Less is more: The power of dimensionality reduction algorithms in Bioinformatics.

Student id 2490386.

02nd May 2023.

I. Introduction

I have learnt a lot as a MSc Bioinformatics student. But one thing that truly fascinated me was the first group project that was handed to us. That group project involved Single cell RNA (ScRNA) analysis of Rheumatoid arthritis samples from mice models. The most interesting tool involved in ScRNA analysis was the Principal Component Analysis, a dimensional reducing package that gave a whole picture of the distribution of the cells and was the key in mapping the cells types for understanding the disease morphology.

II. What is dimensionality reduction

Dimensional reduction tools generally work on the principle of compressing data into smaller dimensions so that they can be interpreted and visualized much easier on a 2D or 3D plot (Fig 1). This type of analysis is best suited for data having high number of features and are subjected to the "curse of dimensionality" where it becomes very complex to analyze the data using supervised learning techniques and computationally intensive [1]. Reducing the dimensions of features into chartable plots helps us illustrate the data for analysis and visualize it, while retaining maximum properties of the data, which will be impossible to do on high dimension dataset or will require multiple graphs[2].

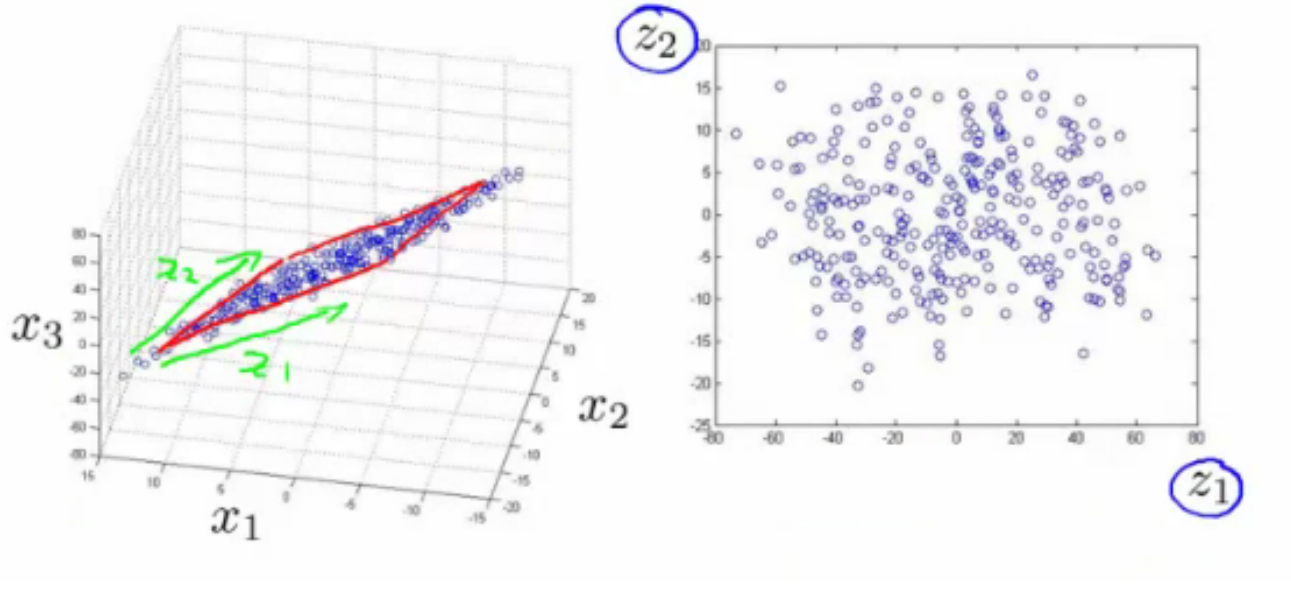

Fig 1: An illustration of dimension reduction from higher to lower dimension(2D plot) [2].

III. How it works

There are many methods for analyzing the high dimensional datasets using this principle. The three most used techniques are as below:

- ## Principal Component Analysis (PCA)

PCA is a linear method of dimensionality reduction, wherein we combine the various features and create new linear factors called principal components. The top ranked principal components retaining the most accurate information of original data is selected[5]. Features having the highest variance, as represented by covariance matrix, are selected. They are known as eigenvectors.

- ## t-distributed stochastic neighbor embedding (t-SNE)

This is a non-linear method which emphasizes on grouping points with similar nature together, by comparing high dimensional features of all points to correlate the relation between them. Points having more similarity have a higher probability distribution, getting clustered together and vice versa to form a distinct spatial representation[6].

- ## Uniform manifold approximation and projection (UMAP)

UMAP, similar to t-SNE, is another non-linear method which is more robust, and dimensionally reduces the high feature datasets by finding the lower dimension graphical representation that best retains the maximum features as the original resolution. This is done by finding the ideal cost function representing the distance between two points in multi dimensional and dimensionally reduced spaces[7].

We did single cell transcriptomics analysis in RStudio using the Seurat package (Fig 2) where we used all three methods for unsupervised clustering analysis and compared their characteristics[3].

Source Visual Outline

````
```{r}
# for running the functions to reduce dimensions to pca/umap/tsne using the first 10 features(dims)
pbmc <- RunPCA(pbmc, dims = 1:10)
pbmc <- RunUMAP(pbmc, dims = 1:10)
pbmc<- RunTSNE(pbmc, dims = 1:10)
```


```{r}
new.cluster.ids <- c("Naive CD4 T", "CD14+ Mono", "Memory CD4 T", "B", "CD8 T", "FCGR3A+ Mono", "NK", "DC", "Platelet") #for annotating the labels
names(new.cluster.ids) <- levels(pbmc) #assigning the labels to clusters ids
pbmc <- RenameIdents(pbmc, new.cluster.ids) #renaming the clusters
#plotting the pca/umap/tsne plots, with no legends
DimPlot(pbmc, reduction = "pca" , label = TRUE, pt.size = 0.5) + NoLegend()
DimPlot(pbmc, reduction = "umap", label = TRUE, pt.size = 0.5) + NoLegend()
DimPlot(pbmc, reduction = "tsne", label = TRUE, pt.size = 0.5) + NoLegend()
```
````

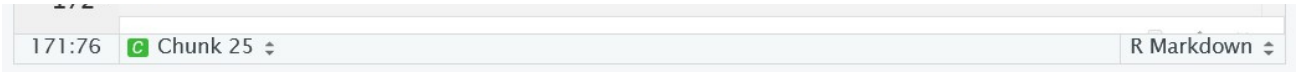

Fig 2: R Script with comments, including all three types of clustering analysis and labeling of cells[3].

In case of PCA (Fig 3), the clusters gets projected linearly, developing a disadvantage when analyzing the nonlinear data like gene expression[5]. But it's more robust nature helps in retaining the maximum important features in the plot.

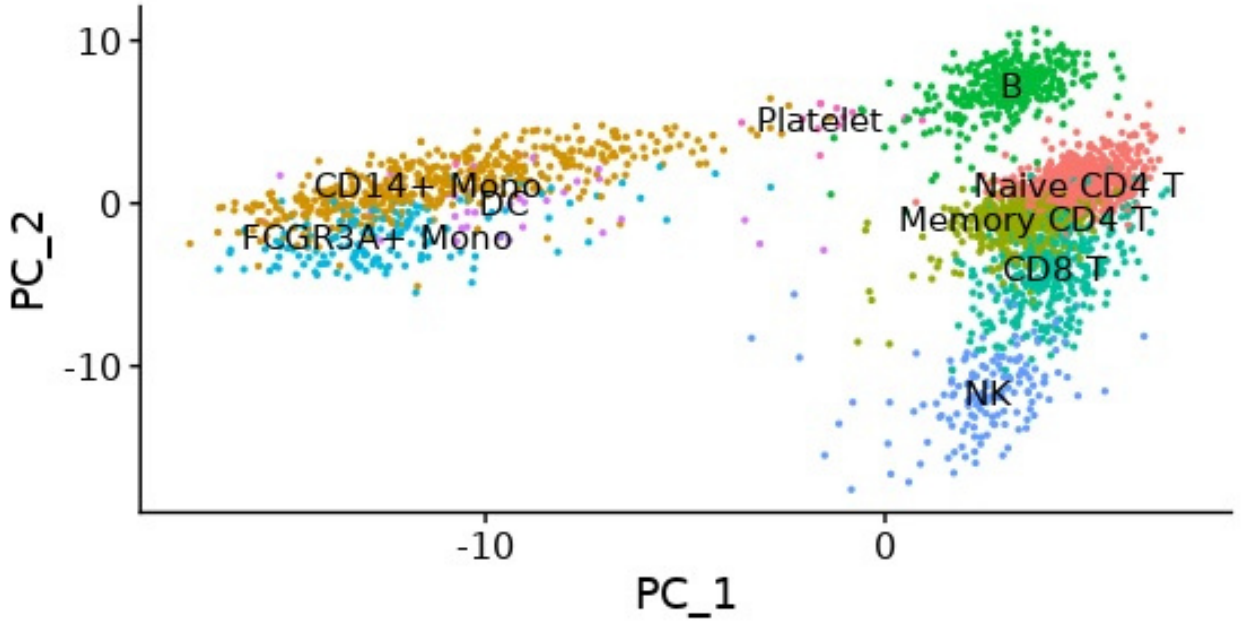

Fig 3: Illustration of cell clusters in dimplot using PCA algorithm[3].

In UMAP (Fig 4), the lower dimension graph closely represents the actual distribution and distance between cells making it ideal for unbalanced or sparse datasets. It does suffer from interpretability issues due to lack of incorporation of spatial positions especially in complex datasets and it's inability to filter the noise due to high sensitivity[7].

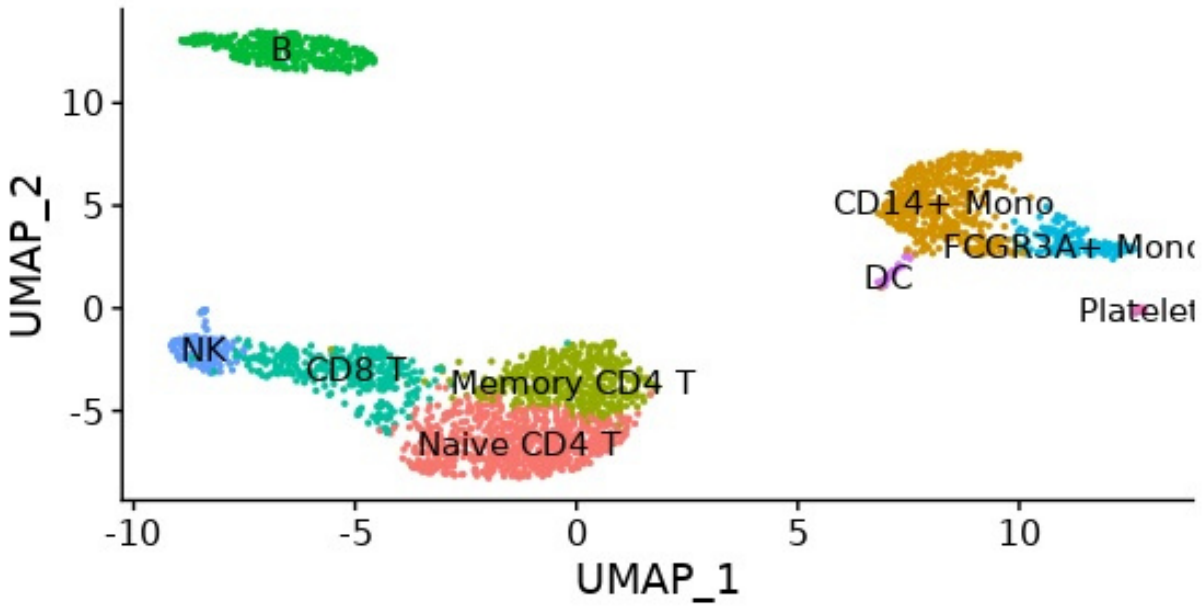

Fig 4: Illustration of cell clusters in dimplot using UMAP algorithm[3].

For t-SNE plots (Fig 5), the clustering is much more uniform and the points are more representative of the structure of the group, leading to better cluster representations. But it is computationally intensive to analyze and there are inaccuracies generated when analyzing large datasets[6].

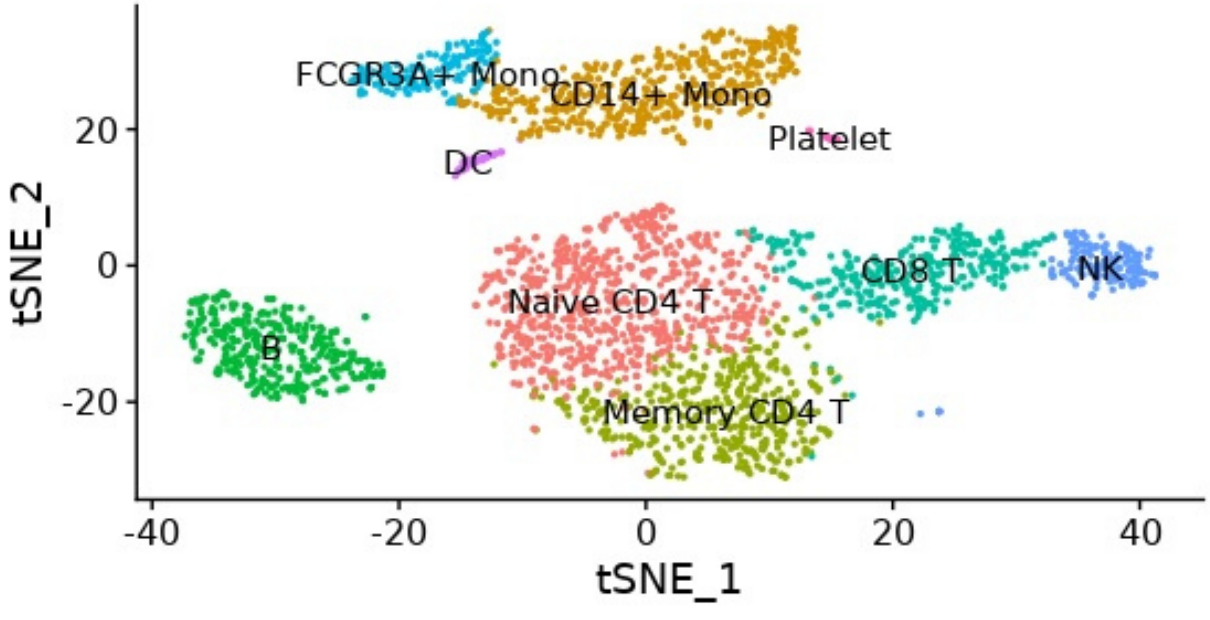

Fig 5: Illustration of cell clusters in dimplot using t-SNE algorithm[3].

In conclusion, these methods made the analysis of data easier, especially when the initial dataset had over 13700 features (Fig 6)!

```
## An object of class Seurat
## 13714 features across 2700 samples within 1 assay
## Active assay: RNA (13714 features, 0 variable features)
```

Fig 6: Secreenshot of the section displaying number of features in the data we analyzed[3].

IV. Advantages and disadvantages

The main advantage of dimensionality reduction over other algorithms is that the data analysis can be computationally efficient and quick, especially for very large features size, sparse and noisy datasets or unequally distributed data. We can also identify significant features, as the features that gets selected are important and redundant ones gets discarded. It reduces all the information into visually interpretative plots for summary[4].

Despite its benefits, it suffers from setbacks, such as data loss due to limitations in selecting the features, leading to dropping some important features[4]. Complex data also makes it difficult for interpretation and analysis.

V. Significance in biological research

It is an important tool having extensive applications in bioinformatics. It is the primary means of analysis of genomic data since tons of genes(i.e features) gets expressed in a single cell, making it vital for analyzing gene expression by feature reduction[8]. This algorithm is useful in branches such as proteomics and metabolomics analysis, and other biological domains like drug discovery, image analysis, etc. where multiple features present in these datasets and sparse data properties aids in initiating downstream analysis.

Dimensionality reduction algorithms have also inspired the development of supervised classifiers, like linear discriminant analysis(LDA) which shares similar properties[9]. Additionally, these clustering models can be integrated with preliminary semi supervised machine learning algorithms for developing primitive elementary modelling tools[10].

References

1. Nguyen, L.H. and Holmes, S. (no date) Ten quick tips for effective dimensionality reduction, PLOS Computational Biology. Public Library of Science. Available at: https://doi.org/10.1371/journal.pcbi.1006907).
2. Ray, S. (2020) Beginners Guide to Learn Dimension Reduction Techniques, Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2015/07/dimension-

reduction-methods/.
3. Seurat - guided clustering tutorial • Seurat. Available at: https://satijalab.org/seurat/articles/pbmc3k_tutorial.html.
4. Hira ZM, Gillies DF. A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data. *Adv Bioinformatics*. 2015;2015:198363. doi:10.1155/2015/198363
5. Migenda N, Möller R, Schenck W. Adaptive dimensionality reduction for neural network-based online principal component analysis. *PLoS One*. 2021;16(3):e0248896. Published 2021 Mar 30. doi:10.1371/journal.pone.0248896
6. Kobak D, Berens P. The art of using t-SNE for single-cell transcriptomics. *Nat Commun*. 2019;10(1):5416. Published 2019 Nov 28. doi:10.1038/s41467-019-13056-x
7. McInnes, L., Healy, J. and Melville, J. (2020) *UMAP: Uniform manifold approximation and projection for dimension reduction*, *arXiv.org*. Available at: https://doi.org/10.48550/arXiv.1802.03426.
8. Sun, S., Zhu, J., Ma, Y. *et al.* Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis. *Genome Biol* **20**, 269 (2019). https://doi.org/10.1186/s13059-019-1898-6
9. Zhao X, Guo J, Nie F, Chen L, Li Z, Zhang H. Joint Principal Component and Discriminant Analysis for Dimensionality Reduction. *IEEE Trans Neural Netw Learn Syst*. 2020;31(2):433-444. doi:10.1109/TNNLS.2019.2904701
10. Liu P, Liu S, Fang Y, et al. Recent Advances in Computer-Assisted Algorithms for Cell Subtype Identification of Cytometry Data. *Front Cell Dev Biol*. 2020;8:234. Published 2020 Apr 28. doi:10.3389/fcell.2020.00234

Word count (excluding title, headings and references): 755 words.