Optimizing Gene Subset Selection for Single-Cell RNA Sequencing Data Clustering Using Genetic Algorithms

Denisse Chacón-Ramírez*, Emilio Rios-Ochoa†,
School of Engineering and Sciences, Tecnologico de Monterrey
Email: *a01562077@tec.mx, †a01378965@tec.mx

***Abstract*—The efficient analysis of single-cell RNA sequencing (scRNA-seq) data is of great importance for the understanding of cellular functional diversity and the development of targeted therapies. In the analysis pipeline, the clustering step is performed with the objective of associating cells according to similarities in their transcriptome. However, high dimensionality, noise and sparse matrices complicate the process. In order to explore different subsets of genes for clustering, we implemented a genetic algorithm (GA) to identify optimal gene subsets that maximize clustering performance in scRNA-seq data, using Normalized Mutual Information (NMI) as the fitness metric. We performed two experiments: providing a subset of marker genes to the GA or not providing it in order to see if it could improve the performance of the GA in finding better subsets for clustering. However, the inclusion of marker genes did not significantly improve performance and, in general, the selected subsets did not show biologically relevant expression patterns, suggesting that the algorithm may have become trapped in local solutions and that further exploration in the solution space is required to obtain more robust and biologically informative results.**

***Index Terms*—scRNA-seq, single-cell, clustering, feature selection, marker genes**

I. Introduction

The advancement of single-cell RNA sequencing (scRNA-seq) technologies has revolutionized our understanding of biological systems by providing detailed information about gene expression in each cell at the individual cell level. By deciphering gene interactions at the cellular level, scRNA-seq provides deep insights into the underlying biological processes [1]. These insights improve our understanding of disease mechanisms, provide greater insight into cellular heterogeneity, and improve understanding of biology in general [2], [3]. However, efficient computational methods are needed to take full advantage of the data obtained from scRNA-seq experiments. Depending on the research questions and the methodology used to obtain the data, the choice of computational tools to analyze it is made [4]. Generally, the analysis of scRNA-seq data involves a number of steps including quality control, normalization, feature selection, clustering, cell type annotations, differential gene expression analysis, and pathway analysis [5].

A critical step in scRNA-seq analysis is the accurate identification of distinct cell types through clustering analysis. During this step the cells are associated by unsupervised grouping into clusters based on similarities in their transcriptome. Then, the clusters are usually manual annotated using marker genes that are differentially-expressed between the different clusters [1]. Consequently, most of the downstream analysis of the data is based on the initial clustering results. For example, one application of scRNA-seq analysis is to sort cells in order to design personalized cancer treatments that target only malignant cells, while avoiding side effects on healthy cells [6]. However, to achieve this, it is necessary to identify cells correctly, and distinguishing between malignant and healthy cells remains a challenge. As reference data, or annotations of cell types, do not exist for all tissues and/or diseases, cell identification must be performed manually for each sample through clustering [1].

However, the clustering step remains challenging due to the special characteristics of scRNA-seq data. First, these kind of data often have high dimensions [7]. Secondly, scRNA-seq data matrices are often sparse [8]. This high dimensionality and sparsity can introduce noise, complicating the identification of cell type.

One possible approach to address this challenge is gene subset selection, which aims to identify the most informative genes for clustering cells into distinct types, usually known as marker genes [9]. Genes used for clustering are typically selected based on their variability, with highly variable genes (HVGs). However, HVGs are not always the most informative for clustering, and conversely, genes that are informative for clustering may not necessarily be classified as HVGs [10]. Thus, the selection of relevant genes not only has the potential to improve computational efficiency by reducing data dimensionality but also provides biological insights by pinpointing genes that are particularly important for distinguishing cell types [11], [12].

This project aims to address this challenge by developing an evolutionary computation approach to optimize gene subset selection for scRNA-seq data clustering. We propose a genetic algorithm (GA) that explores the space of possible gene combinations to identify subsets that maximize clustering accuracy. By iteratively refining the population, the GA aims to converge on a gene subset that maximizes clustering performance. Once the GA has identified high performing gene subsets, we validated the biological relevance of the selected genes, ensuring that they correspond to meaningful biological features.

The rest of the work is structured as follows: Section 2 presents the background of this work. Section 3 the description of the GA algorithm used to explore the search space. Then, Section 4 presents the results from the experiments. Finally, Section 5 presents the conclusions and future work.

II. Background

The analysis of scRNA-seq data provides unprecedented insights into the heterogeneity of cellular states by measuring gene expression at the resolution of individual cells. However, this approach generates high-dimensional, sparse data that poses significant computational challenges, particularly in clustering cells and selecting biologically relevant features. To address these challenges, optimization techniques like genetic algorithms (GAs) offer a promising solution. GAs efficiently explore large search spaces, making them well-suited for tasks such as identifying informative genes and improving clustering accuracy in scRNA-seq analysis pipelines. This section describes the context of the research work.

A. scRNA-seq Data

All the RNA present in a cell is referred as transcriptome [13]. Analyzing the transcriptome provides valuable insights into how a cell's genetic code (its DNA) is translated into functional behavior. When genes are activated, they produce messenger RNA (mRNA), which then directs the synthesis of proteins. By studying the transcriptome, we can better understand cellular responses to various signals, how cells operate, how they develop, and how diseases manifest [14].

While all the cells in an organism have the same genetic material (DNA), they can behave differently because they activate different sets of genes depending on their specific conditions, leading to variations in their transcriptomes [13]. To visualize this, imagine an orchestra where the genome is the sheet music (the same for all cells), but each cell creates its own "sound" based on which genes are active at the time. Since each cell has a unique transcriptome, studying the transcriptome at the level of individual cells becomes crucial.

scRNA-seq enables the measurement of gene expression at the level of individual cells [2]. Unlike bulk RNA sequencing, which measures the average gene expression across a wide population of cells, scRNA-seq provides insight into cellular states, cell populations, and gene expression patterns that would be otherwise masked in bulk RNA analysis [15]. Over the past decade, scRNA-seq has become a revolutionary tool in omics analysis, with applications ranging from characterizing tissue heterogeneity to studying cell dynamics. Thus, being invaluable for providing a better understanding of diseases such as cancer, autoimmune conditions and neurological disorders, where cellular diversity plays a crucial role in disease progression and response to treatment [16].

In general, the analysis of scRNA-seq data involves three main steps: 1) data preprocessing and visualization, 2) general analysis, and 3 exploratory analysis. Data preprocessing step involves removing noise through quality control (eliminating duplicates and low-quality readings), normalization (making gene expression profiles comparable across cells), and feature selection (focusing on biologically relevant genes) [17]. On the other hand, the general analysis includes clustering cells with similar expression profiles to identify cell populations and annotating these clusters as cell types. Finally, the exploratory analysis focuses on specific questions which could have two different approaches: cellular-level analysis that looks at changes like cell differentiation, which is tracked through cell-trajectory inference, and gene-level analysis that identifies specific genes that are expressed under certain conditions.

Nevertheless, despite its transformative potential, the analysis of scRNA-seq data is challenging due to several factors. These include the high dimensionality of the data (with tens of thousands of genes measured in each cell) and the sparsity of the data (many genes have low or zero expression across most cells) [18]. Specifically, one of the most challenging steps and most affected by these challenges is clustering since many of the downstream analysis steps depend on the annotations that are generated for the cells [8].

To address the challenges posed by high dimensionality and sparsity, one approach is to select a subset of genes that are most informative for clustering. Reducing the number of genes used in clustering not only improves computational efficiency, but also helps focus on genes that are most relevant for distinguishing between different cell types [11]. In traditional scRNA-seq analysis pipelines, one common approach is to select highly variable genes (HVGs) as features for clustering [10]. These are genes whose expression varies significantly across cells, as they are assumed to contain the most information about cell type differentiation. However, while HVGs often capture variation that is useful for clustering, they are not always the best choice [10].

B. Genetic Algorithms

Evolutionary algorithms (EAs) are optimization techniques inspired by the process of natural selection. Genetic algorithms (GAs), a subset of EAs, are particularly well suited for solving optimization problems where the search space is large and complex. GAs operate by iteratively refining a population of potential solutions through different operators such as selection, crossover, and mutation. The aim is to evolve a population of solutions that improves over generations, eventually converging on a near optimal solution [19].

In general, genetic algorithms initially generate a random population of individuals or solutions. This population is then subjected to the crossover operator, where two individuals are selected to exchange some values to create new solutions. Then, the mutation operator is applied to explore the search space. And finally, the selection operator is applied where, according to the fitness with the objective function, the individuals that will be part of the new generation are chosen. This process is repeated iteratively [20].

III. Algorithmic Approach

Leveraging the exploration and self-improvement capabilities of GAs, we explored the space of possible combinations

to find the one that maximized clustering performance. We implemented a binary-encoded GA, where each individual in the population is a binary string representing the subset of genes. In such representation, a bit of value 0 means the exclusion of the corresponding gene from the complete pool of genes, whereas 1 refers to its inclusion in the subset, as seen in Figure 1. Using this scheme has the advantage that all individuals have the same and constant length in their genotype, simplifying the evolutionary operators.

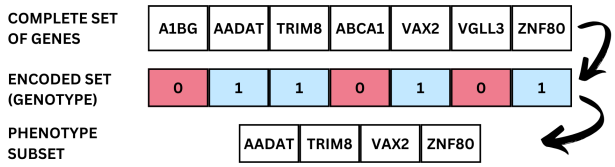

Fig. 1. Example of the individual representation of the genes to cluster.

As for the evolutionary operators, we implemented binary tournament selection, where in an iterative process, two individuals are sampled at random from the population and the best one based on its fitness is selected as parent. Also, we implemented multi-point crossover and multiple bit mutation, in order for operators to have a greater effect, since the individuals' size was over 19,000 bits. Moreover, we followed a $\mu+\lambda$ approach to enforce elitism and guarantee that only the best individuals were kept, since in this problem, exploration was more important than exploitation.

Our GA's implementation in Algorithm 1 requires multiple hyperparameters. Most of these, are standard across GA implementations, such as crossover crossover points, population size, maximum generations and mutation/crossover probabilities. However, we equiped our model with two additional parameters to enhance its robustness: markerGenes and dataset. The dataset is required to compute the fitness based on clustering, while the markerGenes is a collection of genes known to be expressed more in some cell types than in others, allegedly serving as markers in cell identification. Our model can either take into account these marker genes by default in the clustering or treat them as part of the individual, allowing them not to be used, enabling us to evaluating whether including them all and/or always is advantageous.

A. Mathematical Modeling of the Problem

The goal of the genetic algorithm is to select a subset of genes that maximizes the clustering performance, as measured by the Normalized Mutual Information (NMI) score. To achieve this, we set the problem as a combinatorial optimization problem, since the decision variable involves selecting a subset of genes from a finite set. Let $X \in \mathbb{R}^{n\times m}$ be the scRNA-seq dataset, where n is the number of cells and m is the number of features (genes). The objective is to identify the subset of genes $S \subseteq \{1, 2, \ldots, m\}$ that maximizes the alignment between the clustering result and the known cell annotations. The search space of this problem is exponential (2^m), where m is the number of genes in the dataset.

Algorithm 1 Genetic Algorithm Implementation

Require: maxGen, popSize, markerGenes, dataset, bitsToMutate, crossPoints, crossProb, mutProb
Ensure: bestIndividuals, bestFitness
  bestIndividuals ← []
  bestFitness ← []
  population ← Random(size=popSize, choice=[0, 1])
  for gen **in** [1, .., maxGen] **do**
    fitness ← **Evaluation**(population)
    parents ← **Selection**(population, fitness)
    offspring ← **Crossover**(parents)

    fitnessOffspring ← **Evaluation**(offspring)
    fitnessAll ← fitness + fitnessOffspring
    populationAll ← population + offspring

    newPopulation ← **Sort**(fitnessAll, populationAll)
    population ← **Mutation**(newPopulation[popSize:])

    bestIndividual ← **Max**(fitness)
    bestFitness ← fitness[bestIndividual]
    bestIndividuals ← **Decode**(population[bestIndividual])

    if gen % 10 == 0 **then**
      if STD(bestSolutions[gen-10:]) < 0.01 **then**
        return bestIndividuals, bestFitness
      end if
    end if
  end for
  return bestIndividuals, bestFitness

The fitness evaluation for each individual in the population is based on clustering, with the NMI serving as the objective function:

$$\max_{S\subseteq\{1,2,\ldots,m\}} \text{NMI}(S)$$

The NMI between two clusterings U and V is calculated as:

$$\text{NMI}(U,V) = \frac{2\times I(U;V)}{H(U)+H(V)}$$

where $I(U;V)$ is the mutual information between U and V, calculated as:

$$I(U;V) = \sum_{u\in U}\sum_{v\in V} p(u,v)\log\frac{p(u,v)}{p(u)p(v)}$$

and $H(U)$ and $H(V)$ are the entropies of U and V, respectively:

$$H(U) = -\sum_{u\in U} p(u)\log p(u), \quad H(V) = -\sum_{v\in V} p(v)\log p(v)$$

Clustering was performed using the shared nearest neighbor (SNN) graph-based method, implemented with a resolution

parameter of 0.5 in the Seurat package in R (version 5.0.3) [10], as it has been shown to provide high performance when clustering scRNA-seq data. Prior to clustering, a Principal Component Analysis (PCA) was applied for dimensionality reduction, and the ElbowPlot function was used to determine the optimal number of principal components.

B. Data

We used the Muraro pancreas scRNA-seq dataset [21]. The preprocessing of the dataset was carried out using the *scater* R package [22]. First, the cells with less than 200 genes detected were eliminated. Also, cells with an ERCC percentage higher than 10% were filtered out. Genes expressed in fewer than two cells were discarded. Finally, the expression count matrix was normalized to a log-transformed count matrix. After preprocessing, 2308 cells and 19046 genes were obtained. Cell annotations, which served as ground truth for the clustering analysis, were taken directly from the original publication by the authors [21].

Additionally, the marker gene set was constructed using two databases: CellMarker database [23] and PanglaoDB database [24]. All official gene symbols of all cell subtypes corresponding to the pancreas in both human and mouse were included.

IV. Experimental Results

To evaluate how different sets of genes influenced clustering performance, we ran the genetic algorithm in five independent runs, using a population of 20 individuals and a maximum of 30 generations. Two experiments were conducted: the first without providing a specific set of marker genes as input, and the second using the previously described set of marker genes. Additionally, we compared the results obtained by the genetic algorithm with the NMI scores from clustering based on the full set of genes and using the 2000 top expresed genes.

The genetic algorithm was configured with the following parameters: a crossover rate $P_c = 0.8$, indicating the probability of a pair of individuals to create offspring; a mutation rate $P_m = 0.1$, specifying the likelihood of an individual undergoing mutation; 1,000 mutation points per individual $M_p = 1000$, determining the number of bits flipped during mutation; and four crossover points $C_p = 4$ for the recombination of parent individuals. The parameters were defined empirically after running some experiments.

A. Convergence of the Genetic Algorithm

Convergence plots were generated for both experiments (runs with and without marker genes). In addition, to visualize the clustering results, we applied Uniform Manifold Approximation and Projection (UMAP) for dimensionality reduction, reducing the high-dimensional gene expression data into two dimensions. The UMAP projections were colored by both the clusters identified by the genetic algorithm and the true labels of the cell types provided by the authors [21].

1) Performance without Marker Genes: Figure 2 (left) shows the score obtained by the best solution found by the algorithm in each generation. In all cases, the algorithm stopped after reaching the stop criterion set at 10 generations. The best NMI values obtained per generation were in the range of 0.76 to 0.78. This lack of convergence to a single solution suggests that the algorithm did not reach a global optimum, but rather identified multiple subsets of genes that locally improves the clustering metric.

It is important to note that the 5 runs converged to different solutions, although all outperformed the performance of the clustering performed with the full set of genes, which obtained an NMI of 0.66. However, the performance of all subsets is slightly lower than when only the most expressed genes are used (NMI = 0.87). The results also reveal that the subsets of genes with the best clustering performance contain between 2,750 and 3,500 genes, representing less than 20% of the total genes in the dataset (Figure 2 right). This finding highlights the fact that the inclusion of not informative genes could worsen the clustering performance.

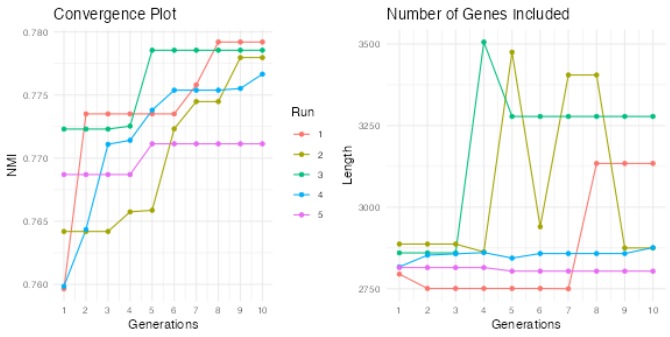

Fig. 2. Convergence plot and the number of genes included across generations for different runs of the genetic algorithm, when no set of marker genes was provided as input.

As an example, we performed UMAP visualization of the gene subset that gave the highest NMI score (Figure 3). It is observed that the clustering algorithm is subdividing some groups of cells, such as alpha cells while is grouping different cell types into the same clusters, as it is the case of delta and pancreatic polypeptide (pp) cells being grouped into cluster 3. However, it seems that the other cell types were clustered correctly.

2) Performance with Marker Genes Included: In contrast to the experiments where no predefined set of marker genes was provided as input to the algorithm, the runs with marker genes showed distinct convergence behaviors. Out of the five runs, two completed all 30 generations, one satisfied the stopping criterion at 20 generations, and two converged at 10 generations (Figure 4). The highest NMI values observed per generation ranged between 0.71 and 0.778, corresponding to subset lengths varying between 6,000 and 12,000 genes. Notably, in Run 4, the size of the gene subset increased steadily, but this was accompanied by a decline in the NMI score. Conversely, Runs 1 and 3 exhibited the lowest NMI values and

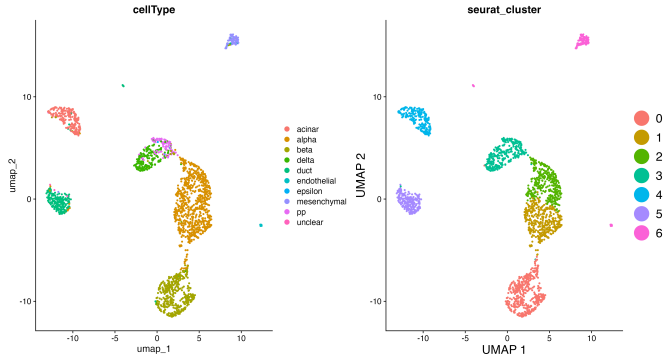

Fig. 3. UMAP plots showing Seurat clustering results for the subset that gave the best NMI score (0.779) with cells colored by true labels and marked by assigned clusters.

the smallest gene subsets, suggesting that the inclusion of non-informative genes negatively impacts clustering performance.

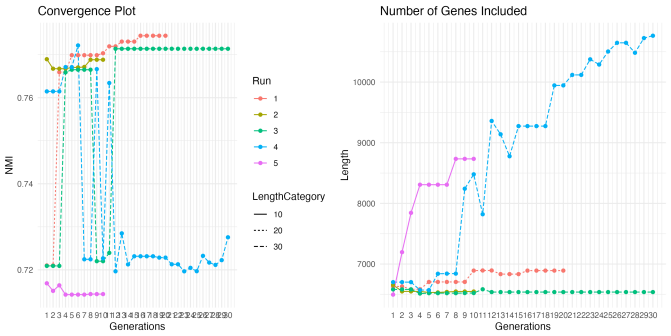

Fig. 4. Convergence plot and the number of genes included across generations for different runs of the genetic algorithm, when a set of marker genes was provided as input.

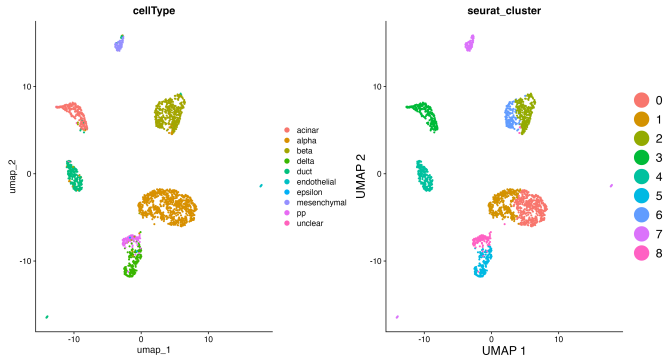

Fig. 5. UMAP plots showing Seurat clustering results for the subset that gave the best NMI score (0.774) for the experiments where a set of marker genes was provided, with cells colored by true labels (left) and by assigned clusters (right).

Similar to the previous experiment, we performed a UMAP analysis for the best score obtained in the experiments with marker genes. The results show that the best score achieved in the five experiments is very similar to that obtained in the case without a set of marker genes. In addition, the UMAP analysis reveals that the clustering algorithm subdivides some groups (Figure 5). For example, in this case, beta cells are divided into two distinct clusters, while alpha cells are also grouped into two separate clusters. The other cell types appear to have clustered appropriately.

B. Impact of Marker Genes on Clustering Performance

To assess whether providing the algorithm with a set of marker genes significantly influences the subsets of genes identified to improve clustering, we performed a Wilcoxon signed-rank test. This comparison focused on the best solutions obtained in each run of the algorithm, considering both experiments in which marker genes were included and those in which they were not provided.

Figure 6 shows the distribution of the best results for each of the 5 runs per experiment, visualized through a box plot. At first glance, it appears that no giving a set of marker genes leads to higher NMI values compared to when the algorithm is given the set of marker genes. However, the$p - value$ obtained after the test was 0.0555, suggesting that there is not enough evidence to reject the null hypothesis that there is no significant difference between the results of the GA with marker genes or without (i.e., that they are equal). However, since this $p-value$ is very close to 0.05, it is possible that the lack of evidence is due to the small size of the observations. To increase the reliability of the test, it would be necessary to run both experiments independently a greater number of times and check whether the null hypothesis continues to be not rejected.

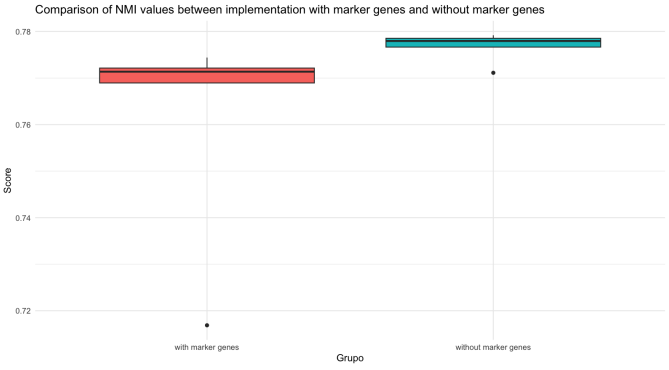

Fig. 6. Convergence plot and the number of genes included across generations for different runs of the genetic algorithm, when a set of marker genes was provided as input.

C. Selected Gene Subsets and Biological Relevance

To analyze the biological relevance of the genes selected in the subsets that got the highest NMI per generation (excluding that genes that were already classified as marker genes). To achieve this, we calculated how often each gene was selected across the best solutions and we selected the six more repeated genes. Then, we investigate their biological relevance.

The six most frequently repeated genes in the best subsets across generations were RNH1, PGA5, MALAT1, NDN,

MAP3K4, and DHX8. To analyze their biological relevance we decided to take as a study case the subset that got the best NMI score (which is shown in Figure 3. As shown in Figure 7, most of these genes are expressed at similar levels across the clusters. This suggests that, despite appearing in all the subsets with the highest NMI scores, these genes may not provide meaningful information for clustering. Only PGA5 exhibits a distinct expression pattern, predominantly in cluster 4. According to Figure 3, this cluster likely corresponds to acinar cells.

PGA5 encodes a protein with a significant role in regulating the function and development of acinar cells, primarily through its involvement in transcriptional control and signaling pathways [25]. This protein is essential for maintaining the specialized functions of pancreatic acinar cells, which produce digestive enzymes. It influences the transcriptional programs necessary for initiating and completing acinar cell differentiation, ensuring these cells develop the machinery required for enzyme production [26]. This aligns with the observation that PGA5 expression is limited to a small number of cells in cluster 4.

On the other hand, the inclusion of the other five genes (RNH1, MALAT1, NDN, MAP3K4, and DHX8), which exhibit similar expression in all clusters, in the best subsets leads us to hypothesize that the search space explored by the algorithm might be too limited. For example, in the case where no marker genes were included, the fact that the algorithm converges rapidly suggests that it may be getting stuck on multiple subsets of genes that produce comparable clustering performance, without exploring more diverse or informative solutions.

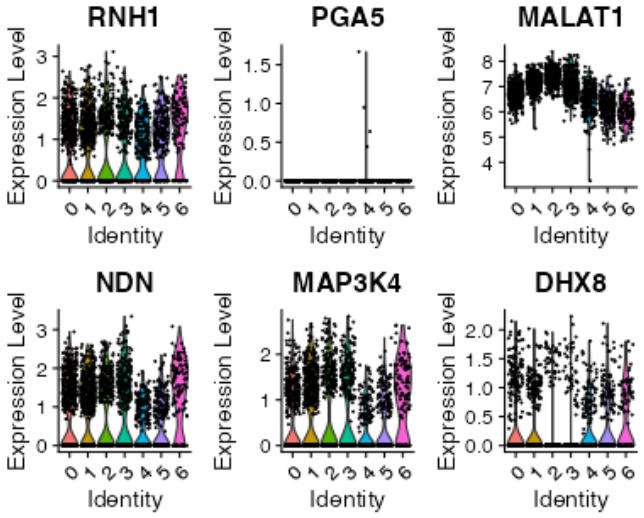

Fig. 7. Violin plot showing the gene expression of the most repeated genes across clustering results for the subset that gave the best NMI score (0.779) without giving marker genes as an input.

V. Conclusions and Future Work

This work uses evolutionary computation, specifically genetic algorithms (GAs), to address complex bioinformatics problems such as gene selection for scRNA-seq clustering. Our experiments showed that, without predefined marker genes, the genetic algorithm was able to identify subsets of genes that outperformed clustering based on the full set of genes. However, the algorithm struggled to converge to a global optimum, often settling on multiple locally optimal solutions. The inclusion of marker genes did not significantly improve clustering performance, as indicated by the statistical analysis, which suggested no significant difference between the results of runs with and without marker genes.

The clustering performance was found to be sensitive to the size and composition of the gene subsets. In both experiments, smaller subsets, tended to achieve better results compared to larger subsets. This suggests that the inclusion of non-informative genes can negatively impact clustering. UMAP visualizations highlighted that while the algorithm successfully clustered most cell types, certain cell populations were subdivided into multiple clusters.

However, the application of GAs to scRNA-seq analysis has some limitations. Due to computational and time constraints, we were able to run the algorithm only five independent times per experiment, which may obscure important insights. A broader range of runs could help capture more robust patterns and validate the consistency of the results, mitigating the impact of randomness inherent in evolutionary computation. However, it is important to keep in mind that the search space of our problem is exponential, 2^{19046} specifically for the dataset used in this study, so exploring all solutions is almost impossible in a short period of time. Consequently, the portion of the search space explored by the algorithm was relatively small compared to its full extent

To address this limitations, in the future we propose combining GAs with complementary techniques, such as deep learning or swarm intelligence, to enhance their efficiency and precision in exploring the vast gene space. Additionally, incorporating multi-omics data could help to create a holistic view of cellular behavior, revealing connections across layers of biological information, thus improving cell type classification. In addition, we propose to combine the genetic algorithm strategy with mathematical methods for gene selection. For example, selecting those genes that have a higher correlation, either negative or positive, since it has been observed that genes that are specific to the same cell type tend to be highly correlated with each other, whereas genes specific to different cell types are more likely to be anti-correlated. This relationship suggests that correlation may be a useful indicator for identifying genes that are relevant for differentiation between cell types [27]. After selecting those genes with high correlations, an evolutionary algorithm could be applied to explore a smaller search space.

REFERENCES

[1] A. Ianevski, A. K. Giri, and T. Aittokallio, "Fully-automated and ultra-fast cell-type identification using specific marker combinations from single-cell transcriptomic data," *Nature Communications*, vol. 13, Mar. 2022.

[2] L. Heumos, A. C. Schaar, C. Lance, A. Litinetskaya, F. Drost, L. Zappia, M. D. Lücken, D. C. Strobl, J. Henao, F. Curion, H. Aliee, M. Ansari, P. B. i Mompel, M. Büttner, E. Dann, D. Dimitrov, L. Dony, A. Frishberg, D. He, S. Hediyeh-zadeh, L. Hetzel, I. L. Ibarra, M. G. Jones, M. Lotfollahi, L. D. Martens, C. L. Müller, M. Nitzan, J. Ostner, G. Palla, R. Patro, Z. Piran, C. Ramírez-Suástegui, J. Saez-Rodriguez, H. Sarkar, B. Schubert, L. Sikkema, A. Srivastava, J. Tanevski, I. Virshup, P. Weiler, H. B. Schiller, and F. J. T. and, "Best practices for single-cell analysis across modalities," *Nature Reviews Genetics*, vol. 24, pp. 550–572, Mar. 2023.

[3] B. DeMeo and B. Berger, "SCA: recovering single-cell heterogeneity through information-based dimensionality reduction," *Genome Biology*, vol. 24, Aug. 2023.

[4] P. V. Kharchenko, "The triumphs and limitations of computational methods for scRNA-seq," *Nature Methods*, vol. 18, pp. 723–732, June 2021.

[5] K. Li, Y. H. Sun, Z. Ouyang, S. Negi, Z. Gao, J. Zhu, W. Wang, Y. Chen, S. Piya, W. Hu, M. I. Zavodszky, H. Yalamanchili, S. Cao, A. Gehrke, M. Sheehan, D. Huh, F. Casey, X. Zhang, and B. Zhang, "scrnasequest: an ecosystem of scrna-seq analysis, visualization, and publishing," *BMC Genomics*, vol. 24, May 2023.

[6] W. Zhao, A. Dovas, E. F. Spinazzi, H. M. Levitin, M. A. Banu, P. Upadhyayula, T. Sudhakar, T. Marie, M. L. Otten, M. B. Sisti, J. N. Bruce, P. Canoll, and P. A. Sims, "Deconvolution of cell type-specific drug responses in human tumor tissue with single-cell rna-seq," *Genome Medicine*, vol. 13, May 2021.

[7] V. Y. Kiselev, T. S. Andrews, and M. Hemberg, "Publisher correction: Challenges in unsupervised clustering of single-cell rna-seq data," *Nature Reviews Genetics*, vol. 20, p. 310–310, Jan. 2019.

[8] Y. Qiu, L. Yang, H. Jiang, and Q. Zou, "sctpc: a novel semisupervised deep clustering model for scrna-seq data," *Bioinformatics*, vol. 40, Apr. 2024.

[9] Z. Chen, C. Wang, S. Huang, Y. Shi, and R. Xi, "Directly selecting cell-type marker genes for single-cell clustering analyses," *Cell Reports Methods*, vol. 4, p. 100810, July 2024.

[10] A. Butler, P. Hoffman, P. Smibert, E. Papalexi, and R. Satija, "Integrating single-cell transcriptomic data across different conditions, technologies, and species," *Nature Biotechnology*, vol. 36, p. 411–420, Apr. 2018.

[11] S. R. Tyler, D. Lozano-Ojalvo, E. Guccione, and E. E. Schadt, "Anti-correlated feature selection prevents false discovery of subpopulations in scrnaseq," *Nature Communications*, vol. 15, Jan. 2024.

[12] G. Y. L. Ng, S. C. Tan, and C. S. Ong, "On the use of qde-svm for gene feature selection and cell type classification from scrna-seq data," *PLOS ONE*, vol. 18, p. e0292961, Oct. 2023.

[13] F. Tang, K. Lao, and M. A. Surani, "Development and applications of single-cell transcriptome analysis," *Nature Methods*, vol. 8, pp. S6–S11, Mar. 2011.

[14] R. Hrdlickova, M. Toloue, and B. Tian, "scpRNA/scp-seq methods for transcriptome analysis," *WIREs RNA*, vol. 8, May 2016.

[15] K.-L. Tiong, D. Luzhbin, and C.-H. Yeang, "Assessing transcriptomic heterogeneity of single-cell rnaseq data by bulk-level gene expression data," *BMC Bioinformatics*, vol. 25, June 2024.

[16] A. A. Khozyainova, A. A. Valyaeva, M. S. Arbatsky, S. V. Isaev, P. S. Iamshchikov, E. V. Volchkov, M. S. Sabirov, V. R. Zainullina, and V. I. Chechekhin, "Opportunities of complex analysis in single-cell rna sequencing," *Biohimiâ*, vol. 88, no. 2, pp. 171–198, 2023.

[17] D. Jovic, X. Liang, H. Zeng, L. Lin, F. Xu, and Y. Luo, "Single-cell RNA sequencing technologies and applications: A brief overview," *Clinical and Translational Medicine*, vol. 12, Mar. 2022.

[18] O. Stegle, S. A. Teichmann, and J. C. Marioni, "Computational and analytical challenges in single-cell transcriptomics," *Nature Reviews Genetics*, vol. 16, pp. 133–145, Jan. 2015.

[19] W. H. Hsu, "Evolutionary computation and genetic algorithms," in *Encyclopedia of Data Warehousing and Mining*, 2009.

[20] K. Deb, "Introduction to genetic algorithms for engineering optimization," 2004.

[21] M. J. Muraro, G. Dharmadhikari, D. Grün, N. Groen, T. Dielen, E. Jansen, L. van Gurp, M. A. Engelse, F. Carlotti, E. J. de Koning, and A. van Oudenaarden, "A single-cell transcriptome atlas of the human pancreas," *Cell Systems*, vol. 3, pp. 385–394.e3, Oct. 2016.

[22] D. J. McCarthy, K. R. Campbell, A. T. L. Lun, and Q. F. Wills, "Scater: pre-processing, quality control, normalization and visualization of single-cell rna-seq data in r," *Bioinformatics*, vol. 33, p. 1179–1186, Jan. 2017.

[23] C. Hu, T. Li, Y. Xu, X. Zhang, F. Li, J. Bai, J. Chen, W. Jiang, K. Yang, Q. Ou, X. Li, P. Wang, and Y. Zhang, "Cellmarker 2.0: an updated database of manually curated cell markers in human/mouse and web tools based on scrna-seq data," *Nucleic Acids Research*, vol. 51, p. D870–D876, Oct. 2022.

[24] O. Franzén, L.-M. Gan, and J. L. Björkegren, "Panglaodb: a web server for exploration of mouse and human single-cell rna sequencing data," *Database*, vol. 2019, p. baz046, 2019.

[25] R. J. MacDonald, G. H. Swift, and F. X. Real, *Transcriptional Control of Acinar Development and Homeostasis*, p. 1–40. Elsevier, 2010.

[26] J. A. Williams, "Regulation of acinar cell function in the pancreas," *Current Opinion in Gastroenterology*, vol. 26, p. 478–483, Sept. 2010.

[27] B. Ranjan, W. Sun, J. Park, K. Mishra, F. Schmidt, R. Xie, F. Alipour, V. Singhal, I. Joanito, M. A. Honardoost, J. M. Y. Yong, E. T. Koh, K. P. Leong, N. A. Rayan, M. G. L. Lim, and S. Prabhakar, "Dubstepr is a scalable correlation-based feature selection method for accurately clustering single-cell data," *Nature Communications*, vol. 12, Oct. 2021.