# 1 First data exploration

I first stared to explore the data By doing:

- Define your labels/response in a categorical manner (high response, low response, late response,..) based on the behaviour of the antibody titers of these individuals. This is what is usually done, or you can keep the antibodies as numbers and try and predict the numerical value.

Measles:

* Antybody titer data
* Cytokine data
* Cytometry data

The initial split I made was divide the samples up in no response (29) and response (11).

A graph with a red line and blue dots

Description automatically generatedA graph showing a number of different colored squares

Description automatically generated

I first tried to do a Principal Component Analysis to determine what features I could use in a model. In this analysis, I performed Principal Component Analysis (PCA) on the full

set of features (so all the data I initially got Antibody titer, Cytokine and Cytometry data):

A close-up of a text

Description automatically generated

gives the following distribution

A graph with different colored bars

Description automatically generated

I than did a PCA Analysis.

To determine an optimal number of components, I examined the explained

variance plot. I observed that the first 10 components captured a majority

of the variance. Including additional dimensions beyond the first 10 principal

components likely captures less relevant variance, as these components explain

progressively smaller portions of the data’s structure.

A graph with a line

Description automatically generated

For feature selection, I applied a loading threshold of 0.3 to identify the most

prominent features in each principal component. The threshold helps capture

features with significant contributions while filtering out minor influences. The selected top features per component, based on this threshold, were as follows:

A list of components with text

Description automatically generated

The final set of selected features was:



Than I performed cross-validation on the selected features using a Random Forest

classifier with a balanced accuracy scoring metric. The balanced accuracy score

was chosen to account for any class imbalances in the dataset.

* **Performance Metrics**:
* Full Feature Set Mean CV Balanced Accuracy: 0.4550
* Full Feature Set Mean CV Accuracy: 0.6524
* Reduced Feature Set Mean CV Balanced Accuracy: 0.4150
* Reduced Feature Set Mean CV Accuracy: 0.5905
* Full Feature Set Test Balanced Accuracy: 0.5000
* Full Feature Set Test Accuracy: 0.7500
* Reduced Feature Set Test Balanced Accuracy: 0.5000
* Reduced Feature Set Test Accuracy: 0.7500
* Full Feature Set Single Class Prediction: Yes
* Reduced Feature Set Single Class Prediction: Yes

The current analysis demonstrates that both the full and reduced feature sets fail to provide meaningful predictive performance, as evidenced by balanced accuracy scores around random guessing and the models’ tendency to predict only the majority class. This outcome highlights critical issues related to class imbalance and possibly insufficient or non-discriminative features.

Than I inspected the loadings. Notably, in the first principal component (PC1), no features met the threshold of 0.3. However, several features, such as IL17A, IFNg, IL12-p70, TGF-a, and IL-2, had loadings around 0.24 , which are still relatively high and suggest these features are correlated and collectively contribute to the variance captured by PC1.

A table with numbers and text

Description automatically generated

These features (IL17A, IFNg, IL12-p70, TGF-a, IL-2) have similar loadings,

suggesting they contribute collectively to the primary variance captured by PC1.

I remembered that these features are likely correlated, a correlation analysis

could further help understand their relationships

A chart with red squares and black text

Description automatically generated

The clustered correlation heatmap (Figure 3) reveals the relationships between

features in the dataset, highlighting groups of features with high correlations.

Based on this heatmap, I identified several feature groups that show distinct

clustering patterns, which may represent shared biological or measurement prop-

erties. These groups are as follows (Explanations are AI-generated by GPT-4o and than checked with my basic knowledge, as I am not fully familiar with the specific functions of these features.):

* Group 1: {IL1Ra, TNFb, MCP3, IL-13} This group includes inflammatory markers and interleukins, which are involved in immune response processes and exhibit moderate to strong correlations with each other.
* Group 2: {IL-9, IL-10, GM-CSF, IL-2, IL-7, IFNa2} Group 2 is composed of cytokines and growth factors that play roles in immune modulation and cell signaling. Their high inter-correlation suggests that they may collectively influence immune responses.
* Group 3: {Gender, RBC Day 0, HGB Day 0, HCT Day 0} This group includes demographic and hematological variables. Their clustering reflects physiological measures, indicating they likely share underlying biological relationships related to blood metrics.
* Group 4: {IL12-p70, IFNg, IL17A, TGF-a, TNFa, MIP1b, FGF-2, VEGF} Group 4 contains a mixture of cytokines, growth factors, and angiogenic markers. These features are associated with inflammation, cellular growth, and vascular health, indicating potential joint involvement in inflammatory or immune-driven pathways.
* Group 5: {Fractalkine, IL-15, IL-1b} This smaller group includes signaling proteins that play roles in cell recruitment and inflammatory response. The correlations within this group suggest a shared function related to immune cell recruitment and activation.

To address the challenge of high dimensionality and multicollinearity among the

predictor variables, I used Partial Least Squares Regression (PLSR). PLSR is

particularly effective when dealing with datasets that have a large number of

predictors, especially when those predictors are highly collinear. Unlike Princi-

pal Component Analysis (PCA), which focuses solely on capturing the variance

within the predictor variables, PLSR identifies components that not only ac-

count for the variance in the predictors but also maximize the covariance with

the response variable. This means that the components derived from PLSR are

more directly related to the outcome we aim to predict, making it a powerful

tool for dimensionality reduction in predictive modeling. By using PLSR, I

compressed the feature space into a smaller set of latent variables that retain

the most relevant information for predicting the outcome, while also mitigating

the effects of multicollinearity among the predictors.

Following the approach from the previous analysis, I selected the first 10 princi-

pal components for further examination. These components capture a significant

portion of the total variance in the data.

A graph showing the number of components

Description automatically generated

To identify the most influential features within each principal component, I

applied a loading threshold of 0.33. This slightly higher threshold compared to the previous analysis (where a threshold of 0.3 was used) was chosen to focus

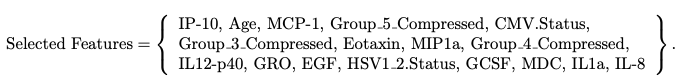
on features with stronger contributions to the principal components, potentially

enhancing the robustness of the feature selection. The selected top features per component, based on this threshold, were as follows:

A screenshot of a computer

Description automatically generated

The final set of selected features was:



I performed cross-validation on the selected features using a Random Forest

classifier with a balanced accuracy scoring metric. The balanced accuracy score

was chosen to account for any class imbalances in the dataset.

* **Performance Metrics:**
* Full Feature Set Mean CV Balanced Accuracy: 0.5300
* Full Feature Set Mean CV Accuracy: 0.7190
* Reduced Feature Set Mean CV Balanced Accuracy: 0.4400
* Reduced Feature Set Mean CV Accuracy: 0.6238
* Full Feature Set Test Balanced Accuracy: 0.5000
* Full Feature Set Test Accuracy: 0.7500
* Reduced Feature Set Test Balanced Accuracy: 0.5000
* Reduced Feature Set Test Accuracy: 0.7500
* Full Feature Set Single Class Prediction: Yes
* Reduced Feature Set Single Class Prediction: Yes

It is evident that the initial approach of reducing the feature set through Principal Component Analysis (PCA) did not yield the intended improvements in model performance. Specifically, both the full and reduced feature sets resulted in a balanced accuracy of 0.5000 on the test set, which aligns with the performance of random guessing, especially considering a balanced accuracy of 0.5 represents no discriminative power beyond chance. Additionally, the overall accuracy remained consistent at 0.7500 for both feature sets on the test set, likely reflecting the underlying class distribution rather than genuine predictive capability.

Furthermore, the cross-validation results reinforce these observations. The Full Feature Set achieved a mean CV balanced accuracy of 0.5000 and a mean CV accuracy of 0.7190, while the Reduced Feature Set scored a mean CV balanced accuracy of 0.4400 and a mean CV accuracy of 0.6286. These balanced accuracy scores indicate that neither feature set provided the model with sufficient discriminatory information to improve upon random chance. The reduced feature set, in fact, performed slightly worse in terms of balanced accuracy, suggesting that the feature selection process may have inadvertently omitted important variables necessary for distinguishing between classes.

In conclusion, the current results demonstrate that dimensionality reduction via PCA did not enhance the model’s predictive performance. Both feature sets failed to achieve balanced accuracy above the random baseline, and the tendency of the models to predict only a single class highlights the need for addressing class imbalance more effectively. Moving forward, I will use strategies such as resampling techniques (e.g., SMOTE for oversampling the minority class), class weighting adjustments within the classifier, and exploring alternative modeling approaches that are more robust to imbalanced data. Additionally, revisiting the feature selection process to ensure that critical predictive features are retained, possibly incorporating supervised feature selection methods that directly consider the relationship between features and the target variable, may help in improving model performance.

LATER THOUGHTS WHEN WRITING DOWN MY RESULTS AND SUGESTIONS FROM FABIO.

PCA is an unsupervised dimensionality reduction technique that focuses solely on maximizing the variance within the predictor variables without considering their relationship to the target variable (this is something I understand way better now than before I stared). We saw that the first principal component (PC1) did not yield any features meeting the loading threshold of 0.3, and several key features had loadings around 0.24. This suggests that while these features are correlated and contribute to the variance captured by PC1, they do not individually provide strong predictive power for distinguishing between response categories. Fabio observed that setting a hard threshold might not be the best approach reinforces the notion that PCA

So given the limitations of PCA in this scenario, transitioning to supervised dimensionality reduction methods like sparse Partial Least Squares (sPLS) or Regularized Canonical Correlation Analysis (RCCA) could offer advantages.

Again using:

The initial set of features included like before (see page 1):

* **Clinical Variables:** ‘Vaccinee’, ‘Gender’, ‘Age’, ‘WBC Day 0’, ‘RBC Day 0’, ‘HGB Day 0’, ‘HCT Day 0’, ‘PLT Day 0’, ‘%LYM Day 0’, ‘%MON Day 0’, ‘%GRA Day 0’.
* **Viral Status Indicators:** ‘CMV.Status’, ‘EBV.Status’, ‘HSV1\_2.Status’, ‘HHV6.Status’.
* **Cytokines and Growth Factors:** ‘EGF’, ‘FGF-2’, ‘Eotaxin’, ‘TGF-a’, ‘GCSF’, ‘Flt3 Ligand’, ‘GM-CSF’, ‘Fractalkine’, ‘IFNa2’, ‘IFNg’, ‘GRO’, ‘IL-10’, ‘MCP3’, ‘IL12-p40’, ‘MDC’, ‘IL12-p70’, ‘IL-13’, ‘IL-15’, ‘sCD40L’, ‘IL17A’, ‘IL1Ra’, ‘IL1a’, ‘IL-9’, ‘IL-1b’, ‘IL-2’, ‘IL-5’, ‘IL-6’, ‘IL-7’, ‘IL-8’, ‘IP-10’, ‘MCP-1’, ‘MIP1a’, ‘MIP1b’, ‘TNFa’, ‘TNFb’, ‘VEGF’.
* **Response Labels:** ‘response\_label’ indicating whether the individual was a responder (‘response’) or non-responder (‘no response’).

**1 PLS the full set of features.**

Using the full set of features, the PLS regression model was constructed to identify the most significant predictors of the response variable. The key results from this initial analysis are as follows:

* **Correlation Between Components:**
  + **Component 1:** X-Y Variates 1 Correlation of **0.49**
  + **Component 2:** X-Y Variates 2 Correlation of **0.57**
* **Top Contributing Features:**
  + **Component 1 (Top 10%):** [‘HHV6.Status’, ‘IL-9’, ‘IL-8’, ‘HGB Day 0’, ‘%LYM Day 0’]
  + **With Loading Threshold:** 0.2039
  + **Component 2 (Top 10%):** [‘FGF-2’, ‘TGF-a’, ‘TNFa’, ‘VEGF’, ‘%MON Day 0’]
  + **With Loading Threshold:** 0.2219

These might play an important role in differentiating between responders and non-responders.

A screenshot of a graph

Description automatically generatedA graph with numbers and circles

Description automatically generated with medium confidence

**2 PLS Reduced set of features.**

During the analysis, concerns were raised by Fabio regarding the quality of certain cytokine measurements. Specifically, some cytokines exhibited minimal deviation from their limits of detection (LOD), leading to potential inconsistencies and unreliable measurements. To address this Cytokines with at least one sample falling below the LOD were identified and subsequently removed from the feature set.

After excluding the problematic cytokine features, the PLS regression was re-executed to assess the impact of this data quality improvement. The updated results are as follows:

* **Correlation Between Components:**
  + **Component 1:** X-Y Variates 1 Correlation of **0.56**
  + **Component 2:** X-Y Variates 2 Correlation of **0.48**
* **Top Contributing Features:**
  + **Component 1 (Top 10%):** [‘HHV6.Status’, ‘IL-8’, ‘HGB Day 0’, ‘%LYM Day 0’]
  + **With Loading Threshold:** 0.2616
  + **Component 2 (Top 10%):** [‘FGF-2’, ‘TNFa’, ‘VEGF’, ‘%MON Day 0’]
  + with **Loading Threshold:** 0.3194

The removal of certain cytokines did not really alter the inial findings. Key features such as ‘HHV6.Status’, ‘IL-8’, ‘HGB Day 0’, and ‘%LYM Day 0’ continued to emerge as significant contributors to **Component 1**. For **Component 2**, ‘FGF-2’, ‘TNFa’, ‘VEGF’, and ‘%MON Day 0’ remained prominent.

A screenshot of a graph

Description automatically generatedA graph with numbers and circles

Description automatically generated with medium confidence

**3 PLS Compressed set of features.**

In the initial stages of data exploration, a clustered correlation heatmap (Figure 3) was generated using the Ward’s method (specifically Ward’s linkage with Euclidean distance, often referred to as Ward2). This hierarchical clustering approach grouped highly correlated features into distinct clusters based on their interrelationships. The identified groups were as follows:

* Group 1: {IL1Ra, TNFb, MCP3, IL-13}
* Group 2: {IL-9, IL-10, GM-CSF, IL-2, IL-7, IFNa2}
* Group 3: {Gender, RBC Day 0, HGB Day 0, HCT Day 0}
* Group 4: {IL12-p70, IFNg, IL17A, TGF-a, TNFa, MIP1b, FGF-2, VEGF}
* Group 5: {Fractalkine, IL-15, IL-1b}

To address multicollinearity and reduce dimensionality, Principal Component Analysis (PCA) was employed within each of these clusters to compress the correlated features into principal components. The expectation was that features within the same group would cluster together in the resulting Sparse Partial Least Squares (sPLS) analysis, maintaining the inherent group structures observed in the correlation heatmap.

Upon performing sPLS with the PCA-compressed features, it was observed that the compressed features from Group 2 and Group 4 did not exhibit the expected clustering behavior. Specifically, these groups did not remain distinctly clustered in the sPLS components, which was unexpected given their clear separation in the hierarchical correlation

heatmap.

This is probably because PCA is an unsupervised dimensionality reduction technique that focuses solely on capturing the maximum variance within the data without considering any relationship to the target variable. It transforms correlated features into a set of uncorrelated principal components based on variance.While sPLS, on the other hand, is a supervised method that seeks to find components that not only capture variance in the predictor variables but also maximize the covariance with the response variable. This dual objective can lead to different component structures compared to PCA.

A screenshot of a graph

Description automatically generatedA graph with red and black text

Description automatically generated with medium confidence

I hightlited the original projection and the compressed projection:

A screenshot of a graph

Description automatically generated

the PLS regression was re-executed to assess the impact of this data quality improvement. The updated results are as follows:

* **Correlation Between Components:**
  + **Component 1:** X-Y Variates 1 Correlation of **0.60**
  + **Component 2:** X-Y Variates 2 Correlation of **0.55**
* **Top Contributing Features:**
  + **Component 1 (Top 10%):** [‘HHV6.Status’, ‘IL-8’, ‘%LYM Day 0’]
  + **With Loading Threshold:** 0.2794
  + **Component 2 (Top 10%):** ['%MON Day 0', 'group\_3\_Compressed\_1', 'group\_4\_Compressed\_1']
  + with **Loading Threshold:** 0.3132

Test the features. I took the features form **Component 1 and Component 2.**

I performed cross-validation on the selected features using a Random Forest

classifier with a balanced accuracy scoring metric. The balanced accuracy score

was chosen to account for any class imbalances in the dataset. (no response: 29 samples, response: 11 samples).

These are the results:

|  |  |  |  |
| --- | --- | --- | --- |
| Performance Metrics | **1** | **2** | **3** |
| Full Feature Set Mean CV Train Balanced Accuracy | 0.4550 | 0.4400 | 0.4600 |
| Full Feature Set Mean CV Train Accuracy | 0.6524 | 0.6238 | 0.6524 |
| Full Feature Set Test Balanced Accuracy | 0.5000 | 0.5000 | 0.5000 |
| Full Feature Set Test Accuracy | 0.7500 | 0.7500 | 0.7500 |
| Full Feature Set Single Class Prediction | Yes | Yes | Yes |
|  |  |  |  |
| Reduced Feature Set Mean CV Train Balanced Accuracy | 0.4550 | 0.4350 | 0.4750 |
| Reduced Feature Set Mean CV Train Accuracy | 0.6524 | 0.6238 | 0.6857 |
| Reduced Feature Set Test Balanced Accuracy | 0.5000 | 0.5000 | 0.5000 |
| Reduced Feature Set Test Accuracy | 0.7500 | 0.7500 | 0.7500 |
| Reduced Feature Set Single Class Prediction | Yes | Yes | Yes |

The class imbalance persists in the test output, indicating that class\_weight='balanced' is ineffective as it still predominantly predicts the majority class. I used SMOTE to counter the imbalance, and here are the results. After SMOTE, The Training Set Class Distributionos (no response: 23, response: 23)

|  |  |  |  |
| --- | --- | --- | --- |
| Performance Metrics | **1** | **2** | **3** |
| Full Feature Set Mean CV Train Balanced Accuracy | 0.7250 | 0.7500 | 0.8900 |
| Full Feature Set Mean CV Train Accuracy | 0.7156 | 0.7378 | 0.8911 |
| Full Feature Set Test Balanced Accuracy | 0.5000 | 0.6667 | 0.4167 |
| Full Feature Set Test Accuracy | 0.7500 | 0.7500 | 0.6250 |
| Full Feature Set Single Class Prediction | Yes | No | No |
|  |  |  |  |
| Reduced Feature Set Mean CV Train Balanced Accuracy | 0.9150 | 0.8700 | 0.8400 |
| Reduced Feature Set Mean CV Train Accuracy | 0.9133 | 0.8689 | 0.8289 |
| Reduced Feature Set Test Balanced Accuracy | 0.5000 | 0.6667 | 0.7500 |
| Reduced Feature Set Test Accuracy | 0.8750 | 0.7500 | 0.8750 |
| Reduced Feature Set Single Class Prediction | No | No | No |

Results (No Smote):

* Balanced Accuracy: Both feature sets consistently achieved balanced accuracy scores around or below 0.5 during cross-validation, indicating performance no better than random guessing.
* Accuracy: The overall accuracy remained at 0.75 across all runs for both feature sets, likely reflecting the majority class proportion.
* Single Class Prediction: The models consistently predicted only the majority class (no response) on the test set, highlighting the ineffectiveness of the class\_weight='balanced' parameter in addressing class imbalance.

Results (Smote):

* Balanced Accuracy:
  + Full Feature Set: Significant improvement in cross-validation balanced accuracy, ranging from 0.7250 to 0.8900.
  + Reduced Feature Set: Exceptional performance with balanced accuracy scores between 0.8400 and 0.9150.
* Accuracy:
  + Full Feature Set: Increased accuracy in cross-validation runs, peaking at 0.8911.
  + Reduced Feature Set: High accuracy scores ranging from 0.8289 to 0.9133.
* Test Balanced Accuracy:
  + Full Feature Set: Varied results across runs, with some improvements but also instances of decreased performance.
  + Reduced Feature Set: Steady improvement, achieving up to 0.7500 When using the compressed features. (3)
* Single Class Prediction:
  + Full Feature Set: Initially continued predicting a single class but showed improvement with the reduced feature (2) set and the compressed feature set (3)
  + Reduced Feature Set: Successfully predicted both classes in all runs post-SMOTE.

The evaluation demonstrates that **class imbalance** significantly hampers the performance of Random Forest classifiers, leading to single-class predictions and balanced accuracy scores indicative of random guessing. Implementing **SMOTE** effectively addressed some of these challenges, particularly for the reduced feature set, resulting in improved balanced accuracy and the ability to predict both classes.

But The reduced feature set ([HHV6.Status, IL-8, HGB Day 0, %LYM Day 0, FGF-2, TNFa, VEGF, %MON Day 0]) and compressed feature set ([HHV6.Status, IL-8, %LYM Day 0

[%MON Day 0, group\_3\_Compressed\_1, group\_4\_Compressed\_1]) demonstrated substantial balanced accuracy improvements when class imbalance was addressed with SMOTE. This suggests that these features collectively carry some information relevant to distinguishing between responders and non-responders.

**Recursive Feature Elimination with SMOTE and cross-validation**

So next I'm going to look for the optimal subset of features by combining Recursive Feature Elimination (RFE) with SMOTE and cross-validation.

Methodology: Recursive Feature Elimination (RFE) with SMOTE and Cross-Validation

To identify the optimal subset of features for the model, the approach combines Recursive Feature Elimination (RFE) with SMOTE (Synthetic Minority Oversampling Technique) and tests two different cross-validation strategies. The process is as follows:

* Cross-Validation with Stratified K-Folds:
  1. Perform stratified K-Fold cross-validation (e.g., 5 folds) to ensure that each fold maintains the class distribution of the dataset.
  2. For each training subset in the folds, apply RFE to rank the features based on their importance and recursively eliminate the least important features.
* Stability Analysis of Feature Selection:
  1. Track how often each feature is selected as important across all the folds.
  2. Features that consistently appear in multiple folds (exceeding a predefined threshold for selection frequency) are considered stable. This stability suggests that these features are robust and contribute significantly to the model regardless of data splits.
* Feature Importance Evaluation on the Entire Dataset:
  1. Using the subset of stable features identified from the cross-validation process, train a final model on the entire dataset.
  2. Extract feature importance scores (e.g., using RandomForestClassifier.feature\_importances\_ or similar methods) to determine which of the stable features have the greatest impact on predictions.

Problems:

* Did not take test accuracy into account.

Assumptions:

* threshold=3, So appear in 3 of the 5 folds mean stable.

**Normal data**:

RandomForestClassifier:

Stable Features: ['GRO', 'MCP-1', 'RBC Day 0', 'HGB Day 0', 'HCT Day 0', '%LYM Day 0', '%MON Day 0', '%GRA Day 0', 'IL17A', 'PLT Day 0']

SVC:

Stable Features: ['IL-8', 'MCP-1', '%MON Day 0', 'HHV6.Status', 'IL1a', 'HGB Day 0', 'VEGF']

GradientBoostingClassifier:

Stable Features: ['GRO', 'HGB Day 0', '%LYM Day 0', '%MON Day 0', 'PLT Day 0', 'RBC Day 0']

LogisticRegression:

Stable Features: ['IL-8', 'IP-10', 'MCP-1', 'VEGF', '%LYM Day 0', '%MON Day 0', 'Gender', 'Age', 'HHV6.Status', 'GCSF']

Common Features (selected by multiple models):

- %MON Day 0

- MCP-1

- HGB Day 0

- %LYM Day 0

- GRO

- RBC Day 0

- PLT Day 0

- IL-8

- HHV6.Status

- VEGF

Unique Features (selected by only one model):

- HCT Day 0

- %GRA Day 0

- IL17A

- IL1a

- IP-10

- Gender

- Age

- GCSF

**Compressed Data**:

RandomForestClassifier:

Stable Features: ['GRO', 'MCP-1', 'WBC Day 0', 'PLT Day 0', '%LYM Day 0', '%MON Day 0', '%GRA Day 0', 'Group\_3\_Compressed', 'IL-8']

SVC:

Stable Features: ['HHV6.Status', 'IL-8', '%MON Day 0', 'Group\_3\_Compressed', 'Group\_5\_Compressed', 'GCSF', 'IL1a', '%LYM Day 0']

GradientBoostingClassifier:

Stable Features: ['GRO', 'WBC Day 0', '%LYM Day 0', '%MON Day 0', 'Group\_3\_Compressed', 'IL-8']

LogisticRegression:

Stable Features: ['HHV6.Status', 'IL-8', '%MON Day 0', 'Age', 'Group\_3\_Compressed', 'Group\_5\_Compressed', '%LYM Day 0']

Common Features (selected by multiple models):

- %LYM Day 0

- %MON Day 0

- Group\_3\_Compressed

- IL-8

- GRO

- WBC Day 0

- HHV6.Status

- Group\_5\_Compressed

Unique Features (selected by only one model):

- MCP-1

- PLT Day 0

- %GRA Day 0

- GCSF

- IL1a

- Age

**Filtered Data**:

RandomForestClassifier:

Stable Features: ['GRO', 'MCP-1', 'RBC Day 0', 'HGB Day 0', 'HCT Day 0', '%LYM Day 0', '%MON Day 0', '%GRA Day 0', 'IL17A', 'PLT Day 0']

SVC:

Stable Features: ['MCP-1', 'VEGF', '%MON Day 0', 'Gender', 'Age', 'HGB Day 0']

GradientBoostingClassifier:

Stable Features: ['EGF', 'GRO', 'WBC Day 0', '%LYM Day 0', '%MON Day 0', 'VEGF']

LogisticRegression:

Stable Features: ['CMV.Status', 'HHV6.Status', 'IL-8', 'IP-10', 'MCP-1', 'VEGF', '%LYM Day 0', '%MON Day 0', 'Gender', 'Age', 'MIP1b']

Common Features (selected by multiple models):

- %MON Day 0

- MCP-1

- %LYM Day 0

- VEGF

- GRO

- HGB Day 0

- Gender

- Age

Unique Features (selected by only one model):

- RBC Day 0

- HCT Day 0

- %GRA Day 0

- IL17A

- PLT Day 0

- EGF

- WBC Day 0

**Cross Validate featuers:**

------Hepatitis -------

Robust features for Cytometry (appearing in at least 3 runs):

Model Feature frequency mean\_importance

34 Cytometry HGB Day 0 5 0.11650

37 Cytometry WBC Day 0 5 0.08596

31 Cytometry %LYM Day 0 5 0.08396

33 Cytometry HCT Day 0 5 0.07548

35 Cytometry PLT Day 0 5 0.06258

30 Cytometry %GRA Day 0 5 0.05792

36 Cytometry RBC Day 0 5 0.04322

32 Cytometry %MON Day 0 5 0.02620

Robust features for RNA Data (appearing in at least 3 runs):

Model Feature frequency mean\_importance

12 RNA Data M11.3.Erythroid cells\_EXP0 3 0.147567

14 RNA Data M14.35.TBD\_EXP0 3 0.141367

13 RNA Data M13.20.Cell cycle\_EXP0 4 0.133275

15 RNA Data M14.38.TBD\_EXP0 4 0.132275

23 RNA Data M15.95.TBD\_EXP0 5 0.130100

16 RNA Data M14.43.TBD\_EXP0 5 0.113440

28 RNA Data M16.93.TBD\_EXP0 5 0.113420

25 RNA Data M16.66.TBD\_EXP0 5 0.109860

17 RNA Data M14.47.Protein synthesis\_EXP0 4 0.109550

21 RNA Data M15.23.Cell death\_EXP0 4 0.109325

-----Measles------

Robust features for Cytometry (appearing in at least 3 runs):

Model Feature frequency mean\_importance

4 Cytometry %GRA Day 0 5 0.32762

10 Cytometry RBC Day 0 5 0.32386

5 Cytometry %LYM Day 0 5 0.23972

9 Cytometry PLT Day 0 5 0.20832

8 Cytometry HGB Day 0 5 0.19090

6 Cytometry %MON Day 0 5 0.15170

7 Cytometry HCT Day 0 5 0.14202

11 Cytometry WBC Day 0 5 0.12392

Robust features for RNA Data (appearing in at least 3 runs):

Model Feature frequency mean\_importance

45 RNA Data M14.38.TBD\_EXP0 4 0.076075

54 RNA Data M15.26.Neutrophils\_EXP0 3 0.008400

47 RNA Data M14.48.Inflammation\_EXP0 3 0.007700

39 RNA Data M12.10.Inflammation\_EXP0 3 0.007633

56 RNA Data M15.58.Monocytes\_EXP0 3 0.004900

RNA (2 labels):

* RNA unclustere

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Chozen model: SVM | | | | | | | | |
| train | | | | | | test | | |
| Baccuracy | Accuracy | P-Value | | Lower | Upper | Accuracy | F1\_0 | F1\_1 |
| 0.6353 | 0.7188 | 0.718 | 0.3333 | | 0.9167 | 0.625 | 0.7692 | 0.0 |

* RNA clusters

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Chozen model: SVM | | | | | | | | |
| train | | | | | | test | | |
| Baccuracy | Accuracy | P-Value | | Lower | Upper | Accuracy | F1\_0 | F1\_1 |
| 0.7029 | 0.7188 | 0.068 | 0.4167 | | 1.0 | 0.875 | 0.9091 | 0.8 |

A red and blue grid

AI-generated content may be incorrect.

* RNA compressed by module

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Chozen model: Random Forest | | | | | | | | |
| train | | | | | | test | | |
| Baccuracy | Accuracy | P-Value | | Lower | Upper | Accuracy | F1\_0 | F1\_1 |
| 0.6787 | 0.7812 | 0.063 | 0.1064 | | 0.8333 | 0.875 | 0.9231 | 0.6667 |

* RNA compressed by module clusters

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Chozen model: SVM | | | | | | | | |
| train | | | | | | test | | |
| Baccuracy | Accuracy | P-Value | | Lower | Upper | Accuracy | F1\_0 | F1\_1 |
| 0.6812 | 0.6875 | 0.189 | 0.5 | | 1.0 | 0.875 | 0.9231 | 0.6667 |

A red and blue squares

AI-generated content may be incorrect.