Advanced Statistics Assessment Report

Bioloigical Process Stage Prediction from Gene Expression Values

Daniel Sean Brown

***Abstract:***

***This project aimed to use reproducible statistical methods to build models and gain insights into how gene expression levels can be used to predict the relevant stage within a biological process. Both supervised and unsupervised learning methods were used, examining differing techniques for recognising patterns in continuous data. Further, data simulation was performed to generate insight into model performance and stability. Logistic regression yielded interpretable model coefficients for direct prediction from gene expression levels, whereas a random forest model was produced for use as a high-performance classifier. Linear discriminant analysis was also used to gain an accurate two-dimensional projection of the 20-feature wide dataset. On unsupervised techniques, a gaussian mixture model yielded accurate clustering and principal component analysis was performed to contrast with linear discriminant analysis. Clustering was also performed on simulated model coefficients.***

Table of Contents

[I. Introduction 1](#_Toc198072091)

[A. Data Profile and Context 1](#_Toc198072092)

[B. Exploratory Data Analysis 1](#_Toc198072093)

[1) Data Cleaning: 1](#_Toc198072094)

[2) Data Exploration and Visualisation: 1](#_Toc198072095)

[3) Outlier Detection and Removal: 2](#_Toc198072096)

[II. Methods 2](#_Toc198072097)

[A. Supervised Learning 2](#_Toc198072098)

[1) Logistic Regression: 2](#_Toc198072099)

[2) Random Forest Classifier: 3](#_Toc198072100)

[3) Linear Discriminant Analysis: 3](#_Toc198072101)

[B. Data Simulation 3](#_Toc198072102)

[1) Bootstrapping for Model Coefficients: 3](#_Toc198072103)

[2) Bootstrapping for Model Metrics: 3](#_Toc198072104)

[C. Unsupervised Learning 3](#_Toc198072105)

[1) Gaussian Mixed Model: 3](#_Toc198072106)

[2) Principal Component Analysis: 3](#_Toc198072107)

[3) Clustering on Simulated Data: 3](#_Toc198072108)

[III. Results 4](#_Toc198072109)

[A. Supervised Learning 4](#_Toc198072110)

[1) Logistic Regression: 4](#_Toc198072111)

[2) Random Forest Classifier: 5](#_Toc198072112)

[3) Linear Discriminant Analysis: 6](#_Toc198072113)

[B. Data Simulation 6](#_Toc198072114)

[1) Bootstrapping for Model Coefficients: 6](#_Toc198072115)

[2) Bootstrapping for Model Metrics: 6](#_Toc198072116)

[C. Unsupervised Learning 6](#_Toc198072117)

[1) Gaussian Mixed Model: 6](#_Toc198072118)

[2) Principal Component Analysis: 7](#_Toc198072119)

[3) Clustering on Simulated Data: 8](#_Toc198072120)

[IV. Discussion 8](#_Toc198072121)

[A. Supervised Learning 8](#_Toc198072122)

[1) Logistic Regression: 8](#_Toc198072123)

[2) Random Forest Classifier: 8](#_Toc198072124)

[3) Linear Discriminant Analysis 9](#_Toc198072125)

[B. Data Simulation 9](#_Toc198072126)

[C. Unsupervised Learning 9](#_Toc198072127)

[1) Gaussian Mixed Model: 9](#_Toc198072128)

[2) Principal Component Analysis: 9](#_Toc198072129)

[3) Clustering on Simulated Data: 9](#_Toc198072130)

[V. Conclusion 9](#_Toc198072131)

[VI. References 10](#_Toc198072132)

[A. QMD R Script References 10](#_Toc198072133)

# Introduction

## Data Profile and Context

The dataset consists of 3000 rows, each a measurement of 20 gene expressions recorded as a continuous value and one stage column, this being one of four values – A, B, C or D. The stages are assumed to process in this order. With little knowledge of data collection methods, data reliability should be taken at face value. The goal will then be to find relationships between state and gene expression levels and use these to make accurate predictions.

## Exploratory Data Analysis

EDA will be performed before any methods are discussed to inform on later decisions.

### Data Cleaning:

With no duplicates and only 0.67% of values being null, cleaning was quick and swift. Class balance was almost equal, ranging from 24.4% to 25.4% - this being adequate for EDA. Outlier removal is left to the end of EDA so that EDA can inform outlier removal techniques.

### Data Exploration and Visualisation:

Since 20 features and 4 target labels is not particularly unwieldly, data exploration was begun by plotting histograms for each feature and colouring by label value.

**Figure 1. Histograms for all Features by Label**

**A screenshot of a graph

AI-generated content may be incorrect.**

Here roughly gaussian distribution can be immediately noted on most features, except for differing peaks on X7, X8 and X9. This gives X8 a skew and X7, X9 some bimodality.

Following this correlation plots were used to examine feature relationships. This was approached both for the dataset as a whole and on a label-by-label basis. This was done as labels may cancel each other out or some correlations may be more pronounced.

**Figure 2. Correlation Plot across All Features**

**A graph of a graph

AI-generated content may be incorrect.**

**Figure 3. Correlation Plots for Individual Labels A graph of different values

AI-generated content may be incorrect.**

Here two ‘pockets’ of correlated features can be seen. Across labels X17, X18, X19 and X20 there is an alternating correlation between each other. Then similarly, this is the case for X7, X8, X9 and X10 - apart from for label D. The pattern shows negative correlation with directly neighbouring features and then alternating again with each further neighbour - perhaps this is best described as a checkerboard pattern. These features were explored next with density plots.

**Figure 4. Density Contour Plot for All features in First PocketA graph of contour plots

AI-generated content may be incorrect.**

**Figure 5. Density Contour Plot for All features in Second Pocket A diagram of contour plots

AI-generated content may be incorrect.**

Density plots were chosen over scatterplots due to ~3000 data points making scatterplots less clear. The first pocket yielded interesting results with clearly separable classes, except for D which had a wider spread. The same cannot be said for the second one, though correlations here were clearer.

Further on linear relationships, correlation does not pick up on non-linear dependencies and most features seem to have no or weak correlations.

A Naive Bayes model works off the assumption that features are independent, thus low performance can indicate strong feature dependencies. Though it should be noted it is not a formal test as even with a few dependent relationships a Naive Bayes model can perform satisfactorily. Regardless, having some idea of weather relationships between features exist will be useful for model construction. Below are metrics for a Naïve Bayes classifier built on the dataset:

**Table 1. Sensitivity and Specificity Metrics for NB Model**

| Label | NB model Metrics (3 D.P.) | |
| --- | --- | --- |
| Sensitivity | Specificity |
| A | 0.875 | 0.925 |
| B | 0.962 | 0.949 |
| C | 0.893 | 0.969 |
| D | 0.675 | 0.964 |

Here a high sensitivity and specificity show high performance across all labels - apart from a noticeably lower sensitivity on D, indicating issues with miss-classifying the other labels as D. This is particularity interesting in the context of the previous scatterplots indicating greater variability on D and will have to be explored further.

### Outlier Detection and Removal:

Outlier removal is approached with caution to minimise information loss. For gaussian features z scores were chosen at absolute values greater than 3.29 as this translates to 1 in 1000. Before removing outliers, they were inspected, and features were totalled for outlier count to examine if any were more prominent:

**Figure 6. Outlier Detection on Gaussian Features**

A graph with blue bars

AI-generated content may be incorrect.

61 Was deemed a safe number of outliers to remove. Some features being more prominent was also irrelevant at this scale so no further action was taken.

For X7, X8 and X9 a different method was used. Here z scores were used on a label basis with the same z score threshold of 3.29. This lent itself to 5 outliers found which were then removed. Figure is included for completeness.

**Figure 7. Outlier Detection on X7, X8 and X9**

A graph of a number of blue squares

AI-generated content may be incorrect.

Following this, classes were rebalanced by up sampling and a standardised version of the cleansed dataset was created for model construction. This was important because some models benefit greatly from data standardisation.

# Methods

## Supervised Learning

All models will use a 10-fold cross validation to reduce overfitting. As this is a multimodal classification problem, model metrics will primarily consist of sensitivity and specificity values, with some attention also given to accuracy. High sensitivity (also known as recall) reflects the model’s ability to correctly identify the given class whereas high specificity means other classes are not incorrectly classified as the given class. Focusing on these metrics will allow meaningful comparison as well insight into why accuracy may be low or high. Alternatively, F-1 score can also be used for model comparison, though in a vacuum is not as useful.

### Logistic Regression:

Multinomial logistic regression will be used initially as it is both a quick and highly interpretable model. Since classification here happens through repeated binary classification between a base class (A) and a reference class (one of B, C, D), this will yield 3 separate coefficient sets. Below is an outline of the log odds these coefficients explain:

Where: and represents the intercept and the model coefficients for respective features and relevant class.

Therefore, positive coefficient values will indicate that the respective feature increases chances of class membership and negative values will indicate the contrary.

The first model will be using the whole feature set, then a second model will be created using the five most significant features of the first model with all feature interactions added. This is done to see how feature selection impacts model performance across labels. Feature interactions are added for the same reason.

Following this L1 (Lasso), L2 (Ridge) regularisation will be performed alongside elastic net. These techniques use a λ penalty parameter to shrink coefficients. The difference being that Ridge attempts to shrink all coefficients but will keep them, whereas Lasso will shrink some coefficients to zero. Elastic net is a mixture of the two and uses an α parameter to find the most optimal balance of the two. Note that at α = 0 Ridge is performed and at α = 1 the model performs Lasso. Meaning if Elastic Net performs best then 0 < α < 1. The highest performing model will then be further analysed to better answer the research question.

### Random Forest Classifier:

Random Forests were chosen as they can capture non-linear relationships which logistic regressors can’t. Having already tried a Naïve Bayes model and due to the high dimensionality of the problem, K Nearest Neighbour and NB were not selected. Support Vector Machines could be a viable alternative here.

Random Forests work by generating Decision Trees on sampled dataset and feature space. This means that they are less prone to overfitting and are a generally high performing. The agglomeration of Decision Trees are however more difficult to analyse on the whole.

Though less interpretable, feature importance will be analysed using Gini decrease, this refers to the amount of information gained by splitting decision trees on a given feature. Having said that, the main goal with this model will be maximising performance to gain a powerful classifier after having understood features.

Two models will be trained, the first performing classification on all labels with all features and a second focusing on classifying between D and non-D. Feature importance can then be compared to examine where how D can be classified better. The motivation for this coming from initial observations suggesting D will be more difficult to classify.

### Linear Discriminant Analysis:

This method was selected primarily to produce a two-dimensional projection of the dataset and to give comparison with PCA (Principal Component Analysis) results. The motivation behind this being that while there are correlations in the dataset, they are not abundant and PCA works best with correlated features. LDA works by maximising variance between classes and minimising it within classes, thus being useful for multiclass problems. It being a supervised learning method, performance will still be analysed to gain confirmation of projection validity. If required, the produced visual can also be used for clustering.

## Data Simulation

Data simulation will be performed through bootstrapping, meaning repeated sampling with replacement. Each sample will then be used for further data generation.

### Bootstrapping for Model Coefficients:

For each sample a logistic regressor will be trained with a simple train/test split due to computing limitations. Then model coefficients will be stored for each model to create a dataset which will be investigated further by clustering.

### Bootstrapping for Model Metrics:

Each model’s accuracy metric will also be extracted and aggregated to examine spread and thus give further credence to findings during logistic regression – that is if spread is low.

## Unsupervised Learning

### Gaussian Mixed Model:

GMM Was selected over K-Means or Hierarchical clustering due to Gaussian distribution on most features. Clustering will be examined for 2 – 6 clusters to see how labels group depending on weather they are squeezed, fit or stretched into clusters.

### Principal Component Analysis:

PCA will be used to get a two-dimensional projection of the dataset and will be contrasted by the results of LDA. PCA Works best with correlated features, which are somewhat present. Should cumulative variance be below 50% on the first two PCs, PCA will be attempted again with selected features.

### Clustering on Simulated Data:

GMM will be used again to cluster on simulated coefficients, here coefficient groupings will be examined between labels. Gaussian distribution on coefficients here is assumed, not known. Therefore results may either support or go to against this assumption.

# Results

Below are all the figures illustrating results. They are discussed and figures are referenced in the discussion section. Some visualisations from the QMD file were not included in the report as they were not deemed informative or useful.

## Supervised Learning

### Logistic Regression:

**Table 2. Model Metrics for Logistic Regression Model**

| Label | Logistic Regressor Model Metrics (3 D.P.) | |
| --- | --- | --- |
| Sensitivity | Specificity |
| A | 0.812 | 0.926 |
| B | 0.906 | 0.949 |
| C | 0.933 | 0.975 |
| D | 0.604 | 0.902 |

**Figure 8. Initial Logistic Model Coefficients**

**A graph with multiple colored lines

AI-generated content may be incorrect.**

**Figure 9. Summed Coefficient Values by Feature.**

**A graph of a number of blue bars

AI-generated content may be incorrect.**

**Table 3. Model Metrics for Second Logistic Regression Model**

| Label | Top 5 Features Logistic Regressor Model Metrics (3 D.P.) | |
| --- | --- | --- |
| Sensitivity | Specificity |
| A | 0.913 | 0.904 |
| B | 0.926 | 0.980 |
| C | 0.993 | 0.964 |
| D | 0.571 | 0.953 |

**Table 4. Regularisation Technique Metric Comparisons**

| Model | Regularisation Comparison (3 D.P.) | |
| --- | --- | --- |
| Accuracy | Sensitivity on D |
| Lasso | 0.856 | 0.671 |
| Ridge | 0.856 | 0.591 |
| Elastic Net | 0.856 | 0.671 |

**Figure 10. Heatmap of Coefficients after Lasso Regularisation**

**A screenshot of a graph

AI-generated content may be incorrect.**

**Figure 11. Coefficient Path on Lasso Regularisation for Classifying A**

**A graph of different colored lines

AI-generated content may be incorrect.**

**Figure 12. Coefficient Path on Lasso Regularisation for Classifying B**

**A graph of a line graph

AI-generated content may be incorrect.**

**Figure 13. Coefficient Path on Lasso Regularisation for Classifying C**

**A graph of different colored lines

AI-generated content may be incorrect.**

**Figure 14. Coefficient Path on Lasso Regularisation for Classifying D**

**A graph of different colored lines

AI-generated content may be incorrect.**

### Random Forest Classifier:

**Table 5. Model Metrics for Random Forest Classifier**

| Label | RFC Model Metrics (3 D.P.) | |
| --- | --- | --- |
| Sensitivity | Specificity |
| A | 0.953 | 0.975 |
| B | 0.973 | 0.989 |
| C | 1.000 | 0.993 |
| D | 0.873 | 0.975 |

**Figure 15. Feature Importance on Random Forest Model (scaled)**

**A graph with numbers and a bar chart

AI-generated content may be incorrect.**

**Figure 16. High Importance Feature Plots for Label and Feature Means**

**A graph of different colored lines

AI-generated content may be incorrect.**

**Figure 17. Medium Importance Feature Plots for Label and Feature Means**

**A graph of different colored lines

AI-generated content may be incorrect.**

**Figure 18. Low Importance Feature Plots for Label and Feature Means**

**A graph of different colored lines

AI-generated content may be incorrect.**

**Table 6. D vs Non-D Random Forest Model Metrics**

| D vs Not D RFC Model Metrics (3 D.P.) | | |
| --- | --- | --- |
| ***Accuracy*** | Sensitivity | Specificity |
| 0.935 | 0.826 | 0.971 |

**Figure 19. D vs Non-D Random Forest Feature Importance**

**A graph with numbers and a bar

AI-generated content may be incorrect.**

**Figure 20. Random Forest Feature Importance Comparison by Classification Target**

**A graph of different colored bars

AI-generated content may be incorrect.**

### Linear Discriminant Analysis:

**Table 7. LDA Model Metrics**

| Label | LDA Model Metrics (3 D.P.) | |
| --- | --- | --- |
| Sensitivity | Specificity |
| A | 0.846 | 0.915 |
| B | 0.973 | 0.933 |
| C | 0.940 | 0.969 |
| D | 0.550 | 0.953 |

**Figure 21. LDA Projection Using Contour Plot**

**A graph with colored lines

AI-generated content may be incorrect.**

## Data Simulation

### Bootstrapping for Model Coefficients:

This section has no figures, see clustering to see results of work on simulated model coefficients.

### Bootstrapping for Model Metrics:

**Table 8. Bootstrapped Accuracy Values**

| Bootstrapped Logistic Model Accuracy Aggregates (3 D.P.) | |
| --- | --- |
| Metric | Value |
| Min | 0.8093 |
| 1st Quartile | 0.818 |
| Median | 0.824 |
| Mean | 0.826 |
| 3rd Quartile | 0.835 |
| Max | 0.857 |
| Standard Deviation | 0.012 |

## Unsupervised Learning

### Gaussian Mixed Model:

**Figure 22. Heatmaps of GMM Clustering Results with Classification Given as a Percentage**

**A blue and white squares with labels

AI-generated content may be incorrect.**

**Figure 23. GMM Cluster for G=4 Projected on PC1 and PC2 from PCA**

**A graph of different colored dots

AI-generated content may be incorrect.**

### Principal Component Analysis:

**Table 9. PCA Variance Results**

| PC | PCA Variance Summary for First 4 Principal Components (3 D.P.) | |
| --- | --- | --- |
| Proportion of Variance | Cumulative Variance |
| 1 | 0.147 | 0.147 |
| 2 | 0.143 | 0.290 |
| 3 | 0.081 | 0.372 |
| 4 | 0.059 | 0.431 |

**Figure 24. PCA Scree Plot**

**A graph with numbers and lines

AI-generated content may be incorrect.**

**Figure 25. Feature Contribution to Each PC**

**A grid with numbers and dots

AI-generated content may be incorrect.**

**Figure 26. Contour Plot on PC1 and PC2**

**A graph showing a contour of a graph

AI-generated content may be incorrect.**

**Table 10. PCA Variance Results for Second PCA Attempt**

| PC | PCA Variance Summary for First 4 Principal Components for Second PCA Attempt (3 D.P.) | |
| --- | --- | --- |
| Proportion of Variance | Cumulative Variance |
| 1 | 0.434 | 0.434 |
| 2 | 0.211 | 0.646 |
| 3 | 0.142 | 0.788 |
| 4 | 0.109 | 0.897 |

**Figure 27. PCA Scree Plot for Important Features by RFC**

**A graph with numbers and a line

AI-generated content may be incorrect.**

**Figure 28. Feature Contribution to Each PC in Second PCA**

**A grid with circles and numbers

AI-generated content may be incorrect.**

**Figure 29. Feature Contribution to First Two PCs on Second PCA**

**A graph of different sizes and numbers

AI-generated content may be incorrect.**

**Figure 30. Contour Plot of PC1 and PC2 on Second PCA**

**A graph of colored lines

AI-generated content may be incorrect.**

### Clustering on Simulated Data:

**Figure 31. Confusion Matric Heatmap for Clustering on Simulated Coefficients.**

**A graph of blue squares

AI-generated content may be incorrect.**

**Figure 32. Plot of Clusters from GMM on Simulated Coefficients**

A graph with different colored circles

AI-generated content may be incorrect.

# Discussion

## Supervised Learning

### Logistic Regression:

Initial model saw high performance; this only being let down by low sensitivity on D (**Table 2**). Coefficients were extracted which can then be used to produce log odds equations directly and visualisations showed which features played a larger role in classification (**Figures 8 and 9**). Interestingly, the first grouping of correlated features all showed up as being significant.

Learning model on selected features with interactions saw slight improvement on all model metrics and a noticeable further drop on sensitivity for D (**Table 3**). The best five features elected were exactly the first set of correlated features found during EDA. These all had strong correlations in A, B and C but not in D. This could explain the drop in performance for the label. Regardless, D rely on the feature set as whole more than the other labels for correct classification. It also shows that the selected best five features are sufficient for classification – ignoring D.

Regularisation was insightful, with Lasso regression performing the best but only by a small margin (**Table 4**). Observing which coefficients were shrunk, especially to zero, performs feature selection (**Figure 10**). Again it is interesting that the second grouping of correlated features all perform badly here. Also, A, B and C each have a standout feature with a more significant coefficient. These tend to all be from the first grouping and tend to have at least one label where they are less impactful. D is not dominated by one single feature, perhaps lending itself to the label being less separable than the others in any one given feature. This tendency for other features is backed up by the coefficient paths plot, (**Figure 14**) where the best feature (X9) is shrunk to zero before other supporting features are. This contrasts well with the paths observed on the other labels (**Figures 11, 12 and 13**) where dominant features are pronounced and shrunk last.

### Random Forest Classifier:

Model performance here was almost perfect having overcome the sensitivity issues with D (**Table 5**). Random forests use of both bootstrapping and bagging and the 10-fold cross validation used should be sufficient to reduce overfitting. Meaning this is a model which can be reliable used to predict on new data.

For answering the research question however, the feature importance graphs are perhaps the most informative figures in the whole report. After feature importance was extracted (**Figure 12**), features were binned into three groups of importance and then these groupings were visualised to see how feature means change across labels (**Figures 16, 17, and 18**). Line ribbons were also added to indicate spread. On impactful features it can clearly be seen why these aid the most in classification, observed from easily separable points.

Further, it can be seen on the whole that most features start at a middling level at A then peak at B, trough at C then return to a similar middling level at D. Perhaps this lends itself to why A has consistently lower sensitivity than B and C. It can also be observed that points are at their most similar and spread at D and why X8 is so significant for classifying on C. X8 not only takes a different path to the other features, within it A, B and D are grouped together, and C is separate.

Moreover, the visual serves well to inform why exactly some features are less useful than others with a clear pattern as descent in importance is observed. Lines become inseparable and spread increases.

Again, I would claim these figures are the key to understanding how gene expressions change by stage and which genes can be used to predict the current stage.

On binary classification with D and non-D, again note high model performance (**Table 6**). Here it can be seen that importance order is roughly maintained, but still some features dominate less (**Figure 19**). It is also worth noting the dramatic decrease in importance for X8 and X7. Next on direct comparison it is again seen that D relies on the feature set as whole more (**Figure 20**). This supports findings from logistic regression about D feature importance.

### Linear Discriminant Analysis

Here model metrics are like previous models with low sensitivity on D (**Table 7**). Regardless, the visual is still trustworthy for the most part and serves well to reinforce why classifications on C and B are clear, slightly less so on A and why D is sometimes incorrectly misclassified (**Figure 21**). Due to D’s larger spread, the most reliable classification of D likely occurs in areas where although points are sparse, the only present ones are from D. This idea is illustrated here well – especially with D being made up of a single wide contour having no visible peak.

## Data Simulation

Agglomerated accuracy had tiny variance, which reinforced confidence in the results from logistic regression (**Table 8**). Further, coefficients were successfully generated, their analysis remains to occur during clustering. It is worth noting though that this part of the code takes significantly longer to execute than other parts of the script.

## Unsupervised Learning

### Gaussian Mixed Model:

GMM performed clustering near perfectly. This retrospectively strengthens the gaussian distribution claimed on most features. The order of label clustering as cluster numbers were increased would indicate most distinct label is C, followed by correctly clustered D values and then A and B (**Figure 22**). Notice also that only real errors came from clustering some D labels as A. This reinforces previous findings surrounding overlap between the two labels. Plot is also included of clustering for completeness (**Figure 23**). Note that projection demonstrating clustering comes from the *fviz\_cluster* function in R automatically performing PCA for plotting.

### Principal Component Analysis:

Initial PCA success was limited. With low variance capture on the first two principal components (**Table 9 and Figure 24**), majority of data was not explained within the produced 2D plot (**Figure 26**). Further, it can be seen how correlated features are grouped and then other feature contribution to PCs were sparse and limited (**Figure 25.**).

PCA on selected features was more successful with adequate variance capture (**Table 10 and Figure 27**). Feature contribution here is also more interesting (**Figures 28 and 29**) showing more varied feature use. The produced visual is informative, demonstrating clear separation between C and B with A as an intermediary, which also has the most overlap with the sparse D (**Figure 30**). This agrees with previous findings regarding label behaviour.

Having said that though, I would advise to use LDA results instead for dataset projection as variance here is still low and is only for selected features – even though the discarded features consistently ranked low in importance.

### Clustering on Simulated Data:

GMM performed clustering perfectly and visualisation shows both high density within clusters and large voids between them (**Figures 31 and 32**). This supports results from initial logistic regression, demonstrating model coefficients are stable. It also goes to show that models for varying labels are distinct.

# Conclusion

Overall, data was successfully interpreted, and engaging visuals were produced to decipher the underlying patterns within. It was understood how and which gene expressions change by label; thus desired outcomes were achieved.

Most performance issues arose surrounding low specificity on D. These were recognised, understood and overcome using model analysis and further model construction.

For improvement, further supervised learning models could be examined, for example support vector machines or gradient boosting machines. I do however stand by the decision to use LDA as the produced projection was much better than was gained from PCA and would argue this is the best single visual for understanding the dataset. It shows exactly why D is more difficult to classify and why most misclassification around A comes from D. The separability of A, B and C is also demonstrated here nicely with distinct, dense and clear groupings.

Final important takeaways would be lasso regularisation coefficients for direct model equation construction, random forest model for high performance classification and finally random forest feature importance mean by label plots. These plots serve as the best visual for how individual gene expression levels change throughout the biological process as well as demonstrate why some features were selected by the models as being more useful.

# References

## QMD R Script References

See accompanying QMD file for code lines being referenced. References are also listed there for completeness. Note that where an AI tool is referenced, it was used to assist in bridging coding solutions. The work there is still original, its use was for coming up with initial ideas which could then be developed independently or with the assistance of other also referenced sources.

[1] - Bobbitt, Z. (2022) *How to Use pivot\_longer() in R*. Statology. Available at: <https://www.statology.org/pivot_longer-in-r/> (Accessed: 5 May 2025).

[2] - Bobbitt, Z. (2021) 'How to Use facet\_wrap() in R', Statology. Available at: <https://www.statology.org/facet_wrap/> (Accessed: 6 May 2025).

[3] - Wilke, C.O. (2024) 'Introduction to cowplot', *cowplot*, 22 January. Available at: <https://wilkelab.org/cowplot/articles/introduction.html> (Accessed: 3 May 2025).

[4] - Bobbitt, Z. (2020) 'K-Fold Cross Validation in R (Step-by-Step)', *Statology*. Available at: <https://www.statology.org/k-fold-cross-validation-in-r/> (Accessed: 11 May 2025).

[5] - Soetewey, A. (2020) 'Outliers Detection in R', *Stats and R*. Available at: <https://statsandr.com/blog/outliers-detection-in-r/> (Accessed: 19 April).

[6] - R-bloggers (2019) 'Methods for dealing with imbalanced data', *R-bloggers*. Available at: <https://www.r-bloggers.com/2019/04/methods-for-dealing-with-imbalanced-data/> (Accessed: 19 April 2025).

[7] - OpenAI (2025) *ChatGPT*, Version GPT-4. Available at: <https://openai.com/chatgpt> (Accessed: 11 May 2025).

[8] - Anquandah, J. (2025) 'Logistic\_Regression.qmd', Available at: <https://blackboard.uwe.ac.uk/ultra/courses/_365666_1/cl/outline> (Accessed: 22 April 2025).

[9] - GeeksforGeeks (2025) 'How to do nested cross-validation with LASSO in caret or tidymodels?', *GeeksforGeeks*. Available at: <https://www.geeksforgeeks.org/how-to-do-nested-cross-validation-with-lasso-in-caret-or-tidymodels/> (Accessed: 1 May 2025).

[10] - Anquandah, J. (2025) 'Simulation.qmd', Available at: <https://blackboard.uwe.ac.uk/ultra/courses/_365666_1/cl/outline> (Accessed: 7 May 2025).

[11] - GeeksforGeeks (2024) 'Building a Random Forest with Caret', *GeeksforGeeks*. Available at: <https://www.geeksforgeeks.org/building-a-randomforest-with-caret/> (Accessed: 28 Apr 2025).

[12] - Wickham, H. (2025) 'geom\_ribbon: Ribbons and area plots in ggplot2', *ggplot2*, Available at: https://ggplot2.tidyverse.org/reference/geom\_ribbon.html (Accessed: 28 Apr 2025).

[13] - Bobbitt, Z. (2020) 'Linear Discriminant Analysis in R (Step-by-Step)', *Statology*. Available at: <https://www.statology.org/linear-discriminant-analysis-in-r/> (Accessed: 5 May 2025).

[14] - Anquandah, J. (2025) 'Hierarchical Clustering.qmd', Available at: <https://blackboard.uwe.ac.uk/ultra/courses/_365666_1/cl/outline> (Accessed: 9 May 2025).

[15] - GeeksforGeeks (2025) 'What is Gaussian Mixture Model Clustering Using R?', *GeeksforGeeks*. Available at: https://www.geeksforgeeks.org/what-is-gaussian-mixture-model-clustering-using-r/ (Accessed: 9 May 2025).

[16] - Anquandah, J. (2025) 'PCA.qmd', Available at: <https://blackboard.uwe.ac.uk/ultra/courses/_365666_1/cl/outline> (Accessed: 8 May 2025).