**CELL TYPE CLASSIFICATION**

**Group Members:**

Anusha Malapaka (45862937), Emma Frost (45966778), Isabelle McGrath (44839594), Di Xiao (45569722), Jiayong Kuang (44872452)

Roles:

Jiayong: Design and implementation of program code for the SVM and Random forest model; Process the data.Editing

Di: Design and implementation of program code for the Seurat model; Process the data, Editing

Isabelle: Explanation of t-SNE and SVM, statistical comparison of cell-type classification algorithms, contribution to the introduction, editing of report

Emma: Explanation of Random Forest, input for discussion, editing of report

Anusha: Group leader, Introduction, Explanation of Seurat, Discussion (compiled with inputs from everyone), Editing

**Goals**

This project aimed to perform cell-type classification of known cell type data sets with three different techniques: Seurat, RandomForest and support vector machine with t-SNE. The process of classification were compared between methods, along with the effectiveness of the method for a data set of RNA expression profiles for 909 cells. A subsection to the goals was also to attempt to compare different aspects of the techniques to the known benchmarks of single-cell sequencing.

**Introduction**

**Single-cell classification**

The ability to study a heterogeneous cell population has substantially improved with single-cell RNA sequencing (scRNA-seq). scRNA-seq allows sequencing of the transcriptome of individual cells, which then can be classified into certain cell types based on this transcript profile. Not only can scRNA-seq delineate the various cell types making up a sample, but it has opened doors to the knowledge of previously unknown cell types and their functions. With the ever-increasing development of scRNA-seq tools, it is becoming increasingly challenging to select the most suitable approach to analyse data. Initially, cell type classification was done manually wherein the unsupervised clustering of transcriptionally similar cells was performed, followed by group annotation. This process was time consuming, laborious, non-reproducible between research groups and sometimes inaccurate. Therefore, automated tools to perform cell-type classification are needed (1). There are a multitude of techniques available (2). A good classifier overcomes the high levels of technical noise associated with scRNA data, is suitable for large datasets, can distinguish between closely-related cell types, is able to perform quickly while consuming minimal processing space (3).

**Seurat**

Seurat is an open-source R toolkit for single-cell genomics; the idea behind this tool is to implement a diverse integration strategy to compare data sets of scRNA-seq across different conditions, technologies or species. This tool is a good usage for broad data sets with more than 5000 genes. The idea was to test the tool in an instance with a small data set. Seurat provides us with a wide range of features to classify cell types and listed below is the method followed (Figure 1) (4).

****

Figure 1. Seurat Methodology

**t-SNE and SVM**

t-distributed Stochastic Neighbour Embedding (t-SNE) is an unsupervised, non-linear dimensionality reduction machine learning algorithm in which high-dimensional data is reduced to two or three dimensions. t-SNE is more suitable for high-dimensional data than classical dimensionality reduction techniques such as Principal Components Analysis (PCA) and multidimensional scaling (MDS), which are linear and therefore struggles to model the low-dimensional representations of similar points as close together. It improves upon SNE, which suffers from difficulty in optimization and has a tendency for points to crowd. In t-SNE, first the high-dimensional pairwise Euclidean distances between data points are converted to a similarity matrix, in which the similarity is defined as the conditional probability that point would choose point as its neighbour, if neighbours were selected in proportion to their probability density under a Gaussian centered at . Subsequently, t-SNE calculates the pairwise similarities in a low dimensional space using a t-distribution, then attempts to minimize divergence between these similarities using a gradient descent method of the Kullback-Leibler divergence (5).

Support vector machine (SVM) is a supervised learning algorithm used for classification and interpreting t-SNE output. SVM involves construction of a hyperplane representing the greatest margin between the groups. When data is not linearly separable in its dimensions, a kernel trick is used to map the data into a higher-dimensional space in which a linear separation is possible. The support vectors are found analytically as a convex optimization problem. Once the training is complete, SVM requires only the support vectors for predictions. When there are multiple classes, strategies such as one-versus-the-rest, pairwise classification and the multi classification formulation can be used. For one-versus-the-rest, the most frequently used multi classification algorithm, each support vector distinguishes each class from the rest. (6)

**Random Forest**

Random forest is an ensemble classifier used in cell type classification model by using multiple independent decision trees to assign a class label to each cell (7,8). Each decision tree is trained on a subset of the training data. The independent decision trees are then applied to the test data, and the final classification is determined from the mode (7). Compared to single classifiers, the accuracy of the method is improved by using multiple classifiers and consolidating the outcomes and are unlikely to overfit (7-9).

The classifier trees can be trained using bagging or boosting. For bagging, the decision trees are trained on a randomly selected subset of the data (10). This approach has shown to make the method less sensitive to noise in the training set (11). For boosting, iterative training of the trees is done on the entirety of the training data (12). This approach is more accurate than bagging and has been shown to reduce classification variance and bias (13, 14). The number of generated trees and variables must be decided. The usual standard is to make 500 trees and the number of variables being the square root of input variables (12). The accuracy of the model is more sensitive to the number variables than the number of trees (15, 16).

**Methods**

**Data pre-processing and normalization**

sc\_CELseq2\_5cl\_p1, sc\_CELseq2\_5cl\_p2 and sc\_CELseq2\_5cl\_p3 (obtained from <https://github.com/LuyiTian/sc_mixology/tree/master/data/csv>) were merged into a single dataset to achieve 909 cells with gene expression profile for 12653 genes. First, the Seurat objects were set up using the function CreateSeuratObject with parameters to filter genes detected in <3 cells and filter cells with <200 gene reading. A global-scaling normalization method “Log Normalization” was applied using the function NormalizeData, which normalizes the feature expression measurements for each cell by the total expression, multiplies this by a scale factor (10,000 by default) (ScaleData function), and log-transforms the result. The normalized dataset was split randomly into training data (80%) and test data (20%). The splitting of the data was repeated to achieve three different pairs of test and training data to be used for analysis in all three techniques.

**Seurat**

*Finding differentially expressed features (cluster biomarkers)*

PCA (RunPCA function) was performed on the training data, which applied a graph-based clustering approach. SLM was applied next to iteratively group cells together to optimize the standard modularity function (FindNeighbour and FindClusters functions). The FindClusters function uses the algorithm Shared-Nearest Neighbour. Non-linear dimensional reduction (UMAP)(RunUMAP function) was used to visualize the clusters. Biomarkers were found for every group compared to all remaining cells via differential expression (FindAllMarkers function). The top biomarker was matched to the known cell types.

*Cell type prediction*

PCA, SLM, and UMAP were performed on the testing data. Biomarkers were used to identify each cluster of testing data. Confusion matrix statistics were applied to measure the prediction accuracy.

**Random Forest**

A Random Forest model was built for each training set. For each Random Forest model generated by different training sets, we used its corresponding testing set to verify the performance of the model and generate a confusion matrix. Thus, by comparing performances of these three Random Forest models, we can determine whether there is any sampling bias for the generated models.

**t-SNE and SVM:**

For the SVM classification, we firstly implemented a dimensionality reduction (t-SNE) for the whole merged dataset (n=909 cells) and visualized it on a two-dimensional graph grouped by dataset ID and cell type. This step is to verify whether cells in different data sets are clustered together, thus affecting our classification of cell types. After that, for the t-SNE result, we separated it into the same three training sets and testing sets based on previous sampling. Finally, using each training sets, we built an SVM model to classify different cell types respectively, and verified its performance on its corresponding testing set.

**Comparison of Algorithm Accuracy**

A statistic called the F measure was calculated using the formula below**.** This statistic was used to provide a balanced measurement of the confusion matrix values embedded into a single value for the model.

In order to determine if one classification algorithm performed better in terms of F measure than another, a one-way ANOVA with Tukey’s correction was performed. Data points compared between algorithms were the mean F measure for the three replicates within each cell type. Significance was when p<0.05.

**Results**

While the analysis was performed on three runs of random data, only the results of the first run of the data set are provided here. However, the confusion matrices of the second and third repetition were considered in the evaluation of the algorithm.

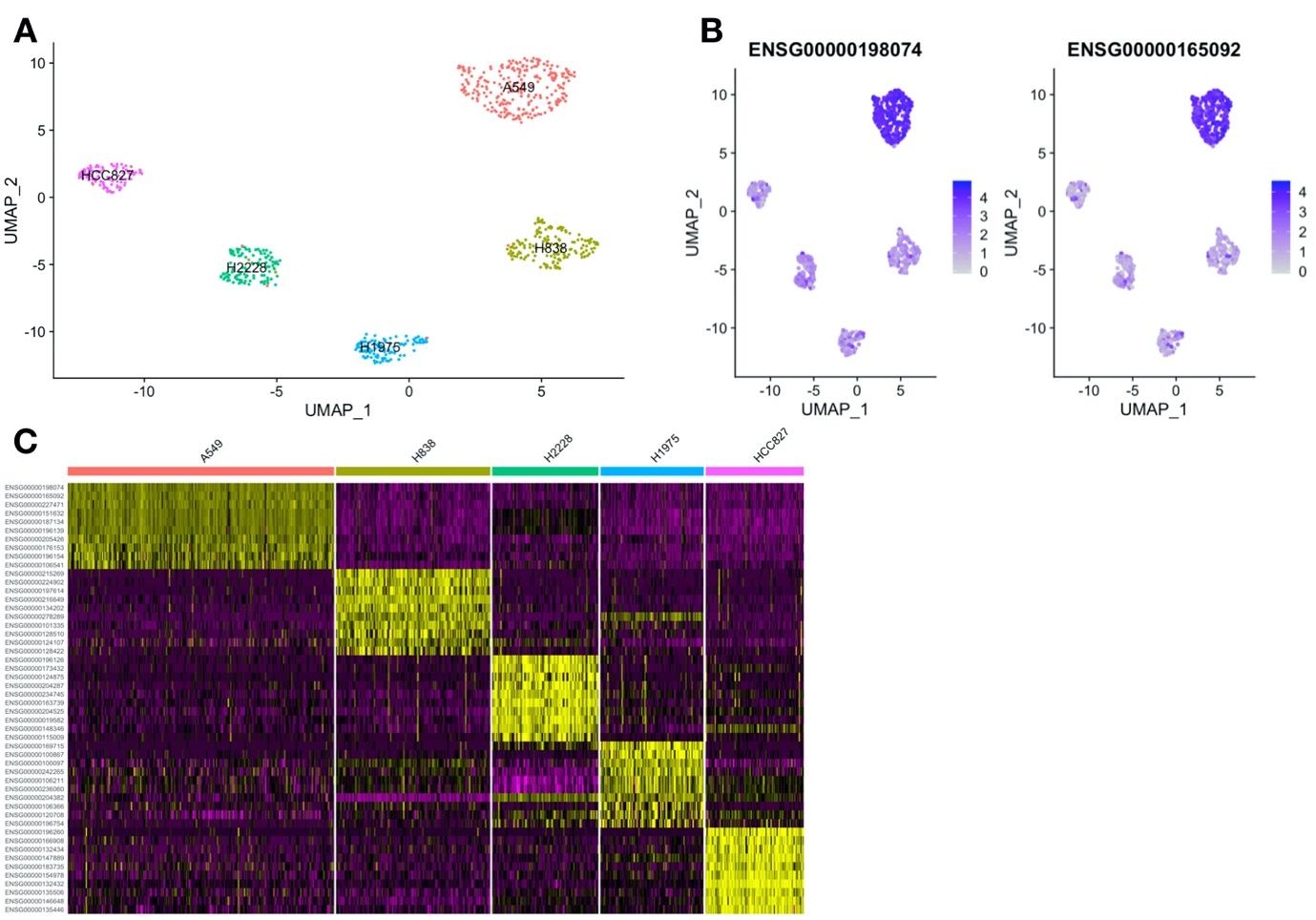
**Seurat**

*Cluster biomarkers*

For training data, five types of cells (A549, H838, H2228, H1775, HCC827) were successfully clustered into 5 clusters (Figure 2A). The top two biomarkers reported (Table 1) are differentially expressed in each group compared to all remaining cells (Figure 2B). Total ten markers were selected for the following classification. Seurat identified cell types with a 97.8% accuracy, 93.81% sensitivity and 98.37% specificity.

Table 1. Top two biomarkers for each cell type.

|  |  |
| --- | --- |
| Cluster | Markers |
| A549 | ENSG00000198074, ENSG00000165092 |
| H838 | ENSG00000216649, ENSG00000216649 |
| H2228 | ENSG00000163739, ENSG00000019582 |
| H1975 | ENSG00000169715, ENSG00000100867 |
| HCC827 | ENSG00000132432, ENSG00000146648 |

 Figure 2. Discovery of biomarkers. A. Clustering of training data. B. The biomarkers enrichment of A549. C. Heatmap of all cell and top ten biomarkers.

Testing data were clustered into five clusters. (Figure 3A) The differential expression of selected biomarkers were visualized for five groups. Five cell types were identified by matching the ten markers to clusters (Figure 3B.C).

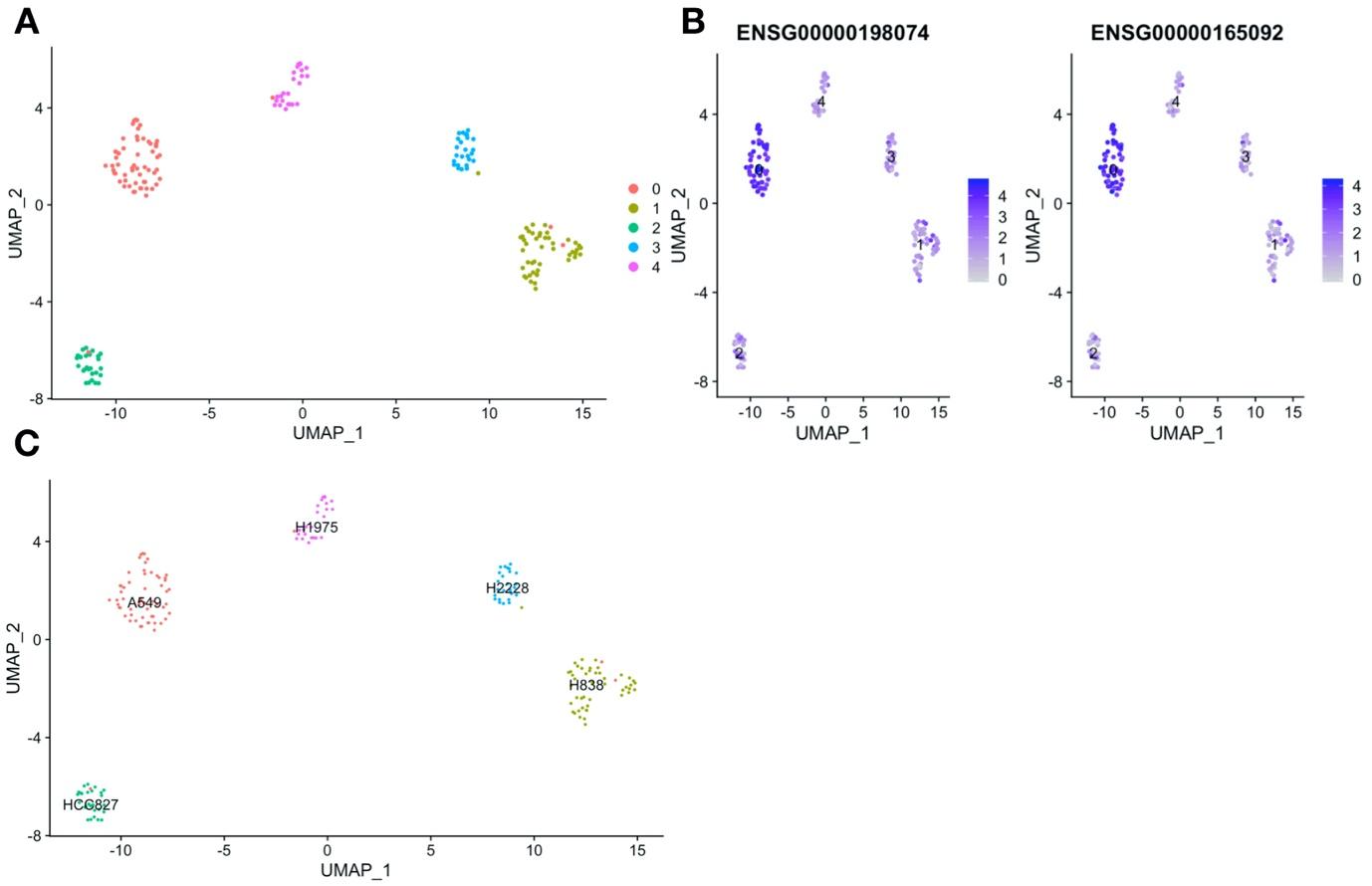


Figure 3. Classification of cell types. A. Clustering of testing data. B. The enrichment of biomarkers selected for A549. C. Identification of each cluster by matching the selected biomarkers.

**Random Forest**

The Random Forest model provides a variable importance ranking for classification (Figure 4). The expression level of ENSG00000165092 is the most important factor to consider when classifying these 5 cell types. Random Forest identified cell types with 98.35% accuracy, 98.53% sensitivity and 100% specificity.

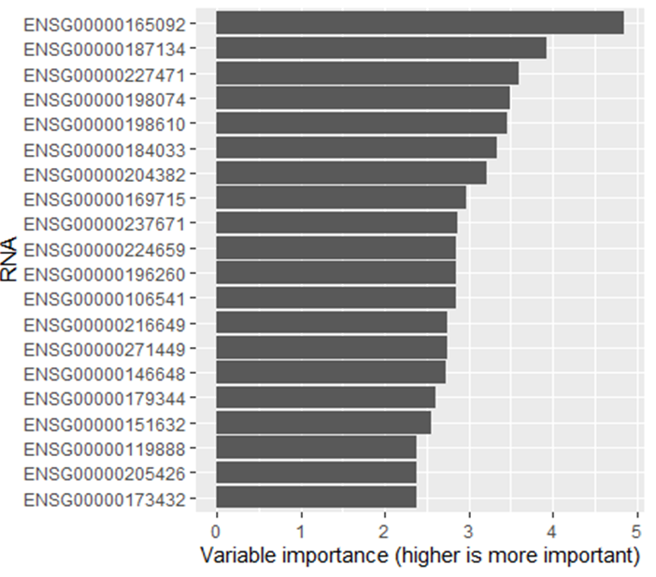


Figure 4. The Random Forest model importance rank based on training set 1

**t-SNE and SVM**

t-SNE was performed on the merged data set. To determine if each data set that was merged clusters differently, the points of each data set were coloured differently. The t-SNE plot (Figure 5A) shows the cells of different datasets do not cluster separately, which means there is no evident influence on the analysis results by merging three datasets. SVM identified cell types with 97.8% accuracy, 96.36% sensitivity and 100% specificity.

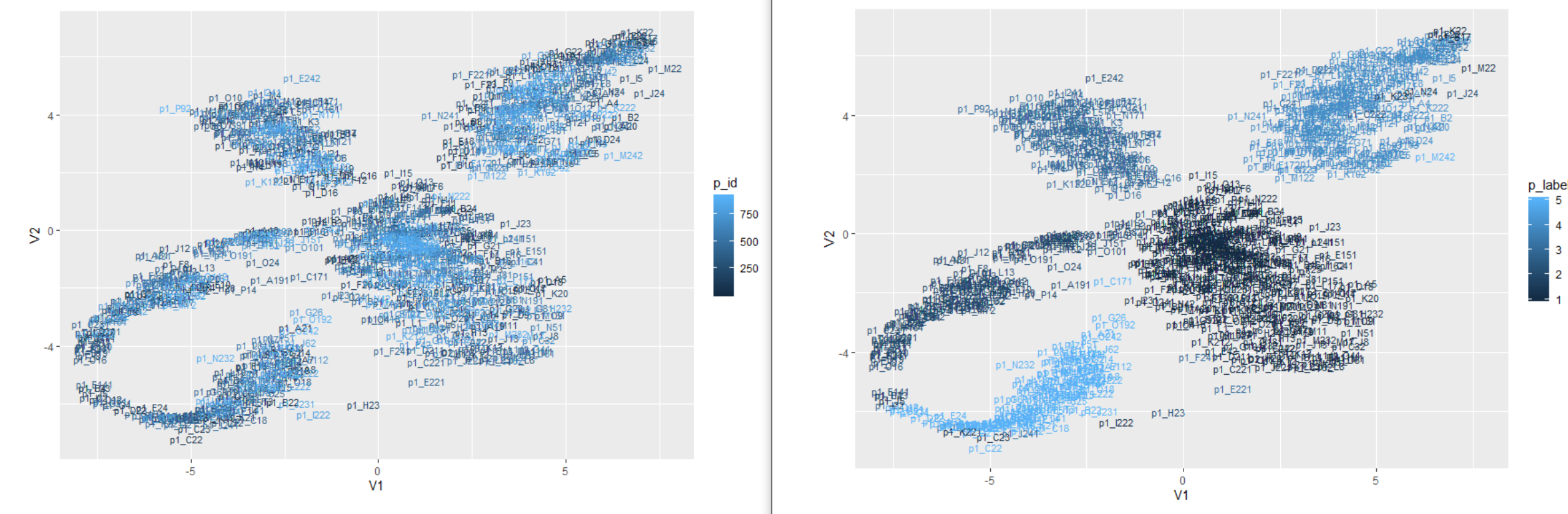


Figure 5. A: t-SNE result grouped by cell ID (IDs of plot data set are from 1 to 297; IDs of plot 2 data set are from 298 to 604; IDss of plot 3 data set are from 605 to 909 ) B: t-SNE result grouped by cell types (1 for A549; 2 for H1975; 3 for H2228; 4 for H838; 5 for HCC827).

**Comparison of F measure between algorithms**

A one-way ANOVA with Tukey’s correction was performed on the mean replicate F measure for each cell type within each algorithm (Table 2). Considering all cell-types, no algorithm performed significantly better (p<0.05) than another.

Table 2: F-Measure values for each cell line in each cell classification technique

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A549 | H1975 | H2228 | H838 | HCC827 | Mean |
| SVM | 0.966 | 0.983 | 0.96 | 0.983 | 0.976 | 0.9736 |
| Random Forest | 0.982 | 1 | 0.969 | 0.960 | 0.965 | 0.9752 |
| Seurat | 0.990 | 1 | 0.988 | 0.991 | 0.972 | 0.9882 |

**Discussion**

The datasets used for this experimental analysis after the combining of the three data sets produced a sample size of 909 cells, which is most likely the minimum data set needed for all the three techniques used in classification. A small data set may cause an inaccurate and/or biased classifier due to the few samples to train and test on. The t-SNE step before SVM required the whole data set to be used as the size of the data has an effect on the dimensionality axis being created. If the data sets divided into test and training were used separately, this would result in dimensionality axis of different measures. In the case of Random Forest, classification is by tree structure using conditions to build the tree, and it is possible to trim the tree with unwanted branches.

The normalization and scaling of the data were done on a basic level as more concern was placed on the classification of the cells. While the Seurat tool provided us with the top biomarkers for each cell type used in the further classification of the cell types, Random Forest offers a ranking of the gene for tree format in the classification of cell types, while SVM technique uses the pattern of margins in a dimensional space to classify cell type. In our interpretation when comparing the working of SVM and Random Forest,in theory the results provided by Random Forest might be much better than those of SVM as the dimensionality space was reduced prior to SVM leading to a loss in information (6,12)

Seurat, Random Forest and SVM all performed well, with overall F measures of 0.999, 0.975 and 0.974, respectively (Table 2). However, as these figures are not statistically significant from each other, it cannot be concluded that one algorithm performs better than another based on F measure. Comparisons of the performance of the models by cell type was not possible as there were no independent replicates.

In this project, the SVM classifier showed to be almost as accurate as the Random Forest classifier with only a 0.6% difference. In a study on support systems for cancer diagnosis, SVM actually showed a lower misclassification rate (17). However, with limited literature on the topic and insignificant differences in accuracy, it is not possible to determine which classifier would be the most accurate.

As the methods were not used on datasets of different sizes, it is not possible to comment on the scalability for our datasets. However, studies have found Seurat could be used on very large data sets of more than 5000 along with SVM and Random Forest that can classify the cell types for large datasets as well (4).

The consistency between the runs with random assignment of data can be proved by the similar results with negligible changes, satisfying the conditions of stability.

Finally, to consider the usability of the programs, Seurat tool is the best option as it was robust and needed no changes to adjust to our dataset because it was designed specifically for scRNA-seq data. Conversely, SVM and Random Forest required the need for adjustments to run the analysis.

In summary, all classification techniques demonstrated accuracy and stability, but only Seurat showed usability. This suggests that Seurat would be the best classification tool for scRNA data.

**Conclusion**

The goal of this study was to test and compare three different techniques for single cell classification. An attempt made to benchmark the tools has been accurate to a certain extent but not entirely reliable due to the presence of small data set and being unable to analyse on scalability . For the 5-cell line data set of 909 cells used for this study, the Seurat tool is more suitable for single cell classification in terms of quality control, user-friendly interface, tutorials and detailed documentation, than Random Forest and support vector machine techniques.

**References**

1. de Kanter JK, Lijnzaad P, Candelli T, Margaritis T, Holstege FCP. CHETAH: a selective, hierarchical cell type identification method for single-cell RNA sequencing. Nucleic Acids Research. 2019;47(16): e95-e.
2. Abdelaal T, Michielsen L, Cats D, Hoogduin D, Mei H, Reinders MJT, et al. A comparison of automatic cell identification methods for single-cell RNA sequencing data. Genome Biology. 2019;20(1):194.
3. Wagner F, Yanai I. Moana: A robust and scalable cell type classification framework for single-cell RNA-Seq data. bioRxiv. 2018:456129.
4. Butler A, Hoffman P, Smibert P, Papalexi E, Satija R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nature Biotechnology. 2018; 36:411.
5. van der Maaten LJP, Hinton GE. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research. 2008;9(nov):2579-605.
6. Awad M, Khanna R. Support Vector Machines for Classification. In: Awad M, Khanna R, editors. Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers. Berkeley, CA: Apress; 2015. p. 39-66.
7. Parmar A, Katariya R, Patel V. A Review on Random Forest: An Ensemble Classifier. 2019;26:1053–60. Available from: http://link.springer.com/10.1007/978-3-030-03146-6
8. Breiman L. Random Forests. Mach Learn. 2001;45(1):5–32.
9. Song Q, Liu X, Yang L. The random forest classifier applied in droplet fingerprint recognition. In: 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2015. 2016.
10. Breiman L. Bagging predictors. Mach Learn. 1996;24(2):123–40.
11. DeFries RS, Chan JCW. Multiple criteria for evaluating machine learning algorithms for land cover classification from satellite data. Remote Sens Environ. 2000;
12. Belgiu M, Drăgu L. Random forest in remote sensing: A review of applications and future directions. ISPRS J Photogramm Remote Sens. 2016;114:24–31.
13. Dietterich TG. Experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Mach Learn. 2000;40(2):139–57.
14. Gislason PO, Benediktsson JA, Sveinsson JR. Random Forests for land cover classification. Pattern Recognit Lett. 2006;27(4):294–300.
15. Ghosh A, Fassnacht FE, Joshi PK, Koch B. A framework for mapping tree species combining hyperspectral and LiDAR data: Role of selected classifiers and sensor across three spatial scales. Int J Appl Earth Obs Geoinf. 2014;26:49–63.
16. Kulkarni VY, Sinha PK. Pruning of random forest classifiers: A survey and future directions. In: Proceedings - 2012 International Conference on Data Science and Engineering, ICDSE 2012. 2012.
17. Floares, A.G., et al. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6685, 2011, pp. 253–262.