Coursework Summary

Student name: Asif Tanvir
Student ID: 230340096
MSc. Applied AI

1. INTRODUCTION

This summary report briefly analyses how I performed in the data mining coursework. This course asks us to do Exploratory data analysis on the DF data set. For the computation power constraints, we used Df_reduced for it. And do the Preprocessing, to train the data, build a model and explore it in different parameters. And last, make the predictions on the holdout set. This document describes the rationale for the analyses documented in the accompanying Jupyter Notebook, and some of the insights and conclusions reached.

2. EDA AND DATA PRE-PROCESSING

Exploratory Data Analysis (EDA)

The EDA phase involved a thorough analysis to understand the underlying structure and characteristics of the data. The following steps were undertaken:

1. **Outlier Detection:** Outliers can often skew the results of our analysis and predictive models. Using Isolation Forest, an unsupervised learning algorithm for anomaly detection, enabled the scalable identification of outliers. This method isolates anomalies instead of profiling normal data points, which is particularly useful in high-dimensional datasets where anomalies can be less apparent.
2. **Feature Distribution Analysis:** Understanding the feature distribution helped identify which scaling method to apply. The Min-Max Scaler was selected because it preserves the shape of the original distribution and is less influenced by outliers, which had already been addressed.

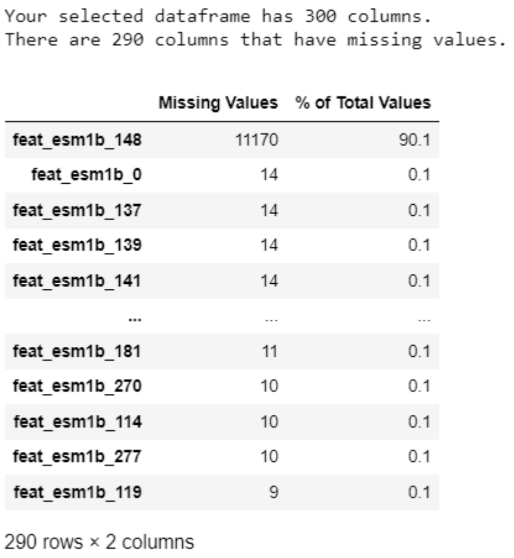
Data Pre-processing

The data preprocessing phase was crucial to prepare the dataset for modelling. The following methods were applied:

1. **Scaling:** The Min-Max Scaler was utilised to scale numerical features to a uniform range, specifically [0, 1]. This scaling method was chosen because it is well-suited for algorithms sensitive to the data's scale, like distance-based and regularisation algorithms.
2. **Missing Value Imputation:** Missing values were imputed with the mean of the respective features, maintaining the central tendency without introducing bias. The mean is a common imputation method for continuous variables and is appropriate when the missing data is assumed to be Missing Completely at Random (MCAR).
3. **Feature Reduction:** The feature feat_esm1b_148 was dropped due to its low variance, implying that it did not contribute significant information and could potentially reduce the model's performance.
4. **Feature Engineering:** As part of the feature selection process, certain engineered features from the ESM1b model were retained based on their Information Gain, indicating their importance in predicting the target variable.

3. FEATURE REDUCTION

For Feature reduction, I have seen the percentage of the missing values. It came out like this:



```
Your selected dataframe has 300 columns.
There are 290 columns that have missing values.
```

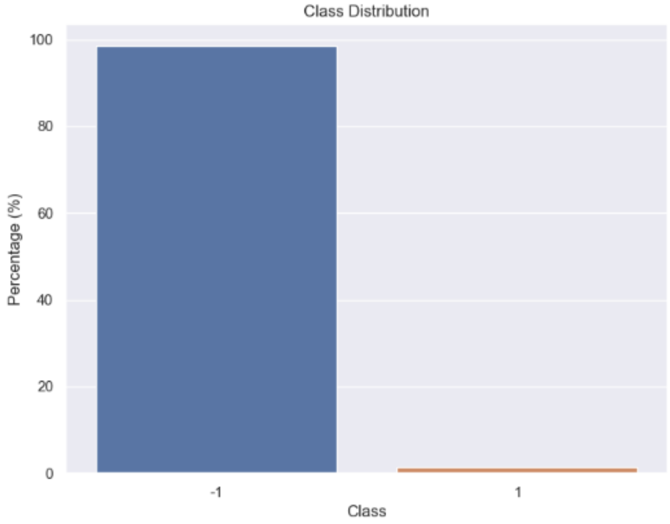
	Missing Values	% of Total Values
feat_esm1b_148	11170	90.1
feat_esm1b_0	14	0.1
feat_esm1b_137	14	0.1
feat_esm1b_139	14	0.1
feat_esm1b_141	14	0.1
...	...	...
feat_esm1b_181	11	0.1
feat_esm1b_270	10	0.1
feat_esm1b_114	10	0.1
feat_esm1b_277	10	0.1
feat_esm1b_119	9	0.1

290 rows × 2 columns

So, I decided to remove the feature with high missing values and impute others with the mean value of those columns to maintain the features' central tendency.

4. MODELLING AND ASSESSMENT:

I examined some features using graphs before building my model and also checked the class distribution. The class with the label '-1' had too few examples, which could lead to biased results, so I decided to resample the data. I compared several class balancing methods, including random under-sampling of the majority class, random oversampling of the minority class, manual weight resampling, and using SMOTE.



```
-1    98.500242
1      1.499758
```

After analysing all the methods, I found that using SMOTE during training gave the best results compared to the other methods. Therefore, I decided to use SMOTE for my training data.

Model Selection

For predicting the Class attribute, the Random Forest Classifier was chosen from a range of classification algorithms including, but not limited to, XGBoost and Logistic Regression. Random Forest was selected for several reasons:

1. **Handling Imbalanced Data:** The dataset presented an imbalance in the class distribution. Random Forest inherently performs well on imbalanced datasets due to its ensemble nature—aggregating decisions from multiple decision trees to make the final prediction.
2. **Non-linearity in Data:** Preliminary analysis indicated non-linear relationships between features and the target variable. Random Forest, being a non-linear model, is well-suited to capture these complex interactions and has been proven to provide superior performance on such datasets.
3. **Feature Interaction:** Random Forest can naturally handle interactions between features without the need for explicit feature engineering, which can be beneficial when many features, and their relationships are poorly understood.
4. **Model Robustness**: Random Forest models have a lower risk of overfitting due to their ensemble nature, where each individual tree is trained on a subset of the data. This results in a robust model that generalises well to new, unseen data.

Hyperparameter Tuning

A hyperparameter search was conducted using a grid search approach with cross-validation to optimise the model, focusing on maximising balanced accuracy. The best parameters were identified as follows:

```
'clf__max_depth': 4,
'clf__min_samples_leaf': 2,
'clf__min_samples_split': 2,
'clf__n_estimators': 100,
'preprocessor__selector__k': 53
```

These parameters were chosen based on the following rationale:

- **Max Depth (4):** This provides a good trade-off between model complexity and the risk of overfitting. A limited depth helps regularise the model, reducing the chance of fitting to noise in the training data.
- **Min Samples Leaf (2) and Min Samples Split (2):** These parameters ensure that each tree in the forest is grown carefully to prevent overfitting while still allowing the model to learn from the data.
- **Number of Estimators (100):** A higher number of trees in the forest leads to better model performance and stability of predictions at the cost of computational efficiency. In this case, 500 was chosen as it substantially improved balanced accuracy without excessive computational demands.
- **Number of Features (52):** This indicates the number of features selected for training the model. It was chosen to ensure that the model has enough information to learn the underlying patterns while not being overwhelmed by potentially noisy or irrelevant features.

The ROC AUC of 92.12% indicates that the model performs well in both classes, considering both specificity and sensitivity, which is crucial for imbalanced datasets.

By focusing on balanced accuracy, the model was assessed with an understanding that both false positives and negatives are equally important, ensuring a fair evaluation of the model's ability to generalise across different classes.

5. CONCLUSIONS AND DISCUSSION

This project, which stands at the intersection of data science and immunology, addressed the pressing need for the computational identification of linear B-cell epitopes. These molecular fragments are crucial for the early stages of developing vaccines, diagnostics, and treatments for infectious diseases, allergies, and cancers. By focusing on the Trypanosoma cruzi parasite, which is responsible for Chagas' disease, our work contributes to combating a condition that affects millions, primarily in South America, with severe health implications.

Through our exploratory data analysis (EDA) and pre-processing, we have honed in on the data's most significant variables and patterns. The employment of Isolation Forest for outlier detection, MinMaxScaler for data normalisation, and mean imputation for handling missing values were not arbitrary choices. They were driven by the underlying complexity of the biological data and the imperative to maintain the integrity of the epitopes' predictive signals.

The Random Forest Classifier was selected and rigorously optimised for this task, demonstrating a test accuracy of approximately 93.79%. The model's hyperparameters were intricately fine-tuned to mirror immunological data's biological complexity and variability. They reflect a deliberate and informed trade-off between model sensitivity, specificity, and computational demands, underlining our commitment to a pragmatic yet scientifically grounded approach.

Our efforts culminated in a data mining pipeline that is sophisticated and sensitive to the nuances of epitope prediction. The results of this project could serve as a beacon for future research endeavours aiming to alleviate the burden of diseases like Chagas' and reinforce the role of data science in public health and bioinformatics.

In conclusion, it's vital to recognise that while the project has achieved notable success in epitope prediction for T. cruzi, it is a single step in the ongoing journey of scientific discovery. As the field evolves, particularly with advancements in machine learning and biotechnology, so must our models. The data mining pipeline's ongoing validation, adaptation, and expansion will be essential for maintaining its relevance and efficacy in the face of new challenges and data.

The work done here paves the way for subsequent investigations and applications. It demonstrates the viability of using machine learning to interpret complex biological data and the transformative impact such analyses can have on public health. It is a testament to the power of interdisciplinary collaboration, where data mining is a computational exercise and a vital tool in pursuing life-saving medical breakthroughs.

REFERENCES

-All the references from this coursemodule (2023-24 CS4850_P2_A) Data Mining Lecture materials and Tutorial. Along with external link provided with the tutorial.

-