Unraveling T Cell Receptor Specificity: An Integrated Approach to Sequence Analysis

Shalomi Fernandes
Department of Engineering and Mathematics
*U*niversity of Bristol
Bristol, United Kingdom
mh23950@bristol.ac.uk

Jiadong Xu
Department of Engineering and Mathematics
*U*niversity of Bristol
Bristol, United Kingdom
jiadong.xu.2021@bristol.ac.uk

Fangnan Wei
Department of Engineering and Mathematics
*U*niversity of Bristol
Bristol, United Kingdom
tj23631@bristol.ac.uk

Jiahui Liu
Department of Engineering and Mathematics
*U*niversity of Bristol
Bristol, United Kingdom
ye23356@bristol.ac.uk

***Abstract*—Predicting the diversity and specificity of T-cell receptors (TCR's) plays a crucial role in immunotherapy. Many studies employ deep learning methods for training, which are time-consuming and demand high computational performance. This study utilizes machine learning techniques to predict the specificity of TCR's, which play a key role in the immune system's response to pathogens and cancer cells. Our research focuses on the broad diversity of TCR sequences in the VDJdb dataset to understand their interactions with specific epitopes. We have implemented approaches including data encoding, distance matrices calculation, dimensionality reduction, clustering and classification to process the TCRs to discover the information from them. The outcomes of these methods have been analysed, revealing both strengths and shortcomings. We also outline potential improvements and future research directions.**

I. Introduction

In the field of immunology, research on T cell receptors (TCR) has always been a hotspot. TCRs activate T cells by recognizing and binding to peptide-MHC complexes on the surface of antigen-presenting cells. With the advancement of biotechnology, especially the popularization of high-throughput sequencing technologies, researchers can obtain a large amount of TCR sequence data, which is crucial for understanding the diversity and pathological mechanisms of the immune system. However, existing studies often rely on traditional biostatistical methods or high-demand deep learning techniques to analyze TCR data. These methods often fall short when handling complex biological data. For example, existing methods struggle to accurately predict the binding affinity between TCRs and specific peptide-MHC complexes, limiting their application in precision medicine and personalized immunotherapy. By integrating various types of data and algorithms, it is possible to enhance the accuracy and efficiency of TCR analysis, thus better advancing research outcomes and clinical applications in immunology. In this study, advanced machine learning methods were used to analyze TCR sequence data. By comparing the performance of traditional methods and new algorithms, the potential applications of these new technologies in TCR research were explored. The results not only demonstrate the advantages of new algorithms in processing TCR data but also provide a new perspective on understanding the diversity of TCRs. With the further development of computing and biotechnologies, TCR research combined with machine learning is expected to become more precise and efficient. This will not only deepen our understanding of how the immune system recognizes and responds to pathogens but also promises to advance personalized medical and precision immunotherapy strategies, providing more personalized and effective treatment options for patients.

II. Literature Review

In a study using single-cell V(D)J sequencing to extract T cells and then perform single-cell sequencing to capture the diversity of TCRs, T cell TCRs from 12 COVID-19 patients were analyzed and compared with 6 healthy controls and other viral infection samples. Particular attention was paid to the analysis of V and J gene combinations, which play a key role in TCR diversity [1]. A mathematical framework was proposed in one study to explain how TCRs bind to pMHC. This finding is crucial for understanding how T cells recognize and efficiently bind to pMHC complexes through TCRs [2]. With the help of a new tool "SPAN-TCR," a TCR library for multiple known antigens was analyzed, comparing and analyzing the amino acid composition of the CDR3 region of TCRs and revealing similarities and differences in amino acid usage and structure between different TCRs through entropy analysis [3]. A new computational method, SETE, uses the effects of adjacent amino acids in the CDR3 beta sequence, focusing on the influence of neighboring amino acids, and employs gradient boosting decision-making and feature learning to predict the binding of TCRs to epitopes [4]. The issue of TCR specificity in structural and biophysical

studies explored the interactions between TCRs, MHC, and peptides [5]. A new deep learning model, EPIC-TRACE, uses the sequences of CDR3, V, and J gene regions of TCR's alpha and beta chains, along with the sequences of epitopes and MHC, to predict TCR binding to unseen epitopes [6]. In an experiment comparing TCRs from COVID-19 patients with those from healthy controls using V(D)J sequencing technology, TCRs from PBMCs and BALF were analyzed. By comparing different TCR characteristics such as CDR3 amino acid length distribution, specific VJ gene segments, and their pairing, the dynamic changes in immune response were reflected [7]. A new mathematical framework, "GIANA," can efficiently cluster TCR sequences and classify multi-disease immune libraries. It transforms the TCR sequence alignment and clustering problem into a nearest neighbor search in high-dimensional Euclidean space, significantly enhancing computational efficiency to handle up to millions of sequences [8]. By comparing the similarity between TCR sequences, clustering of TCRs was performed using sequence alignment and scoring matrices [9]. A study using "metaclonotypes" for TCR analysis identified them using tcrdist3, thereby enhancing the use of TCRs as biomarkers [10].

III. Methodology

This study preprocessed, encoded, and calculated the distance of TCR sequences on the data, and then performed dimensionality reduction, clustering, classification, and predictive analysis. The encodings used in this article are One-Hot, BLOSUM 62, and GIANA Encoding. Among them, distance calculation methods include TCRDist and Levenshtein. After comparing multiple algorithms for dimensionality reduction, UMAP was finally adopted. Two algorithms are used for clustering, hierarchical clustering and DBSCAN. Finally, classification is performed using Logistic Regression, SVM, and Random Forest.

A. Pre-Processing

In this study, the missing values and duplicates in the data are deleted through several key steps of data preparation, data cleaning, and data screening. Next, relevant features are selected based on their importance to the analysis goals. Additionally, applied normalization or standardization techniques to bring features to a comparable scale, and used UMAP to visualize complex data sets and simplify subsequent analysis. Collectively, these steps are critical to preparing the dataset for further exploration and modeling. These steps are discussed in detail in the model building section.

B. Encoding Methods:

1) One-Hot Encoding: One-hot encoding is a technique for representing categorical variables as binary vectors. In this encoding, for each variable, there is a vector representing its possible values, and the length of the vector is the same as the number of these possible values. In this vector, only one element is 1 (representing the variable category), and all other elements are 0. In this work, a function for one-hot encoding is first defined, which accepts two parameters: the sequence to be encoded and the list of all elements of possible amino acids. This function creates a two-dimensional array. The number of rows in the array is equal to the length of the sequence, and the number of columns is equal to the number of amino acid types.

The use of one-hot representation of TCR sequences has some limitations in downstream analysis. A major limitation is its inability to capture the sequential nature of amino acids, which results in the loss of important structural and functional information. In addition, one-hot encoding may lead to high-dimensional sparse representation, especially for those TCR sequences with variable length of CDR3 regions, which brings challenges to efficient sequence analysis and sequence comparison. To overcome these limitations, one possible approach is to incorporate information on the CDR3 length distribution.

2) BLOSUM 62 encoding:

Since one-hot encoding simply represents amino acids as binary vectors without considering the biochemical properties and evolutionary relationships of the amino acids themselves, it may not effectively capture subtle differences between sequences. Instead, the use of BLOSUM62(Blocks Substitution Matrix) for distance matrix calculations has been made. BLOSUM62 scores pairs of amino acids based on their frequency observed in related proteins, capturing similarities and differences between sequences in a biologically meaningful way. This approach provides a more accurate representation of sequence similarity than one-hot encoding and is more suitable for our analysis.

BLOSUM 62 encoding is a protein sequence encoding method based on the substitution matrix score of BLOSUM. The BLOSUM 62 substitution matrix is derived from the statistical analysis of a large number of known protein sequences and allows the assessment of similarity between two amino acids. In BLOSUM 62 encoding, each amino acid is mapped to a vector whose length is equal to the number of rows of the substitution matrix. Each element in the vector represents the similarity score of that amino acid to the corresponding row in the substitution matrix.

3) GIANA encoding:

The GIANA(Geometric Isometry-based TCR Alignment Algorithm) is a efficient tool for clustering TCR sequences [8]. Its encoding mechanism first represent the CDR3 sequence in a numerical format. Each of the twenty amino acids is encoded into a numeric vector, forming the foundational elements for constructing CDR3 sequence representations. Then it applies serial non-commuting linear transformations to these numeric vectors. This process utilises a unitary transformation matrix belonging to a 6-order cyclic group (G6), which transforms the CDR3 sequence data into a high dimensional Euclidean space. The distances between the

transformed vectors of CDR3 sequences are highly correlated with their Smith-Waterman alignment scores, which are traditionally used to assess sequence similarity based on biological relevance.

C. Distance Calculation Methods:

1) TCRDist: TCRdist, a specialized tool for computing pairwise distances between T-cell receptor sequences was used. The dataset was divided into six subsets: human alpha, human beta, combined human alpha-beta, mouse alpha, mouse beta, and combined mouse alpha-beta as shown in Fig.1. which shows the subdivision of data for TCR Distance Matrix Calculation. Each subset was processed to calculate the TCR distance matrix for both alpha and beta chains. The distance matrices were essential for quantifying the similarity between TCR sequences, which is a crucial step for further analysis such as clustering and dimensionality reduction.

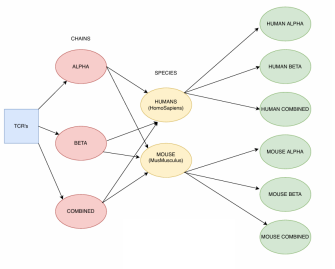

Fig. 1: Subdivision of TCR Distance Matrix Calculation

TCRDist is a toolkit for calculating TCR distances, which can be calculated based on their sequence characteristics (e.g., CDR3 sequences) and possibly other information (e.g., expression patterns of TCRs).For a given set of TCR sequences, the distances between all possible pairs are calculated, and the results are formed into a distance matrix, each element in the matrix represents the distance or similarity between the pairs, and once the distance matrix is calculated, visualization tools can be used to show the similarity or difference between the TCRs.

As with the Levenshtein distances presented in the next section, we computed the distance matrices for a single Alpha chain, a single Beta chain, and a combination of the two, respectively, and used heat maps to display the matrices. The heat maps are showed in the appendix. The results of the distance calculations are used in the clustering part.

2) Levenshtein distance:
Levenshtein distance is a measure of how similar two strings are. It often referred to as edit distance, quantifies the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one string into another. This approach is particularly useful in immunological studies, where similarities in TCR sequences can indicate clonal expansion or shared antigen specificity between different T cells.

D. Dimensionality Reduction Method(UMAP):

Uniform manifold approximation and projection (UMAP) is a nonlinear dimensionality reduction technique that is particularly effective for high-dimensional data. It constructs a high-dimensional map representing the proximity of data points, and then projects the optimal high-dimensional map into a low-dimensional space. In bioinformatics, it can be used for understanding the complexity of TCRs. In this experiment, UMAP was used to reduce high-dimensional TCR data to a two-dimensional space for each TCR type: alpha chain, beta chain, and combined alpha-beta chain. By plotting these reduced dimensions and coloring them according to antigen specificity, it helps to reveal underlying patterns and groupings that are not immediately apparent in high-dimensional data. This visualization allows direct comparison of the clustering of TCR specificities within each chain type. Differences in clustering patterns between alpha, beta, and combined data sets can provide insight into the uniqueness of their antigen recognition properties. In the experiments, we removed epitopes that contained too few data points since these small categories are difficult to be tracked and analysed in the plots. We also removed TCRs with a ”vdjdb.score” of 0, as these are low confidence/no information data. The same goes for the clustering section.

E. Clustering Methods

1) DBSCAN:
The DBSCAN (Density-Based Spatial Clustering of applications with Noise) algorithm is an effective density-based clustering method. Clusters can be identified as high-density regions separated from low-density regions. This method is particularly beneficial for TCR sequences in immunology because it does not require a prespecified number of clusters and can handle noise and outliers efficiently. DBSCAN works by identifying ”core” samples with the minimum number of neighbors within a given radius, and marking points within the radius that do not meet the core criteria as ”boundary” points. This approach can expertly group TCRs that exhibit high similarity in sequence and function, allowing researchers to discern immune response patterns and identify unique or shared antigen encounters between different T cells.

2) Agglomerative Hierarchical Clustering: Agglomerative Hierarchical Clustering is a bottom-up clustering algorithm that gradually merges data points into clusters, and represents the similarity relationship between data points through a tree-like structure. This method is particularly suitable for biological data like TCR sequences, where it can provide detailed insights into the relationship and similarity between various sequences. The process results in a dendrogram that illustrates the sequence of cluster mergers and can be cut at various levels to obtain a specific number of clusters. This method allows researchers to analyze the hierarchical structure of TCR data, potentially revealing new insights into how TCR diversity relates to immune response functions.

3) Evaluation Metrics:
In order to be able to accurately evaluate the specificity-based TCR clustering performance, Pure Clustering Fraction (PCF), Pure Clustering Retention (PCR) and normalized mutual information (NMI) are used as the index results of clustering evaluation. Firstly, purity is defined as the percentage of TCRs in a cluster that are specific for the most common epitope. Pure cluster fraction is defined as the percentage of pure clusters out of all clusters, and pure cluster retention is defined as the percentage of TCRs classified as pure clusters out of all TCRs. These two metrics measure the ability of a clustering algorithm to cluster TCRs with the same specificity into a class. Pure Cluster Score and Pure Cluster Retention focus on clusters that are 100% pure, while 100% purity tends to be more common in smaller clusters, which also means these metrics prefer smaller clusters. Therefore, we also use NMI to evaluate the clustering results which is defined as twice the mutual information divided by the sum of the entropies of the two labels (i.e., cluster label and specificity). Mutual information can reflect the amount of information added to our knowledge of a class when we are told what a cluster is, while entropy increases with the number of clusters, and normalization of mutual information can impose a penalty for too many clusters.

F. Model Building:

1) Data Preparation: The dataset contained several columns that were not necessary for building the predictive models. By retaining key features the dataset is refined to include only those attributes that provide essential information about the TCRs and their interactions with antigens. Epitopes with fewer than 10 occurrences are considered insufficient for reliable modeling and are thus filtered out. This focuses the dataset on more common epitopes, which improves the model's ability to learn relevant patterns and make accurate predictions. Calculation of the length of each CDR3 sequence and retaining those within a practical range (10 to 20 amino acids) is done as depicted by the box plot shown in Fig 2, as very short or very long sequences might represent sequencing errors or unusual variations that could skew the model training.

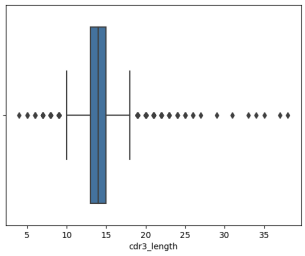

Fig. 2: Distribution of cdr3 sequence lengths

The occurrences of each species were counted and filtered to include only those species with sufficient representation (over 1000 occurrences), thus focusing on more common contexts in the training data. Further, filtration of the dataset was done to exclude rows where complex.id is 0 which removed entries that did not meet certain criteria essential for the analysis, such as incomplete data entries or placeholders that do not correspond to actual TCR sequences. Doing this ensures that the dataset consists only of meaningful and relevant entries, enhancing the quality of our training data.

This model first uses a combination of Label Encoding and Binary Encoding to encode features like 'gene', 'species', 'epitope','mhc.a', 'mhc.b', 'v.segm' and 'j.segm' and 'mhc.class'. Next, using a function GIANA_encoder_pd, the CDR3 sequence is encoded into a vector representation, first traversing each amino acid sequence, for each sequence, removing the C and J segments at the beginning, and the F segment at the end, encoding the sequence using a predefined M6 matrix. For each row, the encoded CDR3 sequence and the normalized eigenvalues are first concatenated, and then they are combined into a single array. Finally, the resulting array contains the encoded CDR3 sequence and all the normalized eigenvalues, so that each row represents a complete sample eigenvector.

2) Data Selection: A decision to focus exclusively on Human Species (Homo Sapiens) was made for model building and predictions due to the significantly larger volume of data available—51,535 entries for humans compared to just 4,173 for mice (Mus Musculus). This stark disparity allows for a more robust and generalizable model for humans, where the richer dataset ensures more reliable predictive performance. Additionally, focusing on human data aligns with the primary goal of applying findings directly to human medicine, avoiding the risks of undertraining and overfitting associated with the limited mouse data. This approach not only makes efficient use of the extensive human data but also enhances the clinical relevance of the predictions.

Model evaluation was performed by segmenting the data set to obtain alpha and beta chains. Each chain potentially interacts differently with antigens, influencing the specificity and strength of immune responses. By analyzing them separately, it's possible to isolate and understand the unique contributions of each chain in the antigen recognition process. This approach can help us perform a more nuanced analysis of TCR behavior, providing insights that may be obscured in combined model. Furthermore, separate evaluations enable the identification of chain-specific patterns, which are crucial for developing targeted therapeutic strategies and improving the accuracy of predictive immunological models. In this study, we used accuracy and F1 score as key evaluation metrics for our models. Accuracy measures the overall correctness of the model across all classes, while the F1 score provides a balance between precision and recall, especially important in situations with class imbalance, assessing the model's ability to correctly classify each class.

G. Model Implementation:

1) Logistic Regression (Baseline Model): Logistic regression was chosen as the baseline model for predicting TCR specificity due to the robustness of the logistic regression model and the simplicity of the binary classification task driving the choice. Verifying the model's performance on unseen data was done by dividing the data set into a training set and a test set. The primary evaluation metric chosen is the F1 score because the accuracy of the model can be misleading due to the unbalanced nature of the dataset. This is particularly important given the variability in epitope representation within the dataset.

2) SVM Model: In this project, the Support Vector Machine (SVM) classifier was employed as another method for building predictive models to understand antigen specificity from TCR sequences. SVM is effective in high-dimensional spaces because it is good at finding the optimal hyperplane that maximizes the separation between different classes. Specifically, the linear kernel was utilized to handle the linearly separable aspects of the data. Training the SVM and verifying its performance was done on unseen data by splitting the data set into training and test sets. This approach provides a systematic approach to model antigen recognition patterns in an attempt to predict responses based on molecular features of TCR sequences.

3) Random Forest Classifier Model: The Random Forest is a significant improvement over logistic regression and SVM models because the random forest classifier uses an ensemble learning method that integrates multiple decision trees by averaging the results of these decision trees to produce a more stable and accurate Prediction. This method effectively reduces the variance and overfitting problems often seen in a single decision tree model, leading to enhanced generalizability across diverse datasets. Specifically, the Random Forest model exhibited higher weighted average F1 scores, indicating that the random forest classifier has good adaptability to imbalanced data sets and has excellent performance in terms of precision and recall for each category.

H. Model Tuning:

Hyperparameter tuning of the Random Forest model significantly enhanced its performance for both Alpha and Beta chains, indicating that optimizing model parameters is important for processing complex immunological data sets. This experiment was tuned using RandomizedSearchCV, where parameters such as the number of estimators, maximum features, maximum depth, minimum sample split, minimum sample leaves, and use of bootstrap are varied in a predefined grid. This process was repeated for 10 iterations using 3-fold cross-validation to maximize robustness and effectiveness.

These traits make Random Forest particularly effective in handling the complex patterns and intricacies of antigen specificity prediction, where robustness against diverse and skewed data distributions is crucial. This enhanced capability to deliver more reliable and consistent predictions across a wider range of T-cell receptor specificities underlines why Random Forest is a preferable choice over the simpler logistic regression and the linear-bound SVM in this context.

IV. Results and Discussion

A. Dimensionality Reduction Results

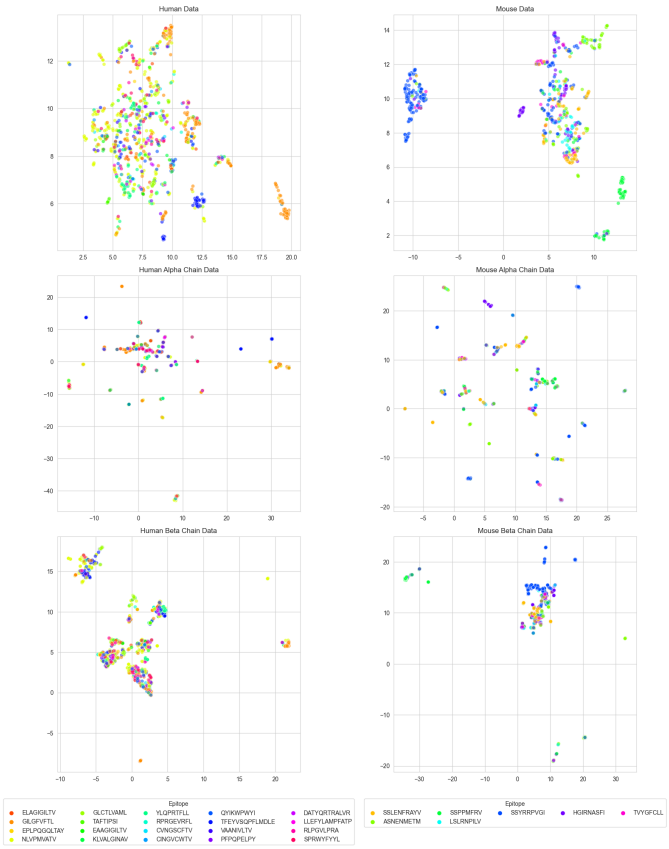

Fig. 3: UMAP results

In the dimensionality reduction results considering both Alpha and Beta chains, the human TCRs are widely distributed across the graphs as shown in Fig. 3, showing a more diverse clustering with some tightly clustered groups and some scattered points. There are some overlapping point, but overall, the data distribution is more dispersed. The mouse TCR data shows a relatively compact cluster structure, with obvious boundaries and distances between TCR clusters. For example, the TCR clusters corresponding to antigens such as "SSYRRVPGI" have distinct boundaries and distances between them and other data points. Therefore, when considering both Alpha and Beta chains in combination, mouse TCRs show more concentrated specificity compared to human.

Compared to the dimensionality reduction results of combining Alpha and Beta chains, the single Alpha chain TCR clustering results for both humans and mouses were

more dispersed, but the number of points appeared to become fewer as a large number of data points overlapped. Overall, the data points form a large number of small clusters that are more dispersed from each other and more tightly packed internally. The data points in the small clusters have many highly correlated features. The results of the Beta chains resulted in tighter clusters with more distinct boundaries than the previous two. The human TCR data formed several larger clusters, and these clusters did not show domination by specific epitopes within the clusters, but rather a uniform distribution of data points for various epitopes. The mouse single Beta chain data showed one large cluster and some discrete points, and unlike the human data, the clusters showed relatively obvious clustering of data points of different epitopes.

Overall, the human TCR is more dispersed and more difficult to form a single cluster compared to the mouse. It is possible that the high genetic diversity of the human population, the more complex immune system, and the exposure to a wide variety of pathogens in different environments have caused the human TCR to exhibit a more diverse clustering behaviour. Single Alpha chain data exhibit a large amount of overlap and form small dispersed clusters; single Beta chain data are denser with distinct cluster boundaries. This may be due to the fact that during gene rearrangement, the Alpha chains does not have a D region and only undergoes VJ rearrangement, which has a lower antigenic specificity compared to Beta. And after combining Alpha and Beta, the data points appeared to be more dispersed overall. We believe the reason is different TCRs have different combinations of features on the Alpha and Beta chains, resulting in a more diverse distribution, making the data points more scattered after combination.

B. Clustering Results

GROUP	PURITY FRACTION	PURITY RETENTION	NMI
Hierarchical Clustering on Human Data	0.93	0.87	0.66
Hierarchical Clustering on Mouse Data	0.91	0.81	0.50
DBSCAN on Human Data	0.52	0.12	0.37
DBSCAN on Mouse Data	0.62	0.20	0.49

Fig. 4: Clustering results with Levenshtein distance

GROUP	PURITY FRACTION	PURITY RETENTION	NMI
Hierarchical Clustering on Human Data	0.91	0.80	0.63
Hierarchical Clustering on Mouse Data	0.92	0.78	0.58
DBSCAN on Human Data	0.45	0.13	0.36
DBSCAN on Mouse Data	0.64	0.22	0.39

Fig. 5: Clustering results with TCRDist distance

From the clustering results showed in Fig. 4 and 5, it is clear that the different ways of calculating the distances have very little effect on the results. With the same species and clustering method, the difference between the two distance calculation methods is below 0.1 for any metric. When using hierarchical clustering, TCRDist produced slightly higher NMIs for both human and mouse data. Results were mixed when using DBSCAN clustering, with TCRDist producing lower purity scores for the human data but slightly higher purity scores for the mouse data. Both TCRDist and Levenshtein distances are widely used tools for TCR distance calculations, although the principles of the two are different, with the former referring to the properties of the CDR3 sequence in relation to different amino acids and the latter focuses on the differences between sequences, they have no significant impact on the task of clustering.

When hierarchical clustering was performed, the clusters of human TCRs performed better than those of mouses in general, but DBSCAN obtained the exact opposite result. This may imply that the human TCR data distribution may have a more pronounced hierarchical structure that is more consistent with hierarchical clustering's assumptions about the data distribution, thus making it easier to generate clusters that conform to the intrinsic organisational structure of the dataset. In contrast, the low performance of DBSCAN is likely due to the algorithm's difficulty in forming coherent clusters with that data distribution. Due to its principle of operation, DBSCAN is difficult to handle data with large differences in density. This suggests that the human TCR is likely to have a more dispersed or variable density distribution compared to the mouse data, which is consistent with the previous analysis of the dimensionality reduction results.

Regardless of the distance calculation method for any species, the performance of DBSCAN is much weaker than that of hierarchical clustering, the purity fraction of DBSCAN results may even be half of that of hierarchical clustering, and the purity retention is often only a quarter of that of hierarchical clustering, and the NMI also has a relatively small difference. The purity fraction of DBSCAN results may even be half of that of hierarchical clustering, the purity retention is often only a quarter of that of hierarchical clustering, and the NMI has a relatively small gap. Even if we choose the parameters that can make all of metrics reach a relatively high level by comparing the parameters before conducting the experiments, the performance of DBSCAN is still far inferior to that of hierarchical clustering. Fig. 6 shows the curves of purity fraction, purity retention and NMI with eps parameter in DBSCAN algorithm. We use this graph to select the optimal value of eps. As demonstrated in the dimensionality reduction section, the data is denser in some regions of the space and sparser in others, with huge gaps in density within different epitopes. it is very difficult for DBSCAN to form effective clusters in regions where the data is too sparse, or where the density varies too much.

This problem makes it impossible to find suitable parameters, if the radius is set too large it will lead to too few

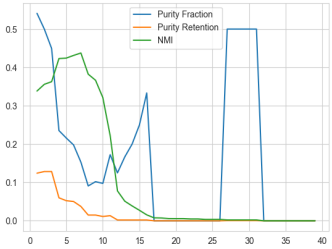

Fig. 6: Curves of Purity fraction, Purity retention and NMI with eps parameter in DBSCAN algorithm used to select the optimal value of eps.

clustering results with 0 purity, while setting the radius too small will lead to many points being classified as noise. We believe that this is the main reason for the poor performance of DBSCAN in clustering against TCR. On the other hand, hierarchical clustering is robust and can handle datasets with different density of clusters by iteratively merging different levels of clusters. Therefore, it is more suitable for dealing with complex and variable data distributions like TCR. Overall, the results of hierarchical clustering are good, with NMIs of 0.58-0.66 indicating moderate to high similarity between the clustering results and the true labels. This suggests that the model captures the structure of the dataset better, but there may be some mismatches or discrepancies. The purity fraction of 0.91-0.93 and the purity retention of 0.78-0.87 suggest that the clustering results retain most of the structure of the dataset, and that in a very large majority of the clusters the points belong to the same class, while most of the data points of the same class points were able to be grouped into the same clusters.

C. Logistic Regression Classifier Results

Model evaluation revealed that while the overall accuracy appeared high at approximately 87.5% for the alpha chains and 87.75% for the beta chains, the F1 scores for many classes were very low, indicating poor performance in correctly classifying many specific epitopes. This discrepancy between high accuracy and low F1 scores highlighted the challenges of working with imbalanced datasets and underscored the necessity of choosing appropriate metrics for performance evaluation. The classification report showed substantial variability in precision, recall, and F1 scores across different classes, with many classes showing zero values in these metrics, suggesting that the model struggled with minority classes. This outcome stresses the importance of further model adjustments and potentially exploring more complex models or resampling techniques to better handle class imbalance and improve the model's ability to generalize across less frequent classes.

D. SVM Results

The SVM classifier, employing a linear kernel, recorded accuracy rates of about 87.51% for alpha chains and 89% for beta chains. This performance is quite similar to what was observed with the logistic regression model, indicating comparable overall effectiveness across the dataset. A closer examination of the F1 scores from the classification report reveals that the SVM faces difficulties with several classes, with many achieving zero F1 scores. This highlights the challenge of dealing with minority classes in an imbalanced dataset. Nonetheless, the SVM does excel in specific areas, achieving perfect F1 scores in some classes where logistic regression struggles. This shows the SVM's strength in handling certain segments of data where its optimization techniques are most effective, particularly in balancing precision and recall, as reflected by a weighted average F1 score of 0.88 for both alpha and beta chains. Although the accuracy of the SVM is similar to that of logistic regression, its higher F1 scores in certain classes suggest it could offer more dependable predictions for specific T-cell receptor specificities, which is especially important in clinical and immunological settings.

E. Random Forest Classifier Results

The Random Forest classifier, when applied separately to the Alpha and Beta chains, shows notable strengths and weaknesses in performance. Both chains achieve high accuracy levels, with approximately 90% for Alpha and 92% for Beta, indicating the model's strong overall predictive power across the dataset. However, a more detailed analysis using F1 scores reveals varied performance across different epitopes. Many classes exhibit low or even zero F1 scores, indicating challenges in classifying minority classes within the imbalanced dataset. On the other hand, some classes achieve high F1 scores, demonstrating the model's ability to accurately identify specific classes.

This variation highlights the Random Forest model's capacity to handle complex data through its ensemble approach, which captures multiple decision-making pathways and minimizes variance compared to simpler models like logistic regression. The notable weighted average F1 scores of 0.91 for Alpha and 0.92 for Beta underscore its effectiveness in balancing precision and recall among diverse classes, positioning it as a strong option for addressing the complexities of TCR specificity. Despite its success in general accuracy and class-specific predictions, there is still room for enhancement in precision across the board, suggesting a need for further model refinement and strategy development.

F. Results after Hyperparameter Tuning on the Random Forest Model

After refining the model's settings, we noticed clear improvements. For the Alpha chain, adjusting the hyperparameters boosted the weighted average F1 score to 0.91, demonstrating enhanced balance in predicting various classes accurately. The Beta chain exhibited similar improvements, achieving a weighted average F1 score of 0.91 as well. These

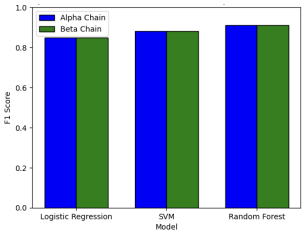

Fig. 7: Comparison of F1 scores across different models for the Alpha and Beta Chains

improvements suggest that the refined Random Forest model handles the dataset's varied and skewed distributions more effectively, resulting in more dependable predictions. Such enhancements are especially valuable in clinical environments, where precise model predictions are essential for designing effective T-cell receptor-based immunotherapies.

V. Further Work and Improvement

A. Short Term Improvements:

1) Refining Feature Sets: If there was a scope for more time, refining the features used could be advantageous. This might include diving into more intricate feature interactions or applying more complex transformations to the existing data like CDR3 lengths.

2) Model Tuning and Regularization Techniques: While initial models have been implemented, additional tuning of hyperparameters and applying regularization methods might improve the accuracy of our models and reduce the risk of overfitting.

3) Enhanced Cross-Validation Techniques: Employing a better cross-validation process would help confirm that our models would perform well across various data segments, ensuring reliability and stability of our findings.

4) Incremental Data Updates: Introducing new features to allow the models to adapt and learn from new data continuously could eliminate the need for frequent retraining.

B. Long Term Extensions:

1) Implementation of Artificial Intelligence and Deep Learning Models: Over time, incorporating advanced methods like convolutional or recurrent neural networks could drastically improve our models' ability to detect complex patterns in antigen specificity. Exploring transformers, known for their effectiveness in sequence-based analysis, could also be beneficial.

2) Automated Real-Time Analytical Systems: Developing an automated system that perfectly integrates with current medical frameworks to offer real-time insights could significantly aid in rapid medical decision-making.

3) Widening the scope of the research: Expanding our models to cover different species and immune conditions could enhance the applicability of our research. This would involve gathering and analyzing a broader array of data sets.

4) Fostering Collaborative Efforts: Forming strategic alliances with educational bodies and industry peers could foster innovation and provide access to exclusive data and specialized expertise. This approach could accelerate advancements in predictive modeling within this field.

VI. Conclusion

This study uses machine learning methods for understanding TCR specificity, which is important for improving immunotherapy. By implementing techniques like SVM, Logistic Regression, and Random Forest, along with new methods to data encoding and clustering, this study aims to tackle the complex nature of TCR sequences. The careful preparation, encoding, and analysis of the data allow for a deeper understanding of how TCRs interact with antigens, which is vital for precision medicine. The results show us that while the models predict well overall, they may vary in how effectively they classify individual epitopes, highlighting the challenges of working with imbalanced data. This study not only advances research in immunology but also aims to improve how we customize treatments, making them more patient-specific. Despite achieving good accuracy, the research suggests there's still room to improve the models to make them even more precise. This ongoing effort will help refine treatments and could lead to better outcomes for patients.

Github URL: https://github.com/UoB-DSMP-2023-24/dsmp-2024-group4.git

References

[1] P. Wang et al., 'Comprehensive analysis of TCR repertoire in COVID-19 using single cell sequencing', Genomics, vol. 113, no. 2, pp. 456–462, Mar. 2021, doi: 10.1016/j.ygeno.2020.12.036.

[2] C. O. Barkan, 'A structural model of TCR-pMHC catch bonding', Biophys. J., vol. 123, no. 3, p. 206a, Feb. 2024, doi: 10.1016/j.bpj.2023.11.1302.

[3] A. M. Xu et al., 'Entropic analysis of antigen-specific CDR3 domains identifies essential binding motifs shared by CDR3s with different antigen specificities', Cell Syst., vol. 14, no. 4, pp. 273-284.e5, Apr. 2023, doi: 10.1016/j.cels.2023.03.001.

[4] Y. Tong et al., 'SETE: Sequence-based Ensemble learning approach for TCR Epitope binding prediction', Comput. Biol. Chem., vol. 87, p. 107281, Aug. 2020, doi: 10.1016/j.compbiolchem.2020.107281.

[5] N. K. Singh, T. P. Riley, S. C. B. Baker, T. Borrman, Z. Weng, and B. M. Baker, 'Emerging Concepts in TCR Specificity: Rationalizing and (Maybe) Predicting Outcomes', J. Immunol., vol. 199, no. 7, pp. 2203–2213, Oct. 2017, doi: 10.4049/jimmunol.1700744.

[6] D. Korpela, E. Jokinen, A. Dumitrescu, J. Huuhtanen, S. Mustjoki, and H. Lähdesmäki, 'EPIC-TRACE: predicting TCR binding to unseen epitopes using attention and contextualized embeddings', Bioinformatics, vol. 39, no. 12, p. btad743, Dec. 2023, doi: 10.1093/bioinformatics/btad743.

[7] X. Zhu et al., 'A comparative analysis of TCR immune repertoire in COVID-19 patients', Hum. Immunol., p. 110795, Apr. 2024, doi: 10.1016/j.humimm.2024.110795.

[8] H. Zhang, X. Zhan, and B. Li, 'GIANA allows computationally-efficient TCR clustering and multi-disease repertoire classification by isometric transformation', Nat. Commun., vol. 12, no. 1, p. 4699, Aug. 2021, doi: 10.1038/s41467-021-25006-7.

[9] M. Vujovic et al., 'T cell receptor sequence clustering and antigen specificity', Comput. Struct. Biotechnol. J., vol. 18, pp. 2166–2173, Jan. 2020, doi: 10.1016/j.csbj.2020.06.041.

[10] K. Mayer-Blackwell et al., 'TCR meta-clonotypes for biomarker discovery with tcrdist3 enabled identification of public, HLA-restricted clusters of SARS-CoV-2 TCRs', eLife, vol. 10, p. e68605, Nov. 2021, doi: 10.7554/eLife.68605.

APPENDIX

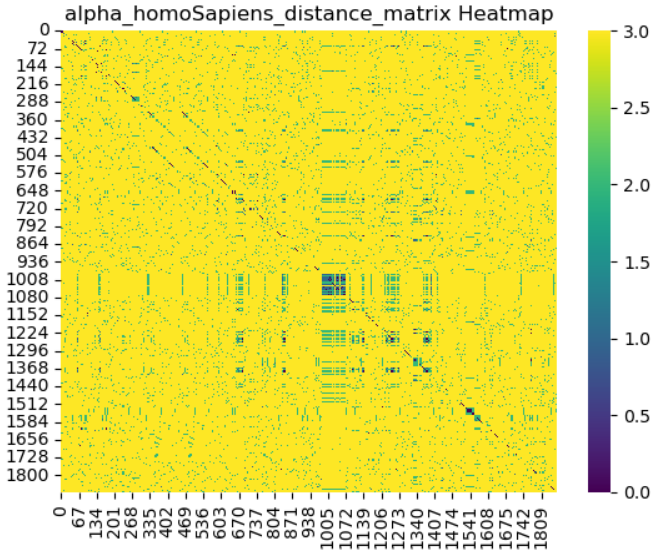

Fig. 8: human alpha chain for TCR

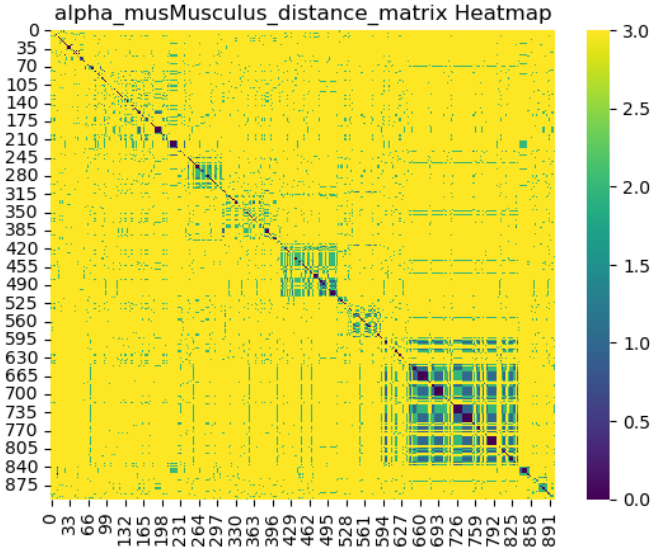

Fig. 9: mouse alpha chain for TCR

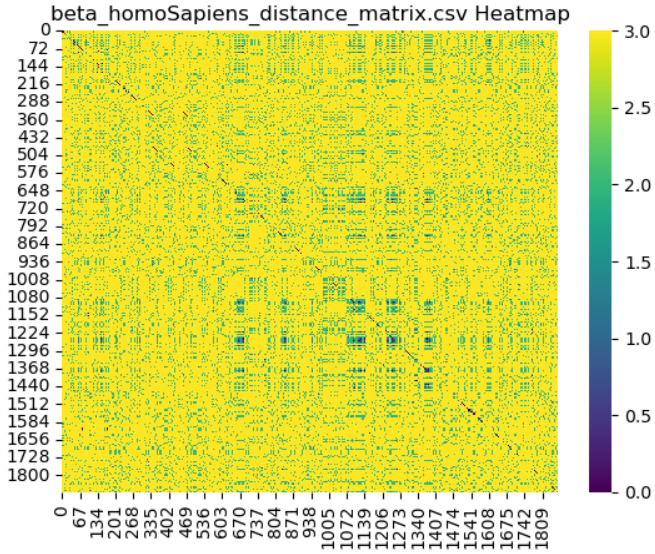

Fig. 10: human beta chain for TCR

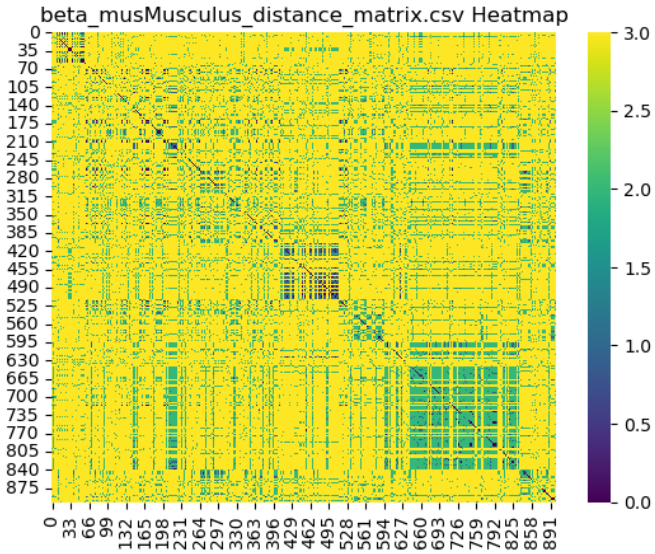

Fig. 11: mouse beta chain for TCR

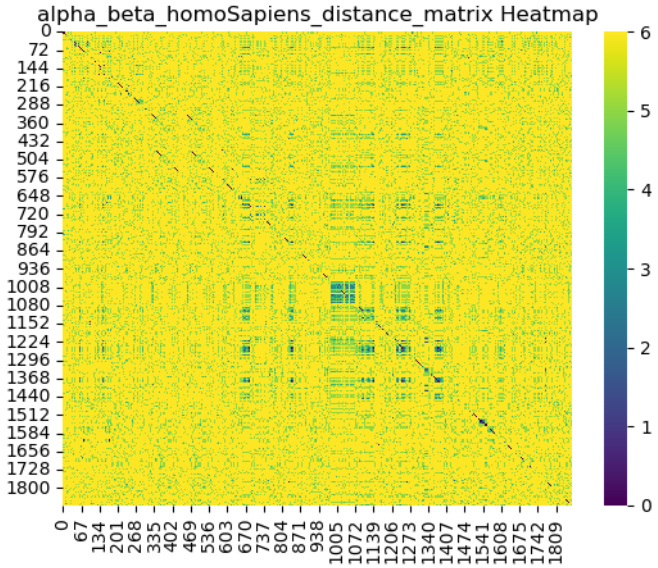

Fig. 12: human alpha beta chain for TCR

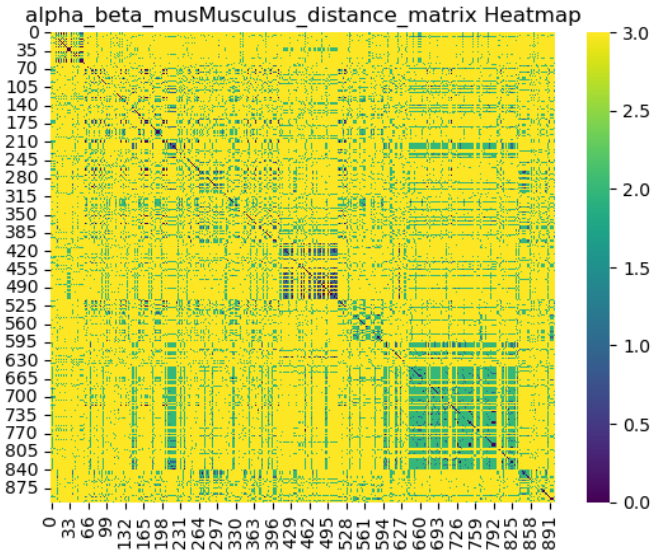

Fig. 13: mouse alpha beta chain for TCR