smaller datasets, potentially reducing the number of patients and animals necessary in initial studies (Heydari, Davalos et al. 2021).

3.1.4 Downstream Analysis

Following pre-processing, downstream analysis methods are used to derive biological understandings and identify the underlying biological mechanism. For example, cell-type clusters are made up of cells with similar gene expression profiles; minor differences in gene expression between similar cells indicate continuous (differentiation) trajectories; or genes which expression profiles are correlated, signaling co-regulation (Zhang, Cui et al. 2021).

3.1.4.1 Clustering and cell annotation

A significant phase in the scRNA-seq study is to classify cell subpopulations and cluster them into biologically relevant entities (Yang, Liu et al. 2017). The creation of several atlas projects such as Mouse Cell Atlas (Han, Wang et al. 2018), Aging Drosophila Brain Atlas (Davie, Janssens et al. 2018), and Human Cell Atlas (Rozenblatt-Rosen, Stubbington et al. 2017) has been initiated by advances in single-cell clustering. Different clustering approaches have emerged in recent years to help characterize different cell populations within scRNA-seq data (Zheng, Li et al. 2019). Given that DL-based algorithms outperform traditional ML models in clustering tasks when applied to image and text datasets (Guo, Zhu et al. 2018), many have turned to designing supervised and unsupervised DL-based clustering techniques for scRNA-seq.

Li et al. presented DESC (Deep Embedding algorithm for Single-Cell Clustering), a AE-based method for clustering scRNA-seq data with a self-training target distribution that can also denoise and potentially remove batch effects. In experiments conducted by Li et al. (Li, Wang et al. 2020)., DESC attained a high clustering accuracy across the tested datasets compared to several existing methods, showed consistent performance in a variety of scenarios, and did not directly need the batch definition for batch effect correction. Given that Li et al. use a deep AE to reconstruct the input data, the latent space is not regularized with additional properties that could additionally help with clustering (Chen, Wang et al. 2020). As with most DL models, DESC can be trained on CPUs or GPUs.

Chen et al. proposed scAnCluster (Single-Cell Annotation and Clustering), an end-to-end supervised clustering and cell annotation framework which is built upon their previous unsupervised clustering work, namely scDMFK and scziDesk. ScDMFK algorithm (Single-Cell Data Clustering through Multinomial Modeling and Fuzzy K-Means Algorithm) combined deep AEs with statistical modeling. It proposed an adaptive fuzzy k-means algorithm to handle soft clustering while using multinomial distribution to describe the data structure and relying on neural network support to facilitate model parameter estimation (Chen, Wang et al. 2020). More specifically, the AE learns a low-dimensional representation of the data, and an adaptive fuzzy k-means algorithm with entropy regularization is used in that space to perform soft clustering. On the other hand, ScziDesk aimed at learning a "cluster-friendly" lower dimensional representation of the data, since many existing deep learning-based clustering algorithms for scRNA-seq do not consider distance and affinity constraints cells from the same populations. For scAnCluster, Chen et al. use the available cell marker information to construct a new DL model that integrates single-cell clustering and annotation. ScAnCluster can do both intra-dataset and inter-dataset cell clustering and annotation, and it also reveals a clear discriminatory effect in the detection of new cell types not found in the reference results (Tran, Ang et al. 2020).

Tian et al. created the scDeepCluster (Single-Cell model-based Deep embedded Clustering) technique, which uses a nonlinear approach to combine DCA modeling and the DESC clustering algorithm. Their approach sought to improve clustering while reducing dimensions directly. ScDeepCluster outperformed state-of-the-art approaches on a variety of clustering efficiency metrics, showing increased scalability with runtime increasing linearly with sample size. In contrast to similar methods, scDeepCluster requires less memory and and is scalable to large (Tian, Wan et al. 2019). On the other hand,

ScDeepCluster lacks the pairwise distance of associated cells and ignores the affinity limit of similar cells. ScDeepCluster does not pre-select such informative genes as input data, which not only excludes clustering precision but also improves memory and processing time (Chen, Wang et al. 2020). Peng et al. (Peng, Wang et al. 2019) developed a strategy for optimizing cell clustering based on global transcriptome profiles of genes. They used a combination of DNN and Gene Ontology (GO) to reduce the dimensions of scRNA-seq data and improve clustering. Their supervised approach is based on a conventional neural network model, and the unsupervised approach is based on an AE model, respectively. Their model consists primarily of two main components: the choosing of important GO terms and the combination of GO terms with the DNN-based model (Peng, Wang et al. 2019). Finally, Grønbech et al. introduced a VAE-based model, called scVAE (Single-Cell Variational Auto-Encoders) for clustering cells. Since scVAE uses the raw count data as input, many of the traditional pre-processing steps are not required. ScVAE can accurately predict expected gene expression levels and a latent representation for each cell and is flexible to use the known scRNA-seq count distributions (such as Poisson or Negative Binomial) as its model assumption (Grønbech, Vording et al. 2020).

3.1.4.2 Cell-Cell communication analysis

In recent years, scRNA-seq has become a powerful method for analyzing cell-cell communication in tissues. Intercellular communication controlled by ligand-receptor complexes is essential for coordinating a wide range of biological processes, including development, differentiation, and inflammation. Several algorithms have been proposed to carry out these analyses. They all begin with a database of interacting molecular partners (such as ligand and receptor pairs) and predict a list of possible signaling pathways between types of cells based on their expression patterns. Although the findings of these studies may provide useful insight into the mechanisms of diverse tissues made up of a variety of cell types, they can be difficult to visualize and analyze using current algorithms (Tian, Wan et al. 2019, Almet, Cang et al. 2021). Algorithms based on DL have not yet been fully developed, however, given the importance of this issue, it is expected that new DL methods will be developed for analyzing cell-cell communication.

3.1.4.3 RNA Velocity

In scRNA-seq data, RNA velocity has created new ways to research cellular differentiation. It represents the rate of change in gene expression for a single gene at a given time point depending on the spliced ratio to unspliced mRNA. In other words, RNA velocity is a vector that forecasts the possible future of individual cells on a timescale of hours, and it has created many new insights into cellular differentiation that have challenged conventional and long-standing methods. Existing experimental models for tracking cell fate and reconstructing cell lineages, such as Genetic methods or time-lapse imaging have limited power and do not show the trajectory of differentiation or reveal the molecular identity of intermediate states. But recently, with the advent of scRNA-seq, a variety of algorithms such as VeloViz (Atta and Fan 2021) or scVelo (Bergen, Lange et al. 2020) by analyzing data are generated to visualize the velocity estimates in low dimensions. Despite these developments, particularly in well-characterized systems, we are far from a complete understanding of cellular differentiation and cell fate decisions. By this means, we foresee the advent of a more integrative model, such as DL-based algorithms, to resolve long-standing concerns regarding cell fate choices and lineage specification in this way.

3.2 DL in SC Genomics

Traditional sequencing methods are limited to measuring the average signal in a group of cells, potentially masking heterogeneity and rare populations (Tang, Huang et al. 2019). ScRNA-seq technologies provide a tremendous advantage for investigating cellular heterogeneity and recognizing new molecular features correlated with clinical outcomes, consequently resulting in the transformation of many biological research fields. SC genomics is being used in many areas, such as predicting the sequence specificity of DNA- and RNA-binding proteins, and enhancer and cis-regulatory regions, methylation status, gene expression, and control splicing, and searching associations between genotype and phenotype. However, SC genomics data are often too large and complex to be analyzed only through visual investigation of pairwise correlations. As a result, a growing number of studies have leveraged DL techniques to process and analyze these large datasets. In addition to the scalability of DL algorithms, another advantage of DL techniques is the learned representations from raw input data, which are beneficial in specific SC genomics and epigenomics applications, such as cell-type dentification, DNA methylation, chromatin accessibility, TF-gene relationship prediction, and histone modifications. In the following sections, we review some applications of DL models for analyzing scDNA-seq data (Fig. 4).

3.2.1 Cell type identification in CyTOF

An essential and challenging task in genomics research is to accurately identify and cluster individual cells into distinct groups of cell types. Li et al. (Li, Shaham et al. 2017) describe AE methods (stacked AE and multi-AE) as a gating strategy for mass cytometry (CyTOF). CyTOF is a recent technology for high-dimensional multiparameter SC analysis. They introduced DeepCyTOF as a standardization procedure focused on a multi-AE neural network. This model allows cells from only one sample to be labeled. This is focused on domain adaptation principles and is a generalization of previous work that helps users in calibrating between a source domain distribution (reference sample) and several target domain distributions (target samples) in a supervised manner. DeepCyTOF was applied to two CyTOF datasets produced from primary immune blood cells: (a) cases with a history of West Nile virus (WNV) infection and (b) normal cases of various ages. They manually gated a single baseline reference sample in each of these datasets to automatically gate the remaining uncalibrated samples.
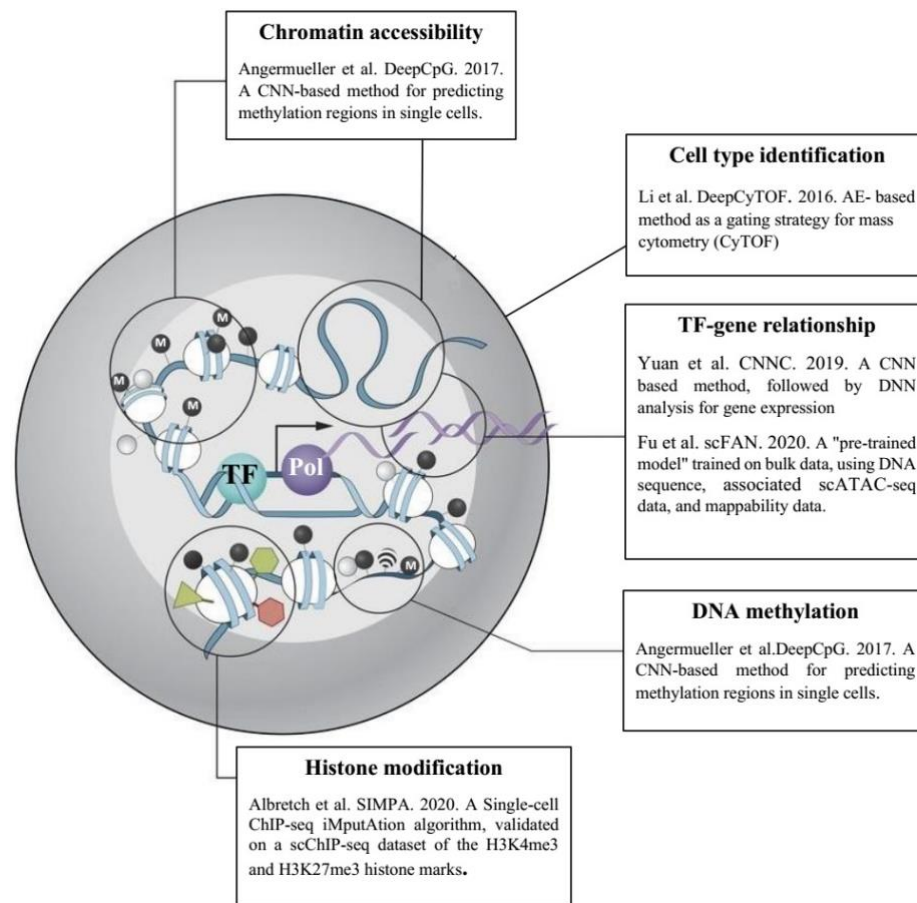
Figure 4. Overview of algorithms that are used in different parts of single-cell genomics analysis.

They revealed that DeepCyTOF cell classification is extremely consistent with cell classification gained by individual manual gating of each sample, with over 99% concordance. Moreover, they used a stacked AE, which is one of DeepCyTOF's key components building blocks, to tackle the semi-automated gating challenge of the FlowCAP-I competition. Li et al. found that their model outperformed other existing gating approaches benchmarked on the fourth challenge of the competition. Overall, stacked AEs combined with a domain adaptation technique suggest promising results for CyTOF semi-automated gating and flow cytometry data, requiring manual gating of one reference sample to precisely gate the remaining samples.

3.2.2 DNA methylation

Recent technological advances have made it possible to assay DNA methylation at SC resolution. Angermueller et al. propose DeepCpG (Angermueller, Lee et al. 2017), a CNN-based computational method for predicting methylation regions, modeling methylation data from single cells with low coverage. DeepCpG is made up of three modules: a DNA module that extracts features from the DNA sequence, a CpG module that extracts features from the CpG neighborhood of all cells, and a multi-task joint module that integrates evidence from both modules for predicting the methylation regions of target CpG sites for different cells. The trained DeepCpG model can be used in various downstream studies, such as inferring low-coverage methylation profiles for groups of cells and recognizing DNA sequence motifs linked to methylation states and cell-to-cell heterogeneity. Angermueller et al. apply their model to both mouse and human cells, which achieves significantly more precise predictions than previous techniques. (Angermueller, Lee et al. 2017). An overview of the DeepCpG model is shown in (Fig. 5).

Interestingly, DeepCpG can be used for differentiating human induced pluripotent stem cells in parallel with transcriptome sequencing to specify splicing variation (exon skipping) and its determinants. Linker et al. (Linker, Urban et al. 2019) presented that variation in SC splicing can be precisely predicted based on local sequence composition and genomic features. DeepCpG, which is used for DNA methylation profiles, imputes unobserved methylation regions of individual CpG sites. The cell-type-specific models were made using CpG and genomic information according to DeepCpG's setup of a joint model. Finally, during cell differentiation, Linker et al. recognized and characterized associations between DNA methylation and splicing changes, which led to indicating novel insights into alternative splicing at the SC level.
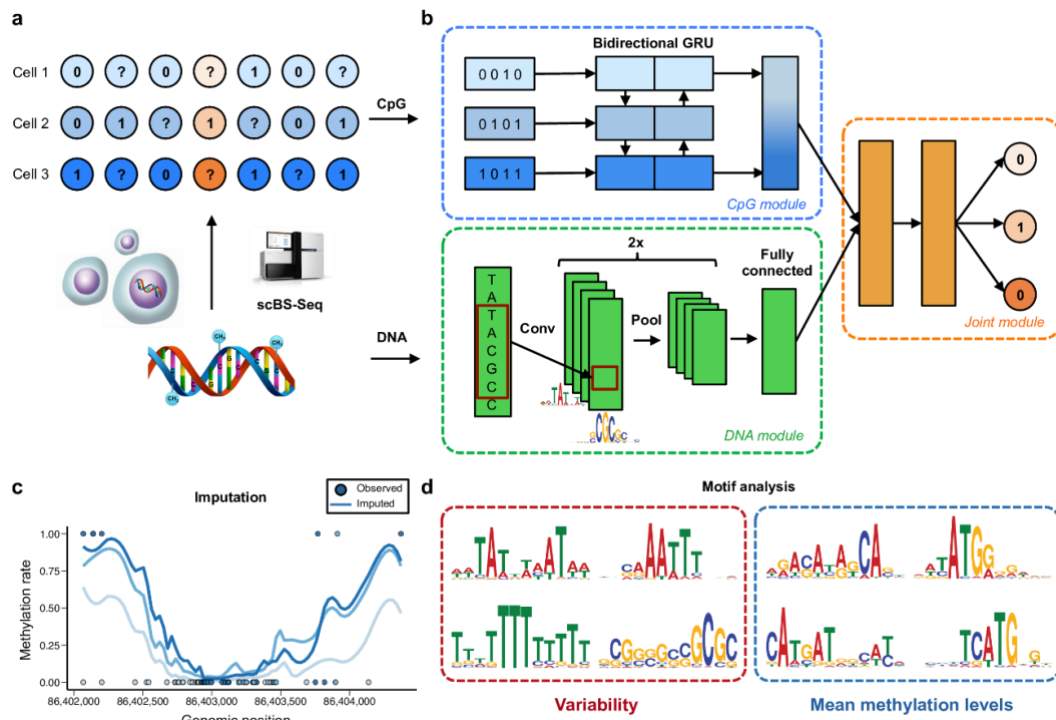
Figure 5. An overview of DeepCpG (taken from Angermueller et al.) **(a)** scBS-seq (Smallwood et al. 2014) scRRBS-seq (Farlik et al. 2015) provide sparse single-cell CpG profiles, which are then pre-processed as binary values (methylated CpG sites are marked by ones, unmethylated CpG sites are denoted by zeros, and CpG sites with unknown methylation state [missing data] are represented by question marks) **(b)** Two convolutional and pooling layers are used to detect predictive motifs from the local sequence context, and one fully connected layer is used to model motif interactions in the DNA model. Applying a bidirectional gated recurrent network (GRU), the CpG model scans the CpG neighborhood of numerous cells (rows in b), providing compressed information in a vector of constant size. To forecast methylation states in all cells, the Joint model learns connections between higher-level properties generated from the DNA- and CpG models (c, d) The DeepCpG model can be utilized for a variety of downstream tasks after training, such as genome-wide imputation of missing CpG sites **(c)** and the finding of DNA sequence motifs linked to DNA methylation levels or cell-to-cell variability **(d)**.

Another method to study chromatin accessibility at single-cell resolution, called Assay of Transposase Accessible Chromatin sequencing (scATAC-seq), has gained considerable popularity in the last few years. In scATAC-seq, mutation-induced hyperactive Tn5 transposase tags and fragments regions in open chromatin sites in the DNA sequence, which is later sequenced using paired-end NGS technologies (Yan, Powell et al. 2020). The pre-processing steps of scATAC-seq data analysis are often analogous to scRNA-seq pipelines; the same tools are often used in both data modalities, although scRNA-seq tools are often not optimized for the particular properties of scATAC-seq data. ScATAC-seq has low coverage, and the data analysis is highly sensitive to non-biological confounding factors. The data is pre-processed and assembled into a feature-per-cell matrix, where common choices for "feature" are fixed-size genomic bins and signal peaks at biological events. This matrix displays particular numerical properties which entail computational challenges: it is extremely high-dimensional, sparse, and near-binary in nature (presence/absence of signal). Several packages have been recently developed specifically for scATAC-seq data, with all of them having some major limitations (Cao, Fu et al. 2021). The tools for processing scATAC-seq are diverse and nescient, and thus, there is no consensus on the best practices scATAC-seq data analysis. Further development of scATAC-centric computational tools and benchmark studies are much needed. ScATAC-seq analyses can help to elucidate cell types and differentially accessible regions across cells. Moreover, it can decipher regulatory networks of cis-acting elements like promoters and enhancers, and trans-acting elements like transcription factors (TFs), and infer gene activity (Baek and Lee 2020). ScATAC-seq data could also be integrated with RNA-seq and other omics data, however, most current software only integrate the derived gene activity matrix with expression data, and important information from whole-genome chromatin accessibility is lost. We delve deeper into this topic in the "*DL in the integration of SC multimodal omics data*" section of this manuscript.

Currently, there are a variety of DL models for bulk ATAC-seq data, such as LanceOtron's CNN for peak calling (Hentges, Sergeant et al. 2021) and CoRE-ATAC for functional classification of cis-regulatory elements (Thibodeau, Khetan et al. 2020). Thibodeau et al. demonstrated CoRE-ATAC's transferable capability on cell clusters inferred from single nuclei ATAC-seq data with a rather small decrease in model prediction accuracy (mean micro-average precision of 0.80 from bulk vs. 0.69 in single-cell clusters).

One common way to reduce the dimensionality of scRNA-data is to identify the most variable genes (e.g. with PCA), since they carry the most biologically relevant information. However, scATAC-seq data is binary, and therefore not allowing for identification of variable peaks for dimension reduction. Instead, dimensionality reduction of scATAC data is done through Latent Semantic Indexing (LSI), a technique used for natural language processing. Although this approach is scalable to large number of cells and features, it may fail to capture the complex reliance of peaks since it is a linear method. SCALE (Single-Cell ATAC-seq analysis via Latent feature Extraction) (Xiong, Xu et al. 2019) combines a deep generative framework with a

probabilistic Gaussian Mixture Model (GMM) as a prior over the latent variables, in order to learn a nonlinear latent space of scATAC-seq features.

Given the nature of scATAC-seq, GMM is suitable distribution for modeling the high-dimensional, sparse multimodal scATAC-seq data. SCALE can also be used for denoising and imputing missing data, recovering signals from missing peaks. In the benchmarking study by Xiong et al., they demonstrated that SCALE outperformed conventional non-DL scATAC-seq tools for dimensionality reduction, such as PCA and LSI. Furthermore, they show its scalability to large datasets in the order of 80000 single cells. While SCALE succeeds at learning nonlinear cell representations with higher accuracy, it assumes that the read depth is constant across cells and ignores potential batch effects. These pitfalls motivated the development of SAILER (scalable and accurate invariant representation learning scheme) (Cao, Fu et al. 2021). SAILER is a deep generative model inspired by VAEs that also learns a low-dimensional latent representation of each cell. For SAILER, the authors aimed at designing an invariant representation learning scheme, where they discard the learned component associated with confounding factors from various technical sources. SAILER captures nonlinear dependencies among peaks, faithfully separating biologically relevant information from technical noise, in a manner that is easily scalable to millions of cells (when using GPUs). Similarly, SCALE also offers a unified strategy for scATAC-seq denoising, clustering, and imputation. However, in multi-sample scATAC-seq integration, SAILER can eliminate batch effects and properly recreate a chromatin accessibility landscape free of confusing variables, regardless of sequencing depths or batch effects.

3.2.3 TF-gene relationship prediction

To unravel gene regulatory mechanisms and differentiate heterogeneous cells, understanding genome-wide binding transcription factor (TF) profile is crucial. There have been several approaches that aim to use expression data to understand gene-gene interactions, such as mutual information and correlation techniques for coexpression analysis, undirected graphical and clustering models for functional assignments, and directional graph-based models for pathway reconstruction (Yuan and Bar-Joseph 2019). In the space of DL-based methods, Yuan et al. (Yuan and Bar-Joseph 2019) presented CNNC (Convolutional Neural Network for Co-Expression), a new encoding method for gene expression data based on CNN, followed by DNN analysis. CNNC is a general computational technique for supervised gene relationship inference that builds on previous approaches in various tasks, including predicting transcription factor targets and recognizing genes related to disease in order to infer cause and effect. The key idea behind CNNC is to turn data into a co-occurrence histogram, making it optimal for CNNs. That is, Yuan et al. produced a histogram (image) for each pair of genes and applies convolutional neural networks (CNNs) to infer relationships among the various levels of expression encoded in the image. CNNC is adaptable and can easily be expanded to integrate other types of genomics data, resulting in additional performance gains. CNNC goes beyond previous approaches for predicting TF-gene and protein-protein interactions and predicting the pathway of a regulator-target gene pair. It may also be used to draw causality inferences, functional assignments (such as biological processes and diseases), and as part of algorithms that recreate known pathways (Yuan and Bar-Joseph 2019).

In a recent study, Fu et al. (Fu, Zhang et al. 2020) introduced scFAN (Single Cell Factor Analysis Network), a DL model for determining genome-wide TF binding profiles in individual cells. The scFAN pipeline consists of a "pre-trained model" trained on bulk data and then used to predict TF binding at the cellular level using DNA sequence, aggregated associated scATAC-seq data, and mappability data. ScFAN can help in overcoming the basic sparsity and noise constraints of scATAC-seq data. This model provides a valuable method for predicting TF profiles through individual cells and could be applied to analyze sc epigenomics and determine cell types. Fu et al. presented scFAN's ability to identify cell types by analyzing sequence motifs enriched within predicted binding peaks and studying the effectiveness of predicted TF peaks. They suggested a novel metric called "TF activity score" to classify each cell and demonstrated that the activity scores could accurately capture cell identity. Generally, scFAN is capable of connecting open chromatin states with transcript factor binding activity in individual cells, which is beneficial for a deeper understanding of regulatory and cellular dynamics (Fu, Zhang et al. 2020).

3.2.4 Histone modification

Because of the effects of protein-DNA interactions between histone marks and TF on the regulation of crucial cellular processes (including the organization of chromatin structures and gene expression), the identification of such interactions is highly significant in biomedical science. ChIP-seq (chromatin immunoprecipitation followed by sequencing) is a widely used technique for mapping transcription factors, histone changes, and other protein-DNA interactions for genome-wide mapping (Furey 2012). ChIP-seq data is suffering from sparsity hence Albretch et al. introduced SIMPA (Albrecht, Andreani et al. 2021), a Single-cell ChIP-seq imputation algorithm that was tested on a scChIP-seq dataset of the H3K4me3 and H3K27me3 histone marks in B-cells and T-cells. Unlike most SC imputation approaches, SIMPA integrates the sparse input of one single cell with a series of 2,251 ENCODE ChIP-seq experiments to extract predictive information from bulk ChIP-seq data. Overall, SIMPA aims to identify statistical patterns that bind protein-DNA interacting sites through specific SC regions of target-specific ENCODE data for different cell types, as well as the presence or absence of potential sites for a single cell. Once these patterns are identified, SIMPA's DL models will use these patterns to make precise predictions. As a new approach in SC-seq, SIMPA's imputation strategy was able to augment sparse scChIP-seq data from single cells, leading to improved cell-type clustering and the detection of pathways that are specific to each cell type.

In conclusion, SC genomics analysis through DL is a promising and increasingly developing field with incredible potential to advance our knowledge of fundamental biological issues. In this respect, DL will enable us to gain a better understanding

of the nature and intricacy of DNA structure and epigenomics effects on human disease for both therapeutic and diagnostic goals. Due to intrinsic challenges, such as the sparsity and complexity of the data, systematic noise, and the features of biological systems, it is required to advance in the development of new DL methods in the SC genomics field.

3.3 DL in SC Proteomics

Proteomics is the large-scale study of proteomes (all proteins produced or modified by an organism or system). Single-cell Proteomics concentrates on the classification, localization, and functional analysis of the cell protein and suggests pictures of cellular activities and phenotypic characteristics. SC proteomics has been shown as an effective method for phenotypic profiling of individual cell types and their physiological structure, quantitative protein evaluation, and the recognition of post-translational disease-related changes which cannot be predicted through genomics or transcriptomics studies.

SC proteomics can be applied to nearly every biological area, such as creating a proteome map of each cell type in an organism or studying disease progression, diagnosis, and treatment prognosis. Moreover, SC proteomics can be used for linage finding of cellular phenotypes, evaluating the activity of particular immune cells in normal, scarce, and abundant conditions, cancer diagnosis and immunotherapy. It can also be used to understand the effect of targeted inhibitors on malignancy cells, cell response to engineered molecular stimulations, signaling pathways, cell motility research, cell-cell separation distance, analysis of cellular dynamics, personalized medicine, finding of new drug targets, high-throughput drug screening, etcetera. (Minakshi, Kumar et al. 2019). The main limiting factor in SC proteomics is the insufficient quantity of protein in single cells, contributing to high noise background in protein analyses. Unlike DNAs and RNAs, protein cannot be amplified directly, which is crucial in minimizing noise in the measured signals. Certain functional proteins, like proteins secreted for signaling and intracellular phosphoproteins, could be as small as tens of copies and show significant protein abundance differences. Another challenge is the proteome's immense intricacy, which means a multitude of proteins (secreted proteins, intracellular proteins, and surface markers) can be spliced, changed, and activated in the same cell on various epitopes. To combat these challenges , extremely sensitive assays for SC proteomics measurement are required (Yang, George et al. 2020). These assays are grouped into three classes:

(i) *Flow cytometry techniques including mass flow cytometry, Fluorescence flow cytometry, and enzyme-linked immunospot (ELISpot).*
(ii) *Technologies based on microchips such as image cytometry, micro engraving, single-cell barcode microchip (SCBC), and SC western blotting.*
(iii) *DNA barcoding methods including DNA barcoded antibodies (Li, Yan et al. 2018).*

Despite this importance of SC proteomics, computational tools are nescient. Given the pace of advances in this field and the growth of multiplexing capacity, more specialized computational tools for handling and analyzing SC proteomics are much needed. Moreover, DL methods for analyzing such data can be paramount to successful studies, since DL-based models can provide a comprehensive view of proteome, offering significant potential for identifying relevant patterns and information within samples. We believe that the ongoing development of DL methods in analyzing SC proteomics can shed light on the heterogeneity of cells and cellular subprocesses, therefore result in unparallel advances in biomedical applications.

3.4 DL (ML) in Spatial Transcriptomics

Since being named the method of the year (Marx, 2021), spatial transcriptomics (ST) is becoming the natural extension of scRNA-seq, unbiasedly profiling transcriptome-wide gene expression. By not requiring tissue dissociation, spatial transcriptomic retain spatial information, adding a spatial component to conventional RNA-sequencing technologies. ST have the potential of revolutionizing the field by bridging the gap between the deep characterization of cellular states and the cellular diversity that constitutes tissue organization. Spatially resolved transcriptomics can provide the genetic profiles of cells while containing information about the positional distribution of the sequenced cells, enhancing our understanding of cell interactions, organ function and pathology. However, the high-throughput spatial characterization of complex tissues remains a challenge. Broadly, spatially-resolved transcriptomics techniques can be divided into two categories: (i) the cyclic RNA imaging techniques that achieve single-cell resolution and (ii) array-based spatially resolved RNA-seq techniques, such as Visium Spatial Transcriptomics (Ståhl, Salmén et al. 2016), Slide-sequencing (Rodriques, Stickels et al. 2019), or HDST (Vickovic, Eraslan et al. 2019).

Though both subgroups can provide spatial information on a single-cell level, the cyclic RNA methods are limited on the number of genes that they can multiplex. On the other hand, the array-based spatially resolved RNA-seq techniques achieve high-throughput data by capturing mRNA across thin tissue sections using a grid of microarrays or bead-arrays, relying on simple molecular biology and histology protocols. However, since array-based mRNA capture does not match cellular boundaries, spatial RNA-seq measurements are a combination of multiple cell-type gene expressions that can correspond either to multiple cells (Visium) or fractions of multiple cells (depending on the spatial resolution of each method). To obtain a comprehensive characterization of underlying tissue, there is the need for computational methods that can produce coupled single-cell and spatially resolved transcriptomics strategy, mapping cellular profiles into a spatial context. Nonetheless, there are some techniques for retrieving relevant biological information from ST data. As remarked by (Lähnemann, Köster et al. 2020), detecting spatial gene expression patterns is one of the most pressing challenges in single-cell omics data science. Identifying such patterns can provide valuable insight on the spatial

distribution of cell populations, pointing out gene marker candidates and potentially leading to identification of new rare cell subpopulations. Moreover, ST not only puts gene expression into a spatial context but also facilitate the integration of tissue-image information with gene expression information. Such data integration will enable researchers to utilize image processing techniques to investigate the morphological information, gaining more intuition in order to obtain more refined inferences, predictions, or cellular profiles. After addressing the mapping issues from gene expression to the spatial coordinates, further computational tools will be required to study cell-cell interactions within tissues and to model transcriptional relations between cell types (Pham, Tan et al. 2020). Given the recency of ST, ML- and DL-based models for studying this type of data are rare and not fully developed. Given the complexity of the ST space, however, we predict that DL models will be the predominant method of choice for ST data integration and analysis, with perhaps many new models borrowing ideas and designs from the existing body of work in computer vision.

3.5 Integrating scRNA-seq and Spatial Transcriptomics (Spot Deconvolution)

Given that spatial transcriptomics (ST) methods normally detect mRNA expression from a mixture of cells, and that they do not distribute the sequenced samples to match cellular boundaries, it is imperative to integrate ST data with scRNA-seq data to obtain a comprehensive mapping. Using scRNA-seq as a reference, this integration aims to infer which cell belong to which gene expression count detected at the various location across the tissue. Following traditional ML trends, computational approaches such as SPOTlight (Elosua-Bayes et al. 2021) are built on seeded Non-Negative Matrix Factorization, which allows to model count data and obtain cell-type-specific profiles that are representatives of the gene expressions associated with the different cells. This process is further refined by a proper initialization of the method that uses unique marker genes of specific cell types. Next, the method uses Non-Negative Least Squares for deconvoluting the captured expression of each spot (location). Even though this method achieves robust results while maintaining computational efficiency, there is a lack of flexibility to integrate datasets from different batches or sequencing technologies. In SPOTlight, several aspects of the underlying biological and technological variance in the data are not addressed nor accounted for, which limits its application. Furthermore, technical procedures to obtain RNA-seq data are notoriously delicate and subjected to notable sources of variation.

Methods like Seurat 3 (Stuart, Butler et al. 2019) make an effort to account for the intrinsic technical variability RNA-seq procedures thorough developing an "anchor"-based method for integrating datasets. However, it is essential to remember that there is intrinsic biological variability regarding the number of cells across positions or the amount of mRNA expressed by each cell or cell type. Therefore, there is unmet need for techniques that can integrate datasets while properly accounting for any source of variability; thus, making statistical approaches viable candidates. Approaches such as Stereotipe (Andersson, Bergenstråhle et al. 2020) frame their model on a statistical framework, modeling gene expressions counts as occurrences under a negative binomial distribution. This model follows previous approaches of obtaining a gene expression profile for each cell type; Andersson et al. follow a two-step approach: First, they estimate the parameter of the negative binomial distribution for all genes within each cell type. Similar parameters for a distribution of the RNA-seq and spatial expression mixture are then formed by a linear combination of the single-cell parameters. The next step is to search a set of weights that can best fit the spatial data (Andersson, Bergenstråhle et al. 2020). The computed weights will reflect the contribution of each cell type to the gene expression counts found in each location, thus explaining the abundance of each cell type across the spots. Following the previous methodology, cell2location (Kleshchevnikov et al. 2020) builds on a Bayesian framework and models gene expression counts as a negative binomial distribution. This approach allows for controlling the sources of variability which is crucial when working with data from different technologies. This method integrates scRNA-seq information into the spatial model on the same statistical framework.

In addition to modeling the gene-specific unobserved rate (mean) as a weighted sum of the cell signature gene expression, it cell2location also adds various parameters to provide the model with prior information regarding technology sensitivity. That is, to further improve integration between different technologies, Kleshchevnikov et al. allow for four hyperparameters which aim to define informative priors on both variables; these hyperparameters are (i) expected number of cells per location, (ii) the expected number of cell types per spot, (iii) the expected number of co-located cell type groups per location and (iv) mean and variance that allow users to define prior beliefs on the sensitivity of spatial technology compared with the scRNA-seq reference. These parameters will scale the weighted sum of cell contributions, and additive shift parameters, for accounting for gene- and location-specific shift, such as due to contaminating or free-floating RNA.

3.6 DL in the Integration of SC Multimodal Omics Data

SC sequencing was chosen as Method of the Year in 2013 due to its ability to sequence DNA and RNA in individual cells (Teichmann and Efremova 2020). SC sequencing allows for gene expression measurements at an unprecedented single-cell resolution, which can provide a comprehensive view of the genome, transcriptome, or epigenome. However, it was essential to integrate all omics data for one sample simultaneously in order to gain an accurate and comprehensive view of the cellular composition in control (normal) and disease conditions (Wani and Raza 2019). Recent technological advances now allow for multimodal omics measurements from the same experiment; these technologies can assess various modalities in one experiment (i.e., conduct multimodal studies) or integrate diverse omics datasets from multiple experiments. Given the enormous potential of these approaches, single-cell multimodal omics was named the 2019

Method of the Year (Teichmann and Efremova 2020). Omics integration holds the promise of linking even small datasets across orthogonal biochemical domains, amplifying biologically significant signals in the process (Grapov, Fahrmann et al. 2018). By analyzing multi-omics data, researchers can produce novel hypotheses or design mathematical algorithms for prediction tasks, such as drug sensitivity and efficacy, gene dependence prediction, and patient stratification. Multi-omics data reflect molecular phenotypes at different molecular systems, and thus each omics dataset could follow a different and specific distribution. This fact poses unique challenges for holistic integration, in addition to the other traditional difficulties such as batch effects from multiple sources. To overcome these hurdles, sophisticated statistical and computational strategies are required. Among the various proposed algorithms thus far, only DL-based algorithms provide the computational versatility necessary to effectively model and incorporate virtually any form of omic data in an unsupervised or supervised manner (Grapov, Fahrmann et al. 2018).

Hao et al (Hao, Hao et al. 2020) proposed the *weighted-nearest neighbor study*, an unsupervised framework for defining cellular identity by leveraging multiple data types for creating a multimodal reference atlas. They apply their technique to a dataset of human PBMCs that includes paired transcriptomes and measurements of 228 surface proteins, forming a multimodal immune system atlas. They evaluate multimodal datasets from single cells, including paired measurements of RNA and chromatin state, and extend beyond the transcriptome to define cellular identity in a coherent and multimodal manner (Hao, Hao et al. 2020).

On the other hand, some algorithms in this area aim to simultaneously calculate several modalities in a single experiment. For example, Zuo et al. (Zuo and Chen 2020) introduced a single-cell multimodal VAE model (scMVAE) for profiling both transcriptomic and chromatin accessibility information in the same individual cells. Given the scRNA-seq and scATAC-seq of the same individual cells, scMVAE's uses three joint-learning strategies to learn a non-linear joint embedding that can be used for various downstream tasks (e.g. clustering). This joint learning distinguishes scMVAE from other VAE-based models (such as scVI) which processes individual omics data separately. Zuo et al. note that scMVAE's feature embeddings are more distinct than the scVI's for each omics data, indicating that the joint learning representation of multi-omics data will produce a more robust and more valuable representation (Zuo and Chen 2020).

Currently there are only a few studies that have used DL for data integration, but their success thus far calls for additional investigation of DL models for this domain. Despite the significant advances made with single-cell multimodal omics technologies, several obstacles remain: First, these techniques are prohibitively expensive when used on a large scale to analyze complex heterogeneous samples and distinguish rare cell types within a tissue. On the other hand, data sparsity is a significant limitation of high-throughput single-cell multimodal omics assays. Furthermore, existing methods cover only a small portion of the epigenome and transcriptome of individual cells, making it challenging to separate technical noise from cell-to-cell variability. While future modification of these approaches will eventually close the gap, fundamentally new algorithms or strategies may be needed to resolve this constraint completely (Zhu, Preissl et al. 2020). Amodio et al. (Amodio et al. 2018) propose Manifold-Aligning GAN (MAGAN), which is a GAN-based model which aligns two manifolds coming from different domains with the assumption that different measurements for the same underlying system contain complementary information. They show MAGAN's potential in the problem of single-cell data integration (CyTOF and scRNA-seq data) and demonstrate the generalization potential of this method in the integration of other data types. However, the performance of MAGAN decreases in the absence of correspondence complementary information among samples (Liu, Huang et al. 2019). Cao et al. (Cao et al., 2020) introduced UnionCom for unsupervised topological alignment of single-cell omics integration without a need for correspondence information among cells or among features, which can be very useful in data integration. However, UnionCom is not scalable to large datasets in the order of millions of cells (Cao, Bai et al. 2020). Other models such as SMILE (Single-cell Mutual Information Learning) (Xu, Das et al. 2021) also allow for unmatched feature types. SMILE is a deep clustering algorithm for different tissues and modalities, even when the feature types are unmatched. SMILE removes batch effects and learns a discriminative representation for data integration using a cell-pairing maximization algorithm (Xu, Das et al. 2021).

SCIM (Single-Cell Data Integration via Matching) (Stark, Ficek et al. 2020) is a deep generative approach that constructs a technology-invariant latent space to recover cell correspondences among datasets, even with unpaired feature sets. The architecture is a modified auto-encoder with an integrated discriminator network, similar to the one in GANs, allowing the network to be trained in an adversarial manner. Multi-modal datasets are integrated by pairing cells across technologies using a bipartite matching scheme that operates on the low-dimensional latent representations (Stark, Ficek et al. 2020). Another data integration model is GLUER (Peng, Chen et al. 2021), which employs three computational approaches: a nonnegative matrix factorization (NMF), mutual nearest neighbor, and a DL neural network to integrate multi-omics data. The NMF stage helps to identify shared signals across data sets of different modalities, resulting in "factor loading matrices" (FLM) for each modality. The FLM from one data modality is defined as the reference matrix while the other FLM are used as query matrices, which are used to calculate putative cell pairs. These cell pair are then used in a DNN that aims to learn a mapping between the query FLMs and the reference FLMs, resulting in co-embedded datasets (Peng, Chen et al. 2021).

Some studies have formulated the integration problem as a transfer-learning task. For example, Lin et al. (Lin, Wu et al. 2021) pose the integration problem as a transfer learning question, where the model is co-trained on labeled RNA and unlabeled ATAC data. Their model, scJoint, integrates atlas-scale collections of scRNA-seq and scATAC-seq data, using a neural network framework (Lin, Wu et al. 2021). Their approach uses scATAC-seq to gain a complementary layer of information at the single-cell resolution which is then added to the gene expression data from scRNA-seq. However,

scJoint requires that both input matrices share the same dimensions, and scATAC-seq is first converted to gene activity scores, where a single encoder can share the weights for both RNA and ATAC data (Lin, Wu et al. 2021).

4. Conclusion

Single-cell omics methods generate massive amounts of data that describe the genomic, transcriptomic, or epigenomic profiles of many individual cells in parallel. Integrative methods have recently opened a new way for delineating heterogeneous mechanistic landscapes and cell-cell interactions in single-cell multi-omics. There has been a surge in using deep learning-based models for such datasets, given the challenges in inferring biological information and constructing predictive models. So far, deep learned models have shown promising results, demonstrating the ability to process and learn from massive quantities of high-dimensional representations of single-cell and multi-omics data. However, a thorough understanding of the underlying models is needed to evaluate the optimal pipeline for analyzing single-cell datasets in order to understand cellular identity and function better. We believe that the single-cell sequencing space will pose unique challenges in representation and deep learning, particularly in multi-omics applications. We foresee an increase in the use of deep learning techniques for addressing these difficulties. Furthermore, we believe that the biological generalizability and interpretability of deep-learned models in understanding complex pathological phenotypes would be of great interest and importance to the genomics field.

References

Akay, A. and H. Hess (2019). "Deep learning: current and emerging applications in medicine and technology." IEEE journal of biomedical health informatics **23**(3): 906-920.

Amodio, M. & Krishnaswamy, S. (2018). MAGAN: Aligning Biological Manifolds. *Proceedings of the 35th International Conference on Machine Learning*, in *Proceedings of Machine Learning Research.* 80:215-223 Available from https://proceedings.mlr.press/v80/amodio18a.html.

Albrecht, S., T. Andreani, M. A. Andrade-Navarro and J.-F. Fontaine (2021). "Interpretable machine learning models for single-cell ChIP-seq imputation." bioRxiv: 2019.2012. 2020.883983.

Alipanahi, B., A. Delong, M. T. Weirauch and B. J. Frey (2015). "Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning." Nature biotechnology **33**(8): 831-838.

Almet, A. A., Z. Cang, S. Jin and Q. Nie (2021). "The landscape of cell-cell communication through single-cell transcriptomics." Current Opinion in Systems Biology.

Amodio, M., D. Van Dijk, K. Srinivasan, W. S. Chen, H. Mohsen, K. R. Moon, A. Campbell, Y. Zhao, X. Wang and M. Venkataswamy (2019). "Exploring single-cell data with deep multitasking neural networks." Nature methods **16**(11): 1139-1145.

Anchang, B., T. D. Hart, S. C. Bendall, P. Qiu, Z. Bjornson, M. Linderman, G. P. Nolan and S. K. Plevritis (2016). "Visualization and cellular hierarchy inference of single-cell data using SPADE." Nature protocols **11**(7): 1264-1279.

Andersson, A., J. Bergenståhle, M. Asp, L. Bergenståhle, A. Jurek, J. F. Navarro and J. J. C. b. Lundeberg (2020). "Single-cell and spatial transcriptomics enables probabilistic inference of cell type topography." **3**(1): 1-8.

Angermueller, C., H. J. Lee, W. Reik and O. Stegle (2017). "DeepCpG: accurate prediction of single-cell DNA methylation states using deep learning." Genome biology **18**(1): 1-13.

Arisdakessian, C., O. Poirion, B. Yunits, X. Zhu and L. X. Garmire (2019). "DeepImpute: an accurate, fast, and scalable deep neural network method to impute single-cell RNA-seq data." Genome biology **20**(1): 1-14.

Arjovsky, M., S. Chintala and L. Bottou (2017). Wasserstein generative adversarial networks. International conference on machine learning, PMLR.

ARMACKI, A. (2018). Application of Autoencoders on Single-cell Data, University OF Novi Sad.

Atta, L. and J. Fan (2021). "VeloViz: RNA-velocity informed 2D embeddings for visualizing cellular trajectories." bioRxiv.

Azizi, E., A. J. Carr, G. Plitas, A. E. Cornish, C. Konopacki, S. Prabhakaran, J. Nainys, K. Wu, V. Kiseliovas and M. Setty (2018). "Single-cell map of diverse immune phenotypes in the breast tumor microenvironment." Cell **174**(5): 1293-1308. e1236.

Bacher, R., L.-F. Chu, N. Leng, A. P. Gasch, J. A. Thomson, R. M. Stewart, M. Newton and C. Kendziorski (2017). "SCnorm: robust normalization of single-cell RNA-seq data." Nature methods **14**(6): 584.

Badsha, M. B., R. Li, B. Liu, Y. I. Li, M. Xian, N. E. Banovich and A. Qiuyan (2020). "Imputation of single-cell gene expression with an autoencoder neural network Running title: Autoencoder for imputation of single-cell gene expression." Quantitative Biology **8**(1): 78-94.

Baek, S. and I. Lee (2020). "Single-cell ATAC sequencing analysis: From data preprocessing to hypothesis generation." Computational and structural biotechnology journal.

Bahdanau, D., J. Chorowski, D. Serdyuk, P. Brakel and Y. Bengio (2016). End-to-end attention-based large vocabulary speech recognition. 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE.

Becht, E., L. McInnes, J. Healy, C.-A. Dutertre, I. W. Kwok, L. G. Ng, F. Ginhoux and E. W. Newell (2019). "Dimensionality reduction for visualizing single-cell data using UMAP." Nature biotechnology **37**(1): 38-44.

Bergen, V., M. Lange, S. Peidli, F. A. Wolf and F. J. Theis (2020). "Generalizing RNA velocity to transient cell states through dynamical modeling." Nature biotechnology **38**(12): 1408-1414.

Briggs, J. A., C. Weinreb, D. E. Wagner, S. Megason, L. Peshkin, M. W. Kirschner and A. M. Klein (2018). "The dynamics of gene expression in vertebrate embryogenesis at single-cell resolution." Science **360**(6392).

Buettner, F., K. N. Natarajan, F. P. Casale, V. Proserpio, A. Scialdone, F. J. Theis, S. A. Teichmann, J. C. Marioni and O. Stegle (2015). "Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells." Nature biotechnology **33**(2): 155-160.

Buettner, F. and F. J. Theis (2012). "A novel approach for resolving differences in single-cell gene expression patterns from zygote to blastocyst." Bioinformatics **28**(18): i626-i632.

Büttner, M., Z. Miao, F. A. Wolf, S. A. Teichmann and F. J. Theis (2019). "A test metric for assessing single-cell RNA-seq batch correction." Nature methods **16**(1): 43-49.

Cao, J., M. Spielmann, X. Qiu, X. Huang, D. M. Ibrahim, A. J. Hill, F. Zhang, S. Mundlos, L. Christiansen and F. Steemers (2019). "The single-cell transcriptional landscape of mammalian organogenesis." Nature **566**(7745): 496-502.

Cao, K., X. Bai, Y. Hong and L. Wan (2020). "Unsupervised topological alignment for single-cell multi-omics integration." Bioinformatics **36**(Supplement_1): i48-i56.

Cao, Y., L. Fu, J. Wu, Q. Peng, Q. Nie, J. Zhang and X. Xie (2021). "SAILER: Scalable and Accurate Invariant Representation Learning for Single-Cell ATAC-Seq Processing and Integration." bioRxiv.

Chellapilla, K., S. Puri and P. Simard (2006). High performance convolutional neural networks for document processing. Tenth International Workshop on Frontiers in Handwriting Recognition, Suvisoft.

Chen, L., W. Wang, Y. Zhai and M. Deng (2020). "Deep soft K-means clustering with self-training for single-cell RNA sequence data." NAR Genomics and Bioinformatics **2**(2): lqaa039.

Davie, K., J. Janssens, D. Koldere, M. De Waegeneer, U. Pech, Ł. Kreft, S. Aibar, S. Makhzami, V. Christiaens and C. B. González-Blas (2018). "A single-cell transcriptome atlas of the aging Drosophila brain." Cell **174**(4): 982-998. e920.

Deng, Y., F. Bao, Q. Dai, L. F. Wu and S. J. Altschuler (2019). "Scalable analysis of cell-type composition from single-cell transcriptomics using deep recurrent learning." Nature methods **16**(4): 311-314.

Ding, B., L. Zheng, Y. Zhu, N. Li, H. Jia, R. Ai, A. Wildberg and W. Wang (2015). "Normalization and noise reduction for single cell RNA-seq experiments." Bioinformatics **31**(13): 2225-2227.

Ding, J., A. Condon and S. P. Shah (2018). "Interpretable dimensionality reduction of single cell transcriptome data with deep generative models." Nature communications **9**(1): 1-13.

Dziugaite, G. K., D. M. Roy and Z. Ghahramani (2015). "Training generative neural networks via maximum mean discrepancy optimization." arXiv preprint arXiv:1505.03906.

Engel, J., K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue and A. Roberts (2019). "Gansynth: Adversarial neural audio synthesis." arXiv preprint arXiv:1902.08710.

Eraslan, G., L. M. Simon, M. Mircea, N. S. Mueller and F. J. Theis (2019). "Single-cell RNA-seq denoising using a deep count autoencoder." Nature communications **10**(1): 1-14.

Esteban, C., S. L. Hyland and G. Rätsch (2017). "Real-valued (medical) time series generation with recurrent conditional gans." arXiv preprint arXiv:1706.02633.

Fedus, W., I. Goodfellow and A. M. Dai (2018). "Maskgan: better text generation via filling in the _." arXiv preprint arXiv:1801.07736.

Fei, T. and T. Yu (2020). "scBatch: batch-effect correction of RNA-seq data through sample distance matrix adjustment." Bioinformatics.

Finak, G., A. McDavid, M. Yajima, J. Deng, V. Gersuk, A. K. Shalek, C. K. Slichter, H. W. Miller, M. J. McElrath and M. Prlic (2015). "MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data." Genome biology **16**(1): 1-13.

Fritzsch, F. S., C. Dusny, O. Frick and A. Schmid (2012). "Single-cell analysis in biotechnology, systems biology, and biocatalysis." Annual review of chemical biomolecular engineering **3**: 129-155.

Fu, L., L. Zhang, E. Dollinger, Q. Peng, Q. Nie and X. Xie (2020). "Predicting transcription factor binding in single cells through deep learning." Science Advances **6**(51): eaba9031.

Furey, T. S. (2012). "ChIP–seq and beyond: new and improved methodologies to detect and characterize protein–DNA interactions." Nature Reviews Genetics **13**(12): 840-852.

Girshick, R., J. Donahue, T. Darrell and J. Malik (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition.

Gong, W., I.-Y. Kwak, P. Pota, N. Koyano-Nakagawa and D. J. Garry (2018). "DrImpute: imputing dropout events in single cell RNA sequencing data." BMC bioinformatics **19**(1): 220.

Goodfellow, I., Y. Bengio, A. Courville and Y. Bengio (2016). Deep learning, MIT press Cambridge.

Goodfellow, I. J., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio (2014). "Generative adversarial networks." arXiv preprint arXiv:1406.2661.

Grapov, D., J. Fahrmann, K. Wanichthanarak and S. Khoomrung (2018). "Rise of deep learning for genomic, proteomic, and metabolomic data integration in precision medicine." Omics: a journal of integrative biology **22**(10): 630-636.

Griffiths, J. A., A. Scialdone and J. C. Marioni (2018). "Using single-cell genomics to understand developmental processes and cell fate decisions." Molecular systems biology **14**(4): e8046.

Grønbech, C. H., M. F. Vording, P. N. Timshel, C. K. Sønderby, T. H. Pers and O. Winther (2020). "scVAE: Variational auto-encoders for single-cell gene expression data." Bioinformatics **36**(16): 4415-4422.

Guo, X., E. Zhu, X. Liu and J. Yin (2018). Deep embedded clustering with data augmentation. Asian conference on machine learning.

Haber, A. L., M. Biton, N. Rogel, R. H. Herbst, K. Shekhar, C. Smillie, G. Burgin, T. M. Delorey, M. R. Howitt and Y. Katz (2017). "A single-cell survey of the small intestinal epithelium." Nature **551**(7680): 333-339.

Haghverdi, L., F. Buettner and F. J. Theis (2015). "Diffusion maps for high-dimensional single-cell analysis of differentiation data." Bioinformatics **31**(18): 2989-2998.

Haghverdi, L., A. T. Lun, M. D. Morgan and J. C. Marioni (2018). "Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors." Nature biotechnology **36**(5): 421-427.

Han, X., R. Wang, Y. Zhou, L. Fei, H. Sun, S. Lai, A. Saadatpour, Z. Zhou, H. Chen and F. Ye (2018). "Mapping the mouse cell atlas by microwell-seq." Cell **172**(5): 1091-1107. e1017.

Han, X., Z. Zhou, L. Fei, H. Sun, R. Wang, Y. Chen, H. Chen, J. Wang, H. Tang and W. Ge (2020). "Construction of a human cell landscape at single-cell level." Nature **581**(7808): 303-309.

Hannun, A., C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta and A. Coates (2014). "Deep speech: Scaling up end-to-end speech recognition." arXiv preprint arXiv:1412.5567.

Hao, Y., S. Hao, E. Andersen-Nissen, W. M. Mauck, S. Zheng, A. Butler, M. J. Lee, A. J. Wilk, C. Darby and M. Zagar (2020). "Integrated analysis of multimodal single-cell data." bioRxiv.

He, J., D. Spokoyny, G. Neubig and T. Berg-Kirkpatrick (2019). "Lagging inference networks and posterior collapse in variational autoencoders." arXiv preprint arXiv:1901.05534.

Hentges, L. D., M. D. Sergeant, D. J. Downes, J. R. Hughes and S. Taylor (2021). "LanceOtron: a deep learning peak caller for ATAC-seq, ChIP-seq, and DNase-seq." bioRxiv.

Heydari, A. A., O. A. Davalos, L. Zhao, K. K. Hoyer and S. S. Sindi (2021). "ACTIVA: realistic single-cell RNA-seq generation with automatic cell-type identification using introspective variational autoencoders." bioRxiv.

Heydari, A. A. and A. Mehmood (2020). SRVAE: super resolution using variational autoencoders. Pattern Recognition and Tracking XXXI, International Society for Optics and Photonics.

Heydari, A. A., Thompson C., and Mehmood A. (2019). SoftAdapt: Techniques for Adaptive Loss Weighting of Neural Networks with Multi-Part Loss Functions. ArXiv 1912.1235. Available from https://arxiv.org/abs/1912.12355

Hie, B., B. Bryson and B. Berger (2019). "Efficient integration of heterogeneous single-cell transcriptomes using Scanorama." Nature biotechnology **37**(6): 685-691.

Hinton, G. E., Simon O, Yee-Whye T. (2006). "Deep belief networks." Neural Computation, 18:1527-1554.

Hogan, S. A., A. Courtier, P. F. Cheng, N. F. Jaberg-Bentele, S. M. Goldinger, M. Manuel, S. Perez, N. Plantier, J.-F. Mouret and T. D. L. Nguyen-Kim (2019). "Peripheral blood TCR repertoire profiling may facilitate patient stratification for immunotherapy against melanoma." Cancer immunology research **7**(1): 77-85.

Hosokawa, M., Y. Nishikawa, M. Kogawa and H. Takeyama (2017). "Massively parallel whole genome amplification for single-cell sequencing using droplet microfluidics." Scientific reports **7**(1): 1-11.

Hochreiter, S., and Schmidhuber, Jurgen. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hua, Y., Z. Zhao, R. Li, X. Chen, Z. Liu and H. Zhang (2019). "Deep learning with long short-term memory for time series prediction." IEEE Communications Magazine **57**(6): 114-119.

Huang, H., Z. Li, R. He, Z. Sun and T. Tan (2018). "Introvae: Introspective variational autoencoders for photographic image synthesis." arXiv preprint arXiv:1807.06358.

Huang, M., J. Wang, E. Torre, H. Dueck, S. Shaffer, R. Bonasio, J. I. Murray, A. Raj, M. Li and N. R. Zhang (2018). "SAVER: gene expression recovery for single-cell RNA sequencing." Nature methods **15**(7): 539-542.

Hubel, D. H. and T. N. Wiesel (1962). "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." The Journal of physiology **160**(1): 106-154.

Jiang, P., J. A. Thomson and R. Stewart (2016). "Quality control of single-cell RNA-seq by SinQC." Bioinformatics **32**(16): 2514-2516.

Johnson, W. E., C. Li and A. Rabinovic (2007). "Adjusting batch effects in microarray expression data using empirical Bayes methods." Biostatistics **8**(1): 118-127.

Kharchenko, P. V., L. Silberstein and D. T. J. N. m. Scadden (2014). "Bayesian approach to single-cell differential expression analysis." **11**(7): 740-742.

Kimmel, J. C., A. S. Brack and W. F. J. b. Marshall (2019). "Deep convolutional and recurrent neural networks for cell motility discrimination and prediction." 159202.

Kingma, D. P. and J. Ba (2015). "Adam: A Method for Stochastic Optimization." arXiv preprint.

Kingma, D. P. and M. Welling (2013). "Auto-encoding variational bayes." arXiv preprint arXiv:1312.6114.

Kingma, D. P. and M. Welling (2019). "An introduction to variational autoencoders." arXiv preprint arXiv:1906.02691.

Klein, A. M., L. Mazutis, I. Akartuna, N. Tallapragada, A. Veres, V. Li, L. Peshkin, D. A. Weitz and M. W. Kirschner (2015). "Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells." Cell **161**(5): 1187-1201.

Krizhevsky, A., I. Sutskever, G. E. Hinton, F. Pereira, C. Burges, L. Bottou and K. Weinberger (2012). "Advances in neural information processing systems."

Lafzi, A., C. Moutinho, S. Picelli and H. Heyn (2018). "Tutorial: guidelines for the experimental design of single-cell RNA sequencing studies." Nature protocols **13**(12): 2742-2757.

Lähnemann, D., J. Köster, E. Szczurek, D. J. McCarthy, S. C. Hicks, M. D. Robinson, C. A. Vallejos, K. R. Campbell, N. Beerenwinkel and A. J. G. b. Mahfouz (2020). "Eleven grand challenges in single-cell data science." **21**(1): 1-35.

Larsen, A. B. L., S. K. Sønderby, H. Larochelle and O. Winther (2016). Autoencoding beyond pixels using a learned similarity metric. International conference on machine learning, PMLR.

LeCun, Y., Y. Bengio and G. Hinton (2015). "Deep learning." nature **521**(7553): 436-444.

LeCun, Y., Y. J. T. h. o. b. t. Bengio and n. networks (1995). "Convolutional networks for images, speech, and time series." **3361**(10): 1995.

Li, H., U. Shaham, K. P. Stanton, Y. Yao, R. R. Montgomery and Y. Kluger (2017). "Gating mass cytometry data by deep learning." Bioinformatics **33**(21): 3423-3430.

Li, L., S. Yan, B. Lin, Q. Shi and Y. Lu (2018). Single-Cell Proteomics for Cancer Immunotherapy. Advances in cancer research, Elsevier. **139:** 185-207.

Li, W. V. and J. J. Li (2018). "An accurate and robust imputation method scImpute for single-cell RNA-seq data." Nature communications **9**(1): 1-9.

Li, X., K. Wang, Y. Lyu, H. Pan, J. Zhang, D. Stambolian, K. Susztak, M. P. Reilly, G. Hu and M. Li (2020). "Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis." Nature communications **11**(1): 1-14.

Lin, P., M. Troup and J. W. Ho (2017). "CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data." Genome biology **18**(1): 59.

Lin, S., Y. Lin, J. R. Nery, M. A. Urich, A. Breschi, C. A. Davis, A. Dobin, C. Zaleski, M. A. Beer and W. C. Chapman (2014). "Comparison of the transcriptional landscapes between human and mouse tissues." Proceedings of the National Academy of Sciences **111**(48): 17224-17229.

Lin, Y., T.-Y. Wu, S. Wan, J. Y. Yang, Y. R. Wang and W. H. Wong (2021). "scJoint: transfer learning for data integration of single-cell RNA-seq and ATAC-seq." bioRxiv: 2020.2012. 2031.424916.

Linker, S. M., L. Urban, S. J. Clark, M. Chhatriwala, S. Amatya, D. J. McCarthy, I. Ebersberger, L. Vallier, W. Reik and O. Stegle (2019). "Combined single-cell profiling of expression and DNA methylation reveals splicing regulation and heterogeneity." Genome biology **20**(1): 1-14.

Liu, J., Y. Huang, R. Singh, J.-P. Vert and W. S. Noble (2019). "Jointly embedding multiple single-cell omics measurements." BioRxiv: 644310.

Liu, Q. and L. X. J. P. c. b. Di He (2019). "Prediction of off-target specificity and cell-specific fitness of CRISPR-Cas System using attention boosted deep learning and network-based gene feature." **15**(10).

Long, J., E. Shelhamer and T. Darrell (2015). Fully convolutional networks for semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition.

Lopez, R., J. Regier, M. B. Cole, M. I. Jordan and N. Yosef (2018). "Deep generative modeling for single-cell transcriptomics." Nature methods **15**(12): 1053-1058.

Love, M. I., W. Huber and S. Anders (2014). "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2." Genome biology **15**(12): 1-21.

Lucas, J., G. Tucker, R. B. Grosse and M. Norouzi (2019). "Don't blame the Elbo! a linear Vae perspective on posterior collapse." Advances in Neural Information Processing Systems **32**: 9408-9418.

Lucic, M., K. Kurach, M. Michalski, S. Gelly and O. Bousquet (2017). "Are gans created equal? a large-scale study." arXiv preprint arXiv:1711.10337.

Luecken, M. D. and F. J. Theis (2019). "Current best practices in single-cell RNA-seq analysis: a tutorial." Molecular systems biology **15**(6): e8746.

Lun, A. T., K. Bach and J. C. Marioni (2016). "Pooling across cells to normalize single-cell RNA sequencing data with many zero counts." Genome biology **17**(1): 1-14.

Luo, X. and Y. Wei (2019). "Batch effects correction with unknown subtypes." Journal of the American Statistical Association **114**(526): 581-594.

Marouf, M., P. Machart, V. Bansal, C. Kilian, D. S. Magruder, C. F. Krebs and S. Bonn (2020). "Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks." Nature communications **11**(1): 1-12.

Märtens, K. and C. Yau (2020). "BasisVAE: Translation-invariant feature-level clustering with Variational Autoencoders." arXiv preprint arXiv:2003.03462.

Marc Elosua-Bayes, Paula Nieto, Elisabetta Mereu, Ivo Gut, Holger Heyn, SPOTlight: seeded NMF regression to deconvolute spatial transcriptomic spots with single-cell transcriptomes, *Nucleic Acids Research*, Volume 49, Issue 9, 21 May 2021, Page e50, https://doi.org/10.1093/nar/gkab043

Marx, V. Method of the Year: spatially resolved transcriptomics. *Nat Methods* **18**, 9–14 (2021). https://doi.org/10.1038/s41592-020-01033-y

Metz, L., B. Poole, D. Pfau and J. Sohl-Dickstein (2016). "Unrolled generative adversarial networks." arXiv preprint arXiv:1611.02163.

Minakshi, P., R. Kumar, M. Ghosh, H. M. Saini, K. Ranjan, B. Brar and G. Prasad (2019). Single-Cell proteomics: technology and applications. Single-Cell Omics, Elsevier**:** 283-318.

Mongia, A., D. Sengupta and A. Majumdar (2020). "deepmc: Deep matrix completion for imputation of single-cell RNA-seq data." Journal of Computational Biology **27**(7): 1011-1019.

Montoro, D. T., A. L. Haber, M. Biton, V. Vinarsky, B. Lin, S. E. Birket, F. Yuan, S. Chen, H. M. Leung and J. Villoria (2018). "A revised airway epithelial hierarchy includes CFTR-expressing ionocytes." Nature **560**(7718): 319-324.

Moussa, M. and I. I. Măndoiu (2019). "Locality sensitive imputation for single cell RNA-seq data." Journal of Computational Biology **26**(8): 822-835.

Nair, V. and G. E. Hinton (2010). Rectified linear units improve restricted boltzmann machines. Icml.

Parekh, S., C. Ziegenhain, B. Vieth, W. Enard and I. Hellmann (2018). "zUMIs-a fast and flexible pipeline to process RNA sequencing data with UMIs." Gigascience.

Patruno, L., D. Maspero, F. Craighero, F. Angaroni, M. Antoniotti and A. Graudenzi (2020). "A review of computational strategies for denoising and imputation of single-cell transcriptomic data." Briefings in Bioinformatics **22**(4).

Peng, J., X. Wang and X. Shang (2019). "Combining gene ontology with deep neural networks to enhance the clustering of single cell RNA-Seq data." BMC bioinformatics **20**(8): 284.

Peng, T., G. Chen and K. Tan (2021). "GLUER: integrative analysis of single-cell omics and imaging data by deep neural network." bioRxiv.

Pham, D. T., X. Tan, J. Xu, L. F. Grice, P. Y. Lam, A. Raghubar, J. Vukovic, M. J. Ruitenberg and Q. H. J. b. Nguyen (2020). "stLearn: integrating spatial location, tissue morphology and gene expression to find cell types, cell-cell interactions and spatial trajectories within undissociated tissues."

Plass, M., J. Solana, F. A. Wolf, S. Ayoub, A. Misios, P. Glažar, B. Obermayer, F. J. Theis, C. Kocks and N. Rajewsky (2018). "Cell type atlas and lineage tree of a whole complex animal by single-cell transcriptomics." Science **360**(6391).

Polański, K., M. D. Young, Z. Miao, K. B. Meyer, S. A. Teichmann and J.-E. Park (2020). "BBKNN: fast batch alignment of single cell transcriptomes." Bioinformatics **36**(3): 964-965.

Prechelt, L. (1998). "Automatic early stopping using cross validation: quantifying the criteria." Neural Networks **11**(4): 761-767.

Qiu, P. (2020). "Embracing the dropouts in single-cell RNA-seq analysis." Nature communications **11**(1): 1-9.

Ran, D., S. Zhang, N. Lytal and L. An (2019). "scDoc: Correcting Drop-out Events in Single-cell RNA-seq Data." bioRxiv: 731638.

Regev, A., S. A. Teichmann, E. S. Lander, I. Amit, C. Benoist, E. Birney, B. Bodenmiller, P. Campbell, P. Carninci and M. Clatworthy (2017). "Science forum: the human cell atlas." Elife **6**: e27041.

Ritchie, M. E., B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi and G. K. Smyth (2015). "limma powers differential expression analyses for RNA-sequencing and microarray studies." Nucleic acids research **43**(7): e47-e47.

Rodriques, S. G., R. R. Stickels, A. Goeva, C. A. Martin, E. Murray, C. R. Vanderburg, J. Welch, L. M. Chen, F. Chen and E. Z. J. S. Macosko (2019). "Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution." **363**(6434): 1463-1467.

Rostom, R., V. Svensson, S. A. Teichmann and G. Kar (2017). "Computational approaches for interpreting scRNA-seq data." FEBS letters **591**(15): 2213-2225.

Roth, A., J. Khattra, D. Yap, A. Wan, E. Laks, J. Biele, G. Ha, S. Aparicio, A. Bouchard-Côté and S. P. Shah (2014). "PyClone: statistical inference of clonal population structure in cancer." Nature methods **11**(4): 396-398.

Rozenblatt-Rosen, O., M. J. Stubbington, A. Regev and S. A. Teichmann (2017). "The Human Cell Atlas: from vision to reality." Nature News **550**(7677): 451.

Rumelhart, D. E., G. E. Hinton and R. J. Williams (1986). "Learning representations by back-propagating errors." nature **323**(6088): 533-536.

Semeniuta, S., A. Severyn and E. Barth (2017). "A hybrid convolutional variational autoencoder for text generation." arXiv preprint arXiv:1702.02390.

Sengupta, D., N. A. Rayan, M. Lim, B. Lim and S. Prabhakar (2016). "Fast, scalable and accurate differential expression analysis for single cells." BioRxiv: 049734.

Shaham, U. (2018). "Batch Effect Removal via Batch-Free Encoding." bioRxiv: 380816.

Shaham, U., K. P. Stanton, J. Zhao, H. Li, K. Raddassi, R. Montgomery and Y. Kluger (2017). "Removal of batch effects using distribution-matching residual networks." Bioinformatics **33**(16): 2539-2546.

Shi, S., Q. Wang, P. Xu and X. Chu (2016). Benchmarking state-of-the-art deep learning software tools. 2016 7th International Conference on Cloud Computing and Big Data (CCBD), IEEE.

Shorten, C. and T. M. Khoshgoftaar (2019). "A survey on image data augmentation for deep learning." Journal of Big Data **6**(1): 1-48.

Ståhl, P. L., F. Salmén, S. Vickovic, A. Lundmark, J. F. Navarro, J. Magnusson, S. Giacomello, M. Asp, J. O. Westholm and M. J. S. Huss (2016). "Visualization and analysis of gene expression in tissue sections by spatial transcriptomics." **353**(6294): 78-82.

Stark, S. G., J. Ficek, F. Locatello, X. Bonilla, S. Chevrier, F. Singer, G. Rätsch and K.-V. Lehmann (2020). "SCIM: universal single-cell matching with unpaired feature sets." Bioinformatics **36**(Supplement_2): i919-i927.

Steffen Albrecht, Tommaso Andreani, Miguel A. Andrade-Navarro, Jean-Fred Fontaine. Interpretable machine learning models for single-cell ChIP-seq imputation. bioRxiv 2019

Steinkraus, D., I. Buck and P. Simard (2005). Using GPUs for machine learning algorithms. Eighth International Conference on Document Analysis and Recognition (ICDAR'05), IEEE.

Stuart, T., A. Butler, P. Hoffman, C. Hafemeister, E. Papalexi, W. M. Mauck III, Y. Hao, M. Stoeckius, P. Smibert and R. J. C. Satija (2019). "Comprehensive integration of single-cell data." **177**(7): 1888-1902. e1821.

Sun, S., J. Zhu, Y. Ma and X. Zhou (2019). "Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis." Genome biology **20**(1): 1-21.

Sun, Y., D. Liang, X. Wang and X. Tang (2015). "Deepid3: Face recognition with very deep neural networks." arXiv preprint arXiv:1502.00873.

Tabula Muris, C., c. Overall, c. Logistical, c. Organ, processing, p. Library, sequencing, a. Computational data, a. Cell type, g. Writing, g. Supplemental text writing and i. Principal (2018). "Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris." Nature **562**(7727): 367-372.

Talwar, D., A. Mongia, D. Sengupta and A. Majumdar (2018). "AutoImpute: Autoencoder based imputation of single-cell RNA-seq data." Scientific reports **8**(1): 1-11.

Tang, X., Y. Huang, J. Lei, H. Luo, X. J. C. Zhu and bioscience (2019). "The single-cell sequencing: new developments and medical applications." **9**(1): 1-9.

Teichmann, S. and M. Efremova (2020). "Method of the Year 2019: single-cell multimodal omics." Nat. Methods **17**(1).

Thibodeau, A., S. Khetan, A. Eroglu, R. Tewhey, M. L. Stitzel and D. Ucar (2020). "CoRE-ATAC: A Deep Learning model for the functional Classification of Regulatory Elements from single cell and bulk ATAC-seq data." bioRxiv.

Tian, T., J. Wan, Q. Song and Z. Wei (2019). "Clustering single-cell RNA-seq data with a model-based deep learning approach." Nature Machine Intelligence **1**(4): 191-198.

Titsias, M. and N. D. Lawrence (2010). Bayesian Gaussian process latent variable model. Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings.

Tolstikhin, I., O. Bousquet, S. Gelly and B. Schoelkopf (2017). "Wasserstein auto-encoders." arXiv preprint arXiv:1711.01558.

Tran, H. T. N., K. S. Ang, M. Chevrier, X. Zhang, N. Y. S. Lee, M. Goh and J. Chen (2020). "A benchmark of batch-effect correction methods for single-cell RNA sequencing data." Genome biology **21**(1): 1-32.

Vallejos, C. A., J. C. Marioni and S. Richardson (2015). "BASiCS: Bayesian analysis of single-cell sequencing data." PLoS computational biology **11**(6).

Van der Maaten, L. and G. Hinton (2008). "Visualizing data using t-SNE." Journal of machine learning research **9**(11).

van Dijk, D., J. Nainys, R. Sharma, P. Kaithail, A. J. Carr, K. R. Moon, L. Mazutis, G. Wolf, S. Krishnaswamy and D. Pe'er (2017). "MAGIC: A diffusion-based imputation method reveals gene-gene interactions in single-cell RNA-sequencing data." BioRxiv: 111591.

Vickovic, S., G. Eraslan, F. Salmén, J. Klughammer, L. Stenbeck, D. Schapiro, T. Äijö, R. Bonneau, L. Bergenstråhle and J. F. N. m. Navarro (2019). "High-definition spatial transcriptomics for in situ tissue profiling." **16**(10): 987-990.

Vitalii Kleshchevnikov, Artem Shmatko, Emma Dann, Alexander Aivazidis, Hamish W King, Tong Li, Artem Lomakin, Veronika Kedlian, Mika Sarkin Jain, Jun Sung Park, Lauma Ramona, Elizabeth Tuck, Anna Arutyunyan, Roser Vento-Tormo, Moritz Gerstung, Louisa James, Oliver Stegle, Omer Ali Bayraktar. bioRxiv 2020.11.15.378125; doi: https://doi.org/10.1101/2020.11.15.378125

Vondrick, C., H. Pirsiavash and A. Torralba (2016). "Generating videos with scene dynamics." arXiv preprint arXiv:1609.02612.

Wagner, D. E., C. Weinreb, Z. M. Collins, J. A. Briggs, S. G. Megason and A. M. Klein (2018). "Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo." Science **360**(6392): 981-987.

Wang, B., D. Ramazzotti, L. De Sano, J. Zhu, E. Pierson and S. Batzoglou (2017). "SIMLR: a tool for large-scale single-cell analysis by multi-kernel learning." bioRxiv: 118901.

Wang, D. and J. Gu (2018). "VASC: dimension reduction and visualization of single-cell RNA-seq data by deep variational autoencoder." Genomics, proteomics & bioinformatics **16**(5): 320-331.

Wang, T., T. S. Johnson, W. Shao, Z. Lu, B. R. Helm, J. Zhang and K. Huang (2019). "BERMUDA: a novel deep transfer learning method for single-cell RNA sequencing batch correction reveals hidden high-resolution cellular subtypes." Genome biology **20**(1): 1-15.

Wang, Z., Q. She and T. E. Ward (2021). "Generative adversarial networks in computer vision: A survey and taxonomy." ACM Computing Surveys (CSUR) **54**(2): 1-38.

Wani, N. and K. Raza (2019). "Integrative approaches to reconstruct regulatory networks from multi-omics data: a review of state-of-the-art methods." Computational biology and chemistry **83**: 107120.

Wu, Y., M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao and K. Macherey (2016). "Google's neural machine translation system: Bridging the gap between human and machine translation." arXiv preprint arXiv:1609.08144.

Xiong, L., K. Xu, K. Tian, Y. Shao, L. Tang, G. Gao, M. Zhang, T. Jiang and Q. C. Zhang (2019). "SCALE method for single-cell ATAC-seq analysis via latent feature extraction." Nature communications **10**(1): 1-10.

Xu, Y., P. Das and R. P. McCord (2021). "SMILE: Mutual Information Learning for Integration of Single Cell Omics Data." bioRxiv.

Xu, Y., Z. Zhang, L. You, J. Liu, Z. Fan and X. Zhou (2020). "scIGANs: single-cell RNA-seq imputation using generative adversarial networks." Nucleic Acids Research **48**(15): e85-e85.

Yan, F., D. R. Powell, D. J. Curtis and N. C. Wong (2020). "From reads to insight: a hitchhiker's guide to ATAC-seq data analysis." Genome biology **21**(1): 22.

Yang, L., J. George and J. Wang (2020). "Deep Profiling of Cellular Heterogeneity by Emerging Single-Cell Proteomic Technologies." Proteomics **20**(13): 1900226.

Yang, L., J. Liu, Q. Lu, A. D. Riggs and X. Wu (2017). "SAIC: an iterative clustering approach for analysis of single cell RNA-seq data." BMC genomics **18**(6): 689.

Yang, X., Y.-N. Chen, D. Hakkani-Tür, P. Crook, X. Li, J. Gao and L. Deng (2017). End-to-end joint learning of natural language understanding and dialogue manager. 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE.

Yang, Z., Z. Hu, R. Salakhutdinov and T. Berg-Kirkpatrick (2017). Improved variational autoencoders for text modeling using dilated convolutions. International conference on machine learning, PMLR.

Yuan, Y. and Z. Bar-Joseph (2019). "Deep learning for inferring gene relationships from single-cell expression data." Proceedings of the National Academy of Sciences **116**(52): 27151-27158.

Zafar, H., N. Navin, K. Chen and L. Nakhleh (2019). "SiCloneFit: Bayesian inference of population structure, genotype, and phylogeny of tumor clones from single-cell genome sequencing data." Genome research **29**(11): 1847-1859.

Zappia, L., B. Phipson and A. Oshlack (2017). "Splatter: simulation of single-cell RNA sequencing data." Genome biology **18**(1): 174.

Zhang, F., Y. Wu and W. Tian (2019). "A novel approach to remove the batch effect of single-cell data." Cell discovery **5**(1): 1-4.

Zhang, Z., F. Cui, C. Lin, L. Zhao, C. Wang and Q. Zou (2021). "Critical downstream analysis steps for single-cell RNA sequencing data." Briefings in Bioinformatics.

Zhao, S., J. Song and S. Ermon (2017). "Infovae: Information maximizing variational autoencoders." arXiv preprint arXiv:1706.02262.

Zhao, Z.-Q., P. Zheng, S.-t. Xu and X. Wu (2019). "Object detection with deep learning: A review." IEEE transactions on neural networks and learning systems **30**(11): 3212-3232.

Zheng, G. X., J. M. Terry, P. Belgrader, P. Ryvkin, Z. W. Bent, R. Wilson, S. B. Ziraldo, T. D. Wheeler, G. P. McDermott and J. Zhu (2017). "Massively parallel digital transcriptional profiling of single cells." Nature communications **8**(1): 1-12.

Zheng, J. and K. Wang (2019). "Emerging deep learning methods for single-cell RNA-seq data analysis." Quantitative Biology **7**(4): 247-254.

Zheng, R., M. Li, Z. Liang, F.-X. Wu, Y. Pan and J. Wang (2019). "SinNLRR: a robust subspace clustering method for cell type detection by non-negative and low-rank representation." Bioinformatics **35**(19): 3642-3650.

Zhou, J. and O. G. Troyanskaya (2015). "Predicting effects of noncoding variants with deep learning–based sequence model." Nature methods **12**(10): 931-934.

Zhu, C., S. Preissl and B. Ren (2020). "Single-cell multimodal omics: the power of many." Nature methods **17**(1): 11-14.

Zhu, J.-Y., P. Krähenbühl, E. Shechtman and A. A. Efros (2016). Generative visual manipulation on the natural image manifold. European conference on computer vision, Springer.

Zilionis, R., J. Nainys, A. Veres, V. Savova, D. Zemmour, A. M. Klein and L. Mazutis (2017). "Single-cell barcoding and sequencing using droplet microfluidics." Nature protocols **12**(1): 44-73.

Zuo, C. and L. Chen (2020). "Deep-joint-learning analysis model of single cell transcriptome and open chromatin accessibility data." Briefings in Bioinformatics **22**(4).