1.0 INTRODUCTION

The use of data analytics is revolutionizing healthcare, particularly through predictive modelling that enhances disease risk prediction. This report analyzes a dataset of 4250 healthcare records, integrating clinical, demographic, and medication data. By applying data mining techniques, we uncover patterns essential for developing accurate risk assessment models (Provost & Fawcett, 2013). The study employs supervised and unsupervised learning to address challenges like data imbalance, aiming to improve the precision of predictive algorithms (James et al., 2013). This research highlights the importance of data-driven approaches in refining healthcare analytics, ultimately improving patient care and decision-making processes (Shalev-Shwartz & Ben-David, 2014).

2.0 DATA EXPLORATION, VISUALISATIONS, AND SUMMARY

2.1 Summary of dataset

The dataset comprises 4,250 records and 24 variables, tailored for healthcare predictive modelling. It encompasses a mix of numerical and categorical data, which is pivotal for assessing disease risk levels. The target variable categorizes risk into three categories: low, moderate, and high, which aids in focused risk assessment. The attributes include clinical test results (test_X1 to test_X6) that offer direct insights into health conditions; age and demographic details that are crucial for disease prognosis; and medication usage (medication_A, medication_B, etc.) that sheds light on treatment patterns. It also features additional health status indicators, such as the presence of disorders or tumors, and treatment variables like surgeries, which are correlated with the patient's risk level. This comprehensive amalgamation of medical and demographic information facilitates the development of robust models designed to precisely classify health outcomes by risk.

2.2 Distribution of the Target Variable

The distribution of the target variable was analyzed to understand the prevalence of different risk levels within the dataset. A count plot visualization indicated that 'low_risk' was the most common class, followed by 'moderate_risk', with 'high_risk' being the least frequent. The exact counts from the data are as follows: 'low_risk' with 3,612 instances, 'moderate_risk' with 489, and 'high_risk' with 149. This distribution highlights the imbalance in the dataset, underscoring the potential challenges in modelling and the necessity for appropriate stratification or rebalancing techniques.

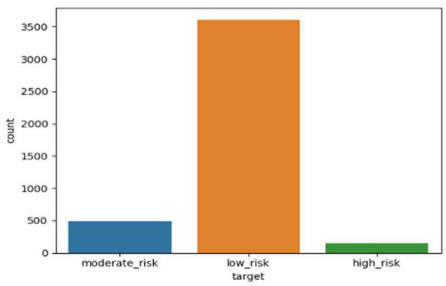


Figure 1

2.3 Data types

The dataset primarily consists of columns classified as objects, which typically represent categorical variables. Additionally, there are columns with float and integer data types, indicative of numerical variables. Upon review, the data types for each column were verified as appropriate for the analysis. For a detailed breakdown of the data types associated with each column used in the modelling, please refer to Appendix A.

2.4 Missing data

The dataset exhibited missing values across several columns, notably within the 'gender' and various test columns. Particularly, 'test_X6' showed a high incidence of missing entries—4,096 out of 4,250 total rows, representing 96.38% of the column. This percentage was determined after a careful analysis of the data. Due to the substantial amount of missing information, 'test_X6' was excluded from further analysis, reducing the total number of columns to 23, including the target variable. Details of the missing data analysis and the decision process are documented in Appendix B.

2.5 Handling of the ID Column

The ID column in the dataset serves as a unique identifier for each entry, distinguishing it from typical numerical or categorical variables. It is not suitable for mathematical operations, nor does it provide meaningful categorization for analysis. Consequently, to avoid potential overfitting where the model might learn to associate specific outcomes with IDs rather than underlying data patterns, the ID column was excluded from the feature set used for modelling. Instead, it was designated as the index, reducing the effective number of feature columns to 22, including the target variable.

2.6 Analysis of Unique Values

Upon reviewing the unique values in the dataset, it was found that the "disorder" column contained only one unique value. Such uniformity suggests that this column would not contribute meaningfully to model differentiation and is unlikely to impact prediction outcomes. Consequently, the "disorder" column was earmarked for removal during the preprocessing phase of the analysis. Detailed findings from this review can be found in Appendix C.

2.7 Statistical Analysis Overview

Statistical examination of the numerical columns in our dataset revealed critical insights:

Age: A highly anomalous maximum value of **65,526** and a large standard deviation suggest the presence of outliers or data entry errors. The median age of **55** indicates a predominantly middle-aged demographic.

Test Variables (Test_X1 to Test_X5): Each test exhibited potential outliers, with maximum values significantly exceeding the 75th percentiles, suggesting right-skewed distributions across these metrics.

These findings are further detailed in the diagram below, which represents the distribution and range of values. The presence of outliers will be thoroughly investigated to determine their accuracy and relevance to the modelling efforts, ensuring that the final analysis is based on robust and error-free data.

	age	test_X1	test_X2	test_X3	test_X4	test_X5
count	4250.000000	3839.000000	3007.000000	4034.000000	3858.000000	3863.000000
mean	67.374824	7.342463	2.035580	104.919623	0.970846	110.090834
std	1004.518821	32.657963	0.920404	35.496255	0.162474	39.837621
min	1.000000	0.005000	0.050000	2.000000	0.250000	1.400000
25%	37.000000	0.600000	1.600000	87.000000	0.870000	92.000000
50%	55.000000	1.500000	1.900000	102.000000	0.960000	107.000000
75%	67.000000	3.000000	2.300000	121.000000	1.060000	125.000000
max	65526.000000	530.000000	18.000000	430.000000	1.960000	642.000000

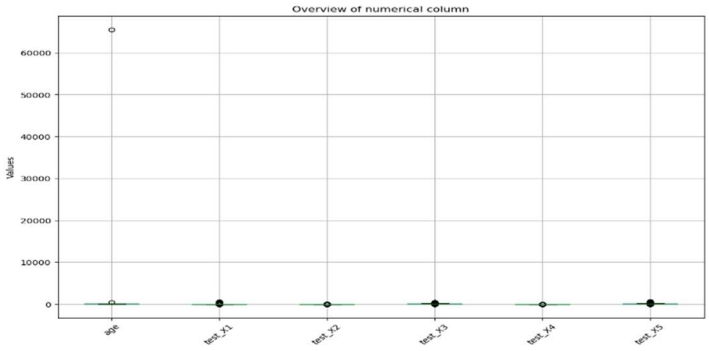
Figure 2

2.8 Categorical and Numerical Columns

After checking for the datatypes, this helped in dividing the columns into categorical and numerical columns. The age and test columns were named as numerical columns, while the rest of the columns, excluding the target column were named as categorical columns. This helped us to perform more visualisations and also this was used at the encoding process later in the analysis.

2.9 Box Plot Visualization Analysis

Box plot visualization was performed on the numerical columns of the dataset, effectively highlighting the outliers previously identified in the statistical analysis. The visualization, as illustrated below, clearly demonstrates the presence of a significant outlier in the age column, distinctly separated from the other data points. This graphical representation confirms the extreme values noted in our earlier analysis and aids in visual confirmation of data points that deviate markedly from the typical range.



2.10 Risk Level Distribution Analysis Summary with numerical columns

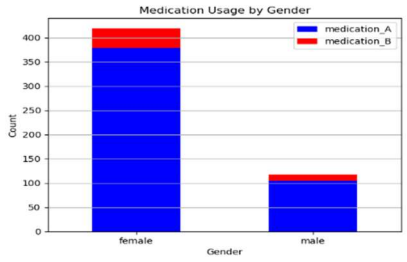
An extreme age outlier identified over 60,000 in the moderate-risk category will be adjusted to the median to ensure accuracy. Test_X1 shows a broad range of values with essential outliers for diverse health conditions, while Test_X2 and Test_X3 confirm increasing median values with higher risks, underscoring their predictive reliability. Test_X4, though showing trends, requires combination with other metrics due to overlap, and Test_X5, effectively distinguishing higher risks, will be prioritized in models. Outliers in test results are retained to capture all health scenarios, ensuring the model's robustness and comprehensive risk assessment capabilities. For visual reference, the box plots illustrating these distributions can be found in Appendix D .

2.11 Summary of Risk Level Distribution Analysis with Emphasis on Categorical Variables

The analysis of categorical variables within our dataset reveals a consistent pattern across visualizations: the 'low risk' category consistently shows the highest count, followed by 'moderate risk', and 'high risk' being the least frequent. Notably, in the gender-specific data, females are more prevalent than males. Furthermore, across health condition and status categories, a larger number of patients are noted to have 'no' responses prevailing over 'yes' responses, indicating fewer instances of the conditions being present. For detailed visual insights into these distributions, refer to the diagram in Appendix E. This clear delineation across categorical variables underscores the varying risk levels and demographic dynamics within the patient population being studied.

2.12 Summary of Additional Visualizations

The dataset's visual analysis includes pie charts for health condition columns, illustrating a predominant proportion of 'no' responses exceeding 95% across these variables. Specifically, the gender distribution within the dataset shows that females comprise 67.8% while males account for 32.2%. For detailed visuals, refer to the pie charts in Appendix F. Further visualizations focus on the relationship between medication usage and gender, revealing a higher uptake of Medication A among patients, particularly among females compared to males. These visualizations provide a clear depiction of health condition prevalence and medication patterns within the study population. See below for the visuals of the medication patterns.



Scatter Plot Visualisation:

The scatter plots reveal relationships between test results (Test_X1, Test_X3, Test_X5) and patient age against risk levels (low, moderate, high), showing clear groupings and potential for risk differentiation. Test_X1 and Test_X2 show tight clustering for moderate and high-risk levels with lower test values, while Test_X3 and Test_X5 display wider spreads, indicating that higher test values could correspond to higher risks. Age combined with Test_X2 shows a slight increasing trend in test values with age across risk groups. Notably, the plots also reveal outliers that extend well beyond the main clusters, emphasizing extreme test values or ages that may require additional clinical attention or consideration in modelling. The visualisation can be seen in Appendix G.

Scatter plot visualizations of the numerical columns against gender revealed a pattern of outliers predominantly among females, including a notable outlier in age. These visualizations are detailed in Appendix G(i).

3.0 DATA CLEANSING AND PRE-PROCESSING

3.1 Duplicate Records Analysis

The dataset was examined for duplicate entries, revealing a total of 40 duplicates. Upon closer inspection, these records, despite having identical data points, were associated with distinct patient IDs, indicating that the duplicates represent different individuals rather than erroneous data repetitions.

3.2 Handling Missing Values

For handling missing values in our dataset, we employed two methods: the KNN imputer and a combination of mode imputation for categorical (object) columns and mean imputation for numerical columns. The KNN imputer method estimates missing values using the nearest neighbours approach, which is especially effective in maintaining the integrity of complex patterns in medical data. The second method involves using the most frequent values to fill missing entries in categorical columns and the mean for numerical columns, aimed at preserving the central tendency without introducing significant bias. After assessing both methods, the combined approach of mode and mean imputation was preferred. It effectively preserved the dataset's statistical characteristics such as mean and standard deviation, while also maintaining consistency in the data's distribution, making it more suitable for subsequent analysis and modelling. Below is the descriptive analysis after imputation, which you can compare with the initial descriptive analysis in Figure 2.

	age	test_X1	test_X2	test_X3	test_X4	test_X5
count	4250.000000	4250.000000	4250.000000	4250.000000	4250.000000	4250.000000
mean	67.374824	7.342463	2.035580	104.919623	0.970846	110.090834
std	1004.518821	31.038321	0.774158	34.582253	0.154798	37.980106
min	1.000000	0.005000	0.050000	2.000000	0.250000	1.400000
25%	37.000000	0.700000	1.700000	88.000000	0.880000	94.000000
50%	55.000000	1.700000	2.035580	104.000000	0.970846	110.000000
75%	67.000000	4.500000	2.100000	120.000000	1.050000	123.000000
max	65526.000000	530.000000	18.000000	430.000000	1.960000	642.000000

3.3 Outliers Detection

Three distinct outlier detection methods were applied—DBSCAN, Isolation Forest, and Local Outlier Factor (LOF)—to understand the distribution of anomalies in our dataset. DBSCAN uses min_samples=10 and eps=10 to define dense clusters, effectively identifying 581 outliers in sparse regions. Isolation Forest, employing max_samples=3158, contamination=0.01, and random_state=2, isolates anomalies through random feature selection and splitting, with the contamination rate adjusting sensitivity. Local Outlier Factor (LOF) utilizes n_neighbors=25 and contamination=0.01 to measure local density deviations, flagging significantly less dense points as outliers. The other two methods gave an output of 43 outliers. The aim of this analysis is not to remove all detected outliers but to understand how each method characterizes them. This knowledge is crucial as our model needs to be robust enough to handle outliers effectively in real-world scenarios. Using various outlier detection methods allows for a comprehensive assessment of anomalies. This diversity in approaches ensures that we capture a broad spectrum of outliers, enhancing the robustness of our data analysis. Each detection method provides a unique perspective on identifying outliers, and their combined application offers a thorough exploration of potential data irregularities. Outliers are retained in the data as this is a medical data This approach not only helps in fine-tuning our analysis but also in building models that are resilient in diverse conditions.

3.4 Target Variable Encoding

To effectively prepare the dataset for modelling, the target variables were encoded using a label encoder, a method well-suited for categorical target data. Post-encoding, the risk levels were assigned numerical values: 'low risk' as 1, 'moderate risk' as 2, and 'high risk' as 0. This encoding facilitates the machine learning algorithms' ability to process and learn from the data. In the

prepared dataset, the feature set is designated as 'X' and the encoded target variable as 'Y', with 'Y' corresponding to the risk level in the 'target' column. This organization supports clear distinction and manipulation in predictive modelling tasks.

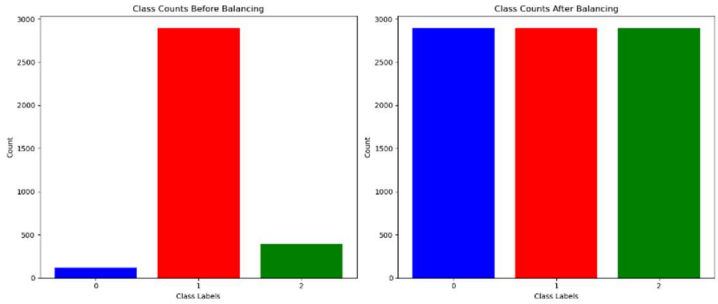
3.5 Dataset Splitting Strategy

For our machine learning model development, we've opted for an 80/20 split between the training and validation sets. This decision balances maximizing our training data, with 3,400 observations, to effectively capture complex patterns, while dedicating 850 observations to the validation set, ensuring robust model performance evaluation. This split prevents underfitting and overfitting, maintaining an optimal balance for model robustness and accuracy assessment. The 80/20 split is widely recognized as a best practice in data science, especially for datasets of moderate size like ours. Preliminary tests confirmed that this ratio provides the best compromise between training capacity and validation accuracy, making it the ideal choice for achieving reliable and generalizable model predictions.

3.6 Balancing the Data for Model Training

Balancing the dataset is a crucial preprocessing step to ensure that our models perform optimally on both training and validation sets. Initially, the dataset was imbalanced, as highlighted in Figure 1. To address this, two balancing methods were implemented: manual upsampling of the majority class and Synthetic Minority Over-sampling Technique (SMOTE). The initial value count in the target column was: low risk at 2,890, moderate risk at 391, and high risk at 119, identifying low risk as the predominant category. Post-manual upsampling, the training dataset expanded to 8,670 rows, achieving uniform distribution across risk categories with each having 2,890 instances.

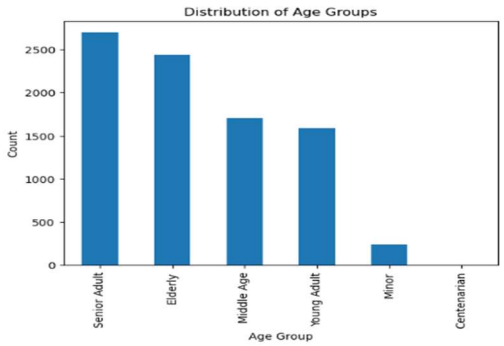
Both manual and SMOTE upsampling techniques were applied to the training set to evaluate which method enhances model performance more effectively. Visualizations of the target variable distribution, before and after balancing, are provided to illustrate the improvements in dataset uniformity. This comparative approach helps determine the most effective strategy for data balancing in our modelling process.



3.7 Feature Engineering

Feature engineering was undertaken on the training dataset to enhance model accuracy. A significant addition was the creation of a new feature, 'age group', derived from the age column. This categorization was based on specific age ranges: individuals under 18 were classified as 'minor'; those between 18 and 35 as 'young adult'; 35 to 50 as 'middle age'; 50 to 65 as 'senior adult'; 65 to 100 as 'elderly'; and those over 100 as 'centenarian'.

This categorization process was applied consistently across both the training and validation datasets. Visualization of the age groups within the training set indicated a prevalence of 'senior adults', followed by 'elderly', 'middle age', 'young adult', and 'minor'. See below;



Subsequently, the feature set 'X' was updated to include this new 'age group' category, and the categorical columns were redefined to incorporate it. This enhancement aims to provide a more nuanced input into our models, potentially improving their predictive performance.

3.8 Pair plot Visualization Analysis

A pair plot visualization was conducted to explore the relationships between the newly created 'age group' feature and various test results. This analysis revealed significant variability in test outcomes across different age groups, reflecting physiological differences likely influenced by age. Furthermore, the visualization highlighted that certain age groups demonstrate strong correlations among test results, indicating potential connections between specific health factors.

Visual evidence from the pair plot, which effectively maps these relationships and trends, is available for detailed review in Appendix H. This visualization helps underscore the relevance of age in medical diagnostics and emphasizes the interconnected nature of the test variables within specific age demographics.

3.9 Dropping of column

In addition to dropping the 'test_X6' column due to a high incidence of missing values, the 'disorder' column was also removed from both the training and validation datasets. This decision was made after it was determined that the 'disorder' column contained only a single unique value, as verified during the uniqueness count (see Appendix C). Given its lack of variability, this column was deemed to have no predictive value and therefore unnecessary for inclusion in the modelling process. Subsequently, the feature set 'X' was redefined, removing the 'disorder' column to ensure that the model focuses only on variables that could influence the outcomes, enhancing the effectiveness and accuracy of our predictive models.

3.10 Outlier Management in Age Data

Upon splitting the dataset into training and validation sets, we conducted a detailed examination of outliers, particularly focusing on the age variable. This analysis identified exceedingly high age values that appeared implausible for human ages, such as 65,526 in the training set and 455 in the validation set.

To address these anomalies, we replaced such outlier values with the median age value of 55, rather than the mean. The choice of median over mean is strategic; the median is less affected by extreme values and therefore provides a more accurate reflection of the central tendency, especially in skewed distributions. This approach not only preserves the integrity and original characteristics of the dataset but also ensures that no valuable data is lost by removing entire records.

This method of handling outliers maintains the robustness of our dataset, ensuring that subsequent analyses and modelling efforts are based on reliable and representative data.

3.11 Standardizing of Numerical Columns

In our medical data analysis, numerical variables such as "age" and test results ("test_X1" to "test_X5") were standardized using scikit-learn's StandardScaler, which transforms data to have zero mean and unit variance. This crucial preprocessing step ensures uniformity in scale, allowing all features to contribute equally and eliminating bias from varying scales. Additionally, standardization optimizes machine learning algorithms that assume normally distributed data, improving both convergence speed and overall model accuracy. By applying the same scaling to training, validation, and test datasets, we maintain consistency across different data sets, reduce

the influence of outliers, and enhance the robustness and reliability of our predictive models in medical diagnostics.

3.12 Encoding of categorical columns

To optimize the encoding process for our medical dataset, we first analyzed the value count of each unique value in the categorical columns. This analysis informed our choice of encoding methods for different variables based on their characteristics. For most categorical variables, including 'gender', 'sick', 'pregnant', and others, we employed One-Hot Encoding. This method was chosen because these variables are nominal with no inherent order, ensuring that the encoding does not introduce any artificial ordinal relationships that could influence the models. One-Hot Encoding transforms these categories into separate binary columns, which is ideal for maintaining unbiased inputs for the models.

For the 'age_group' column, which clearly exhibits an ordinal structure with categories like Young, Middle Age, and Elderly, we utilized Ordinal Encoding. This method preserves the natural ordering of the categories, thereby providing the model with meaningful information that reflects the intrinsic progression of age.

Following this encoding process, the transformed dataset expanded to a total of 35 columns in both the training and validation sets. These encoding strategies are crucial for representing data effectively in machine learning applications, enhancing model compatibility and efficiency while minimizing bias. This approach ensures more accurate and reliable predictive outcomes.

Note: All preprocessing techniques were consistently applied to both the manually upsampled data (with majority) and the SMOTE-sampled data. This ensures uniform treatment across different datasets, enhancing the comparability and reliability of our modelling outcomes.

3.13 Feature Selection and Correlation Analysis in Predictive Modelling

This report outlines the feature selection and correlation analysis performed on a medical dataset, aimed at enhancing predictive modelling. The analysis was conducted on two differently prepared training sets: the manually upsampled train set labelled as 'X_train' and the SMOTE-sampled train set named 'train_X_smote'.

Methods and Results:

Feature Selection: We employed the F-score method to pinpoint key predictive features, notably 'test_X3', 'test_X5', and 'test_X2'. These features were identified as highly impactful for our models.

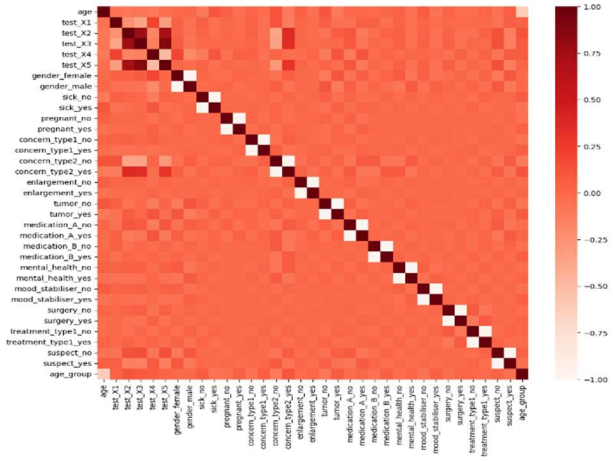
Correlation Analysis: To mitigate issues from multicollinearity, a correlation threshold of 0.8 was applied. This threshold was instrumental in identifying and addressing high correlations, particularly between 'test_X3' and 'test_X5', ensuring that only the most independent variables were retained for modeling.

Model Development Strategy:

Dual Approach: Models were developed using both the full set of features and a refined subset determined after the correlation analysis. This dual approach allows for effective comparison and leverages data optimally to ascertain the best performing feature set.

For detailed findings and the specific analysis of feature op-selection for each training set, please refer to Appendices I and J.

See below a visualisation of the correlation analysis.



4.0 SUPERVISED MODEL TRAINING, TUNING, AND EVALUATION:

Our approach to modelling complex data is influenced by established methods discussed by Bishop in Pattern Recognition and Machine Learning (2006). His thorough exploration of support vector machines provides a strong theoretical foundation for one of the choice of algorithms, ensuring our applications are both rigorous and effective. Furthermore, Bishop's detailed treatment of dimensionality reduction techniques, particularly Principal Component Analysis (PCA), has informed our data preprocessing strategies. This guidance has been instrumental in enhancing model performance by reducing computational complexity and minimizing the risk of overfitting.

4.1 Description of Models

Our study utilized four key supervised learning models, chosen for their effectiveness in handling complex medical data:

k-Nearest Neighbours (k-NN): Optimized via grid search to enhance classification accuracy by adjusting parameters like neighbour count and distance metrics.
Support Vector Machine (SVM): Fine-tuned for high-dimensional data spaces, focusing on kernel type, penalty parameters, and gamma to improve class separation.
Decision Tree: Refined for better interpretability and accuracy with grid search adjustments on depth and splitting criteria.
Random Forest: Enhanced stability and performance through grid search modifications of the number of trees and feature selection.

4.2 Model Training

Models were initially trained on both manually upsampled with majority class and SMOTE-sampled datasets to establish baseline performance. Feature selection was applied in two phases: using all available features initially, and then refining this set after correlation analysis to optimize feature relevance. The parameters of each model were first checked to see the exact parameters that works with the base model. Hyperparameter tuning was conducted via grid search and cross-validation to ensure robust generalization—a crucial requirement in medical applications where prediction accuracy directly impacts patient outcomes.

The evaluation will centre on the top four models, assessing which preprocessing and feature selection strategies most effectively enhance model performance. This analysis will help identify the most beneficial techniques for improving accuracy and reliability in medical predictive modelling.

KNN

Base Model

The KNN base model demonstrated higher performance on the training set compared to the validation set when applied to medical data with all features included. This discrepancy, indicative of overfitting, is highlighted by an average accuracy of 96.5% derived from two balanced training sets, compared to 79.5% on the validation data. For further details, please refer to the code in the Appendix notebook."

Hyper-parameter tunned model

The KNN tuned model showed better performance on the training set than on the validation set in medical data analysis after correlational feature selection, indicating overfitting with an accuracy of 100% from two balanced training sets and lesser in validation set. However, the manually upsampled tuned model, named 'knn_best,' with features post-correlational analysis, emerged as superior. It utilized parameters of n=1, algorithm=balltree, p=1, and weight='uniform', achieving 89% accuracy on validation. For more details, refer to the code in the Appendix notebook.

SVM

Base Model

The base SVM model performed slightly below the KNN on the training set but surpassed it on the validation set in terms of accuracy. Using features selected after correlational analysis and manual upsampling, this model, named 'svm_model_corr,' achieved a consistent accuracy of 94% on both the training and validation sets, indicating strong generalization without signs of overfitting or underfitting. It operated with default parameters, proving to be the most effective SVM configuration during training, and effectively capturing essential data patterns while maintaining performance stability across different data sets. For further details, please refer to the code in the Appendix notebook.

Hyper-parameter tunned model

The tuned SVM model underperformed compared to the tuned KNN and outperformed the base SVM model on the training set, but it showed weaker results on validation data than the base model. However, it performed significantly better with SMOTE-balanced data, achieving 97% accuracy on the training set and 92% on the validation set using both sets of features. For further details, please refer to the code in the Appendix notebook.

DECISION TREE

Base Model

The model excelled across all preprocessing techniques, achieving a perfect 100% accuracy on the training set and outperforming both SVM and KNN models significantly. On the validation set, it maintained robust performance, particularly with manually upsampled data, where it achieved an impressive average accuracy of 98.7% using a variety of feature sets. For further details, please refer to the code in the Appendix notebook.

Hyper-parameter tunned model

The tuned model demonstrated exceptional performance across all preprocessing techniques, achieving 100% accuracy on the manually upsampled training set and 99% on the SMOTE training set, significantly surpassing both SVM and KNN models. The standout decision tree model, named 'classifier_decision_tree_1,' was optimized with manual upsampling and utilized all features, configured with optimal parameters: {'criterion': 'entropy', 'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 2}. It achieved 100% accuracy on the training set and 99% on the validation set. The consistent high performance across both train and validation sets, despite using all features, suggests the model is well-tuned without overfitting. This robustness and ability to generalize indicate that the model is effectively safeguarded against underfitting, maintaining high accuracy without being overly tailored to the training data.

RANDOM FOREST

Base Model

The model excelled across all preprocessing techniques, achieving 100% accuracy on the training set, matching the performance of the base decision tree model. On the validation set, it showed strong generalization, particularly with manually upsampled data, achieving nearly 99% accuracy using all features. This high level of accuracy on both training and validation sets suggests that the model, named 'rf_model', is well-calibrated and effectively avoids overfitting, despite using default parameters. Its robust performance on unseen data also indicates that it is not underfitting, successfully capturing the underlying patterns without being overly simplistic.

Hyper-parameter tunned model

The tuned model exhibited outstanding performance across various preprocessing techniques, achieving 100% accuracy on both balanced training datasets. However, its performance on the validation sets was somewhat lower compared to the baseline random forest model. Notably, the tuned model performed better on the validation set with manually upsampled data than with SMOTE data, which showed significantly lower results.

4.3 Findings

Our training analysis revealed key insights:

Class Performance: Models generally struggled with class 0 but performed well on classes 1 and 2. Detailed metrics are in Appendix K.

Data Sampling: Models showed improved outcomes with manually upsampled data, suggesting better training with balanced datasets.

Overfitting: Noticeable reductions in overfitting were observed in the decision tree and random forest models.

Feature Suitability: SVM and KNN models trained more effectively with features selected post-correlation analysis, whereas decision tree and random forest models excelled with a full feature set.

4.4 Evaluation Metrics

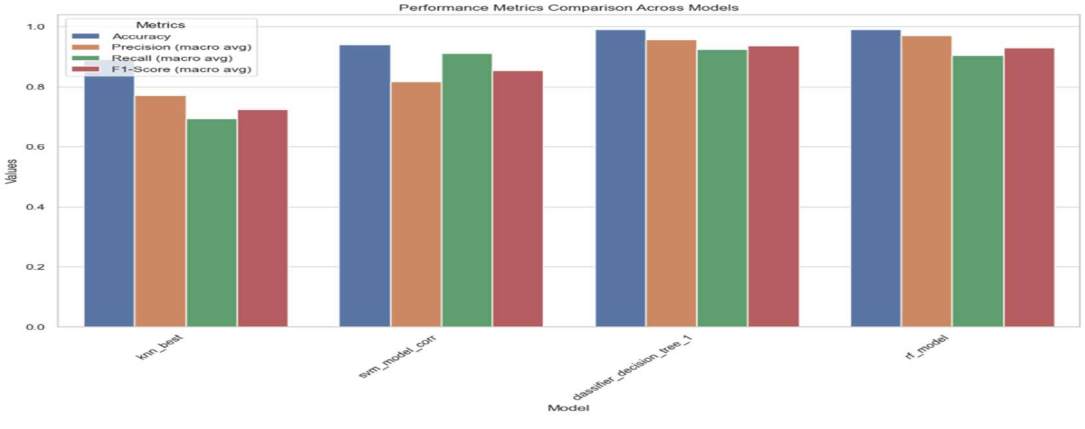
Guided by Hastie, Tibshirani, and Friedman (2009) in "The Elements of Statistical Learning," we selected accuracy, precision, recall, and F1-score as our evaluation metrics and implemented cross-validation to avoid overfitting. These practices ensure robust model performance, as demonstrated in diverse applications such as medical diagnostics and market prediction.

In our multi-class classification model, particularly pertinent to medical diagnostics, macro averages was utilised for precision, recall, and F1-score to ensure a balanced evaluation across all classes. This method treats each class equally, crucial in handling data imbalances, and helps identify any model bias towards predominant classes, ensuring comprehensive performance insights and fairness in metric evaluation. Macro averaging is essential for verifying that the model accurately and equitably diagnoses all conditions, irrespective of their frequency.

The classification report can be referred to in the appendix K, but below is a table of evaluation metrics and visualisation showing the report of the four models using their macro averages.

Model Names	Accuracy	Precision (macro avg)	Recall (macro avg)	F1-score (macro avg)
Knn_best(Knn)	0.89	0.770	0.693	0.723
Svm_model_corr(svm)	0.94	0.817	0.910	0.853
Classifier_decision_tree_1 (Decision tree)	0.99	0.957	0.923	0.937
rf_model (randon forest)	0.99	0.970	0.903	0.930

The bar chart displays a comparison of key performance metrics across the four models;



Accuracy: The Decision Tree and Random Forest models lead, both achieving near-perfect scores, indicating highly accurate classifications across all classes.

Precision: Random Forest showcases the highest precision, suggesting it is the best at minimizing false positives among the models evaluated.

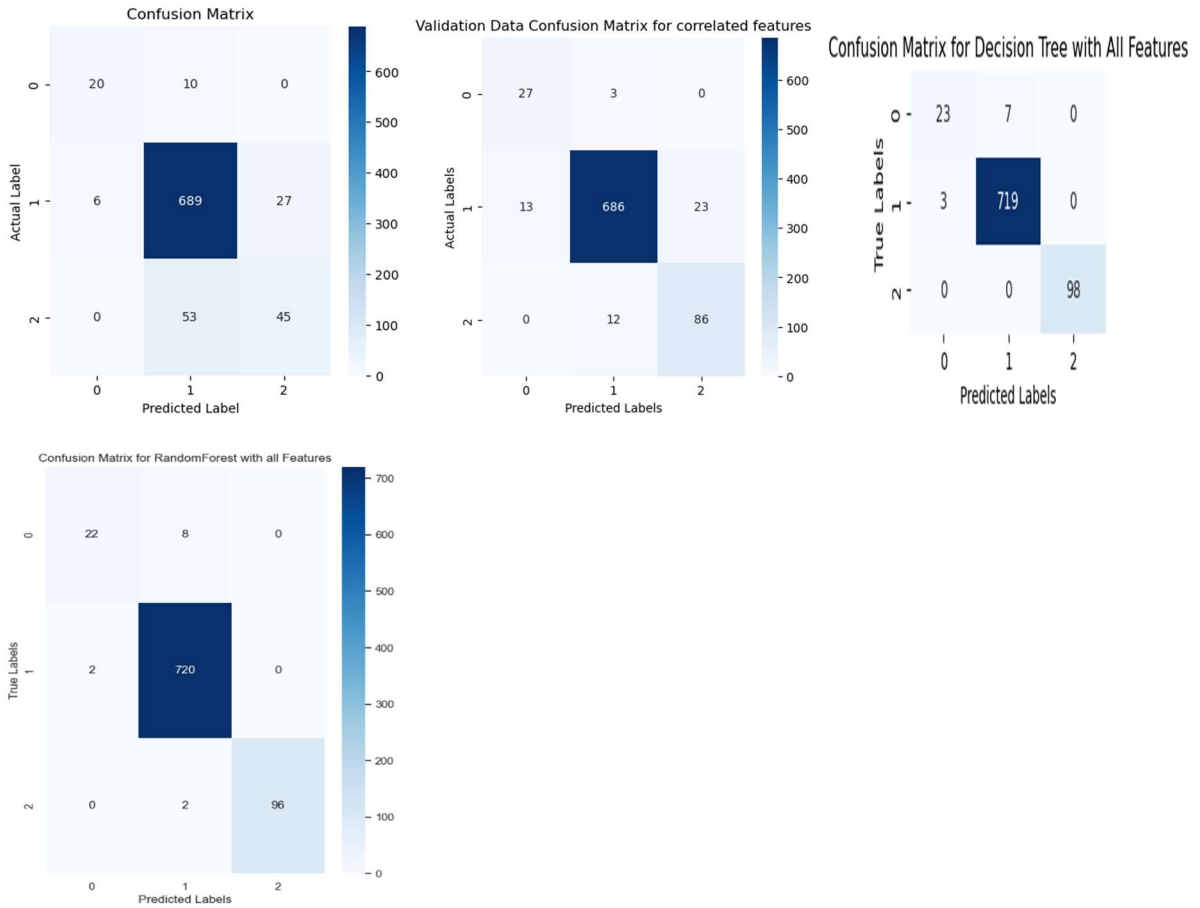
Recall: The Decision Tree model exhibits superior recall, making it the most reliable for identifying all positive instances across classes without missing crucial cases.

F1-Score: Both the Decision Tree and Random Forest models show high F1 scores, indicating a strong balance between precision and recall, which is vital for models where both types of classification errors have severe implications.

These insights underline the strengths and weaknesses of each model, aiding in selecting the most appropriate model based on the specific requirements of the application, such as prioritizing fewer false positives (precision) or fewer false negatives (recall). The summary of these metrics shows decision tree and random forest as good models.

4.5 Confusion Matrix

To determine the most effective model for our classification task, we closely examined the confusion matrixes of each candidate model. These matrices provide critical insights into each model's ability to accurately classify instances across different classes. Below, we present the confusion matrices for the four evaluated models.



The first picture of the confusion matrix is for the KNN model, second for the SVM model, third for the decision tree model and fourth for the random forest model.

Interpretation: In the context of health risk conditions, where accurate classification is crucial, here's a brief yet detailed assessment of the performance of four models based on their confusion matrices:

Random Forest :

High Risk (Class 0): Detected 22 true positives but misclassified 8 as low risk, raising concerns about under-diagnosis in critical cases.

Low Risk (Class 1): Highly accurate with 720 correct out of 722, showcasing excellent identification.
Moderate Risk (Class 2): Reliable with 96 accurately identified and only 2 misclassifications.
Summary: Strong overall performance with some concerns in high-risk accuracy.

Decision Tree:

High Risk (Class 0): Identified 23 true positives, though 7 were misclassified as low risk, needing improvement for critical risk assessments.
Low Risk (Class 1): Near-perfect accuracy with 719 correct classifications.
Moderate Risk (Class 2): Flawlessly identified all 98 cases.
Summary: Excellent in moderate risk, with minor high-risk classification issues.

KNN Model:

High Risk (Class 0): Managed 20 true positives but had 10 misclassifications as low risk, indicating significant identification issues.
Low Risk (Class 1): Correctly classified most but had 27 errors classifying as moderate risk.
Moderate Risk (Class 2): Struggled considerably, misclassifying over half.
Summary: Shows major deficiencies, notably in moderate and high-risk conditions.

SVM Model:

High Risk (Class 0): Effectively identified most high-risk cases with few errors, indicating strong reliability.
Low Risk (Class 1): Several low-risk cases misclassified as moderate, potentially causing over-treatment.
Moderate Risk (Class 2): Generally good but with notable errors.
Summary: Robust in high-risk detection but less accurate in low and moderate risk, impacting potential treatment strategies.

4.6 Overall Recommendation:

The Decision Tree model stands out in our analysis, particularly for its superior performance in identifying high-risk health conditions—a crucial advantage in medical settings where accuracy is imperative. It exhibits fewer misclassifications in the high-risk category compared to the Random Forest model and achieves excellent performance metrics across all risk levels, including perfect classification of moderate risk and near-perfect handling of low risk.

Moreover, the Decision Tree provides exceptional clarity and interpretability, essential for clinical decision-making and transparency in model predictions. Given its robust performance and precision in critical risk assessments, the Decision Tree model is recommended as the optimal choice for deployment on test data, ensuring reliable and effective risk stratification. This decision tree model is named classifier_decision_tree_1 which was hyper-tuned to give the best parameters as (random_state=42, criterion='entropy', max_depth=None, min_samples_split=2, min_samples_leaf=1), using all features and the manual upsampled train data.

5.0 UNSUPERVISED LEARNING USING CLUSTERING ALGORITHMS

Clustering algorithms are essential for identifying homogeneous groups within large datasets, allowing researchers to discover patterns and insights that are not readily apparent, thereby facilitating data-driven decision-making in various fields such as marketing, biology, and public health (Tan, Steinbach, & Kumar, 2005)

5.1 Summary of dataset: The Disease training dataset was utilized for the clustering process as well, applying the same imputation method to handle missing values as was used in the supervised learning. Additionally, the target variable was removed to facilitate the unsupervised learning aspect of the clustering.

5.2 Preprocessing and cleaning

The same preprocessing methodology applied in the supervised learning phase was also employed for this analysis. This included dropping the 'test_X' and 'disorder' columns, setting 'ID' as the index, and addressing age outliers using the median age. The dataset was then segmented into numerical and categorical columns to facilitate encoding for model usage. The numerical columns were standardized, and all categorical columns were one-hot encoded to ensure they were properly formatted for the analysis.

5.3 KDE plot visualisation of features

In our analysis of the Kernel Density Estimation (KDE) plots, we noted that binary and categorical variables like gender and pregnancy status show sharp, distinct peaks indicative of specific conditions. These features are crucial for uncovering medical patterns, even though they may exhibit skewness. Continuous features, representing physiological data, also provide deep insights into patient variation and are integral to nuanced clustering analysis. Emphasizing the clinical importance of each feature, our approach incorporates all available data to ensure comprehensive patient profiling. This holistic analysis is vital in medical research for integrating demographic and clinical data, enhancing our understanding of health outcomes. To refine our clustering methodology and avoid redundancy, we will further investigate correlations between features. Overall, the KDE plots have revealed critical data distributions, guiding our use of all features to uncover clinically relevant patient subgroups and support advanced personalized medical strategies. kindly refer to the visualisation of the kde plots in L.

5.4 Correlational Analysis

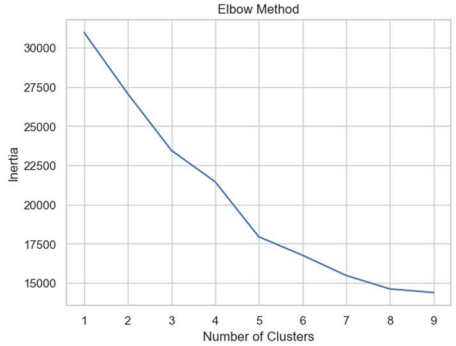
In the correlation heatmap analysis of our dataset, several pertinent observations were noted. The heatmap's diagonal displays perfect correlations, as expected, since it represents each feature's correlation with itself. Notably, the majority of the features exhibited low to moderate positive correlations, indicating minimal redundancy among them, which supports our decision to include all features in the clustering analysis to capture a broad spectrum of information. Negative correlations were also observed, suggesting inverse relationships between certain features, such as gender_male versus gender_female and sick_yes versus sick_no, which highlight binary feature interactions where the presence of one attribute directly implies the absence of the other. These interactions are crucial for accurately capturing the distinct states or conditions relevant in a medical context. Furthermore, the lack of high correlations between many variables suggests that each feature contributes unique information, reinforcing the utility of employing a comprehensive dataset for clustering. Overall, the correlation heatmap has provided essential insights into the dataset’s structure, guiding our clustering approach to ensure robust and informative results.

5.5 Clustering techniques

In the definitive work "Data Mining: Concepts and Techniques," Jiawei Han et all (2011) provides a thorough explanation of how clustering techniques such as KMeans and hierarchical clustering categorize data into clear, distinct groups. This segmentation aids significantly in improving decision-making processes in various sectors, notably healthcare and retail.

5.5.1 KMeans Clustering

The elbow method was used to find the best value of cluster, below is the visualisation.



The elbow visualization indicates that the optimal number of clusters for the dataset might be around 3 to 5, as the rate of decrease in inertia begins to slow down significantly after these points, suggesting diminishing returns from adding more clusters beyond this range.

The silhouette score

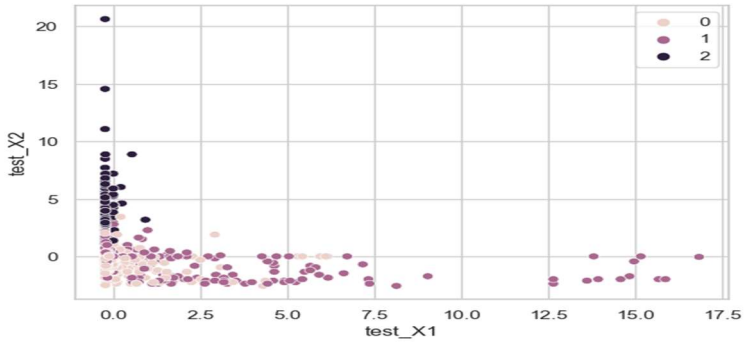
The silhouette score was further evaluated to identify the best clustering configuration based on the highest score achieved. The silhouette score is a measure of how similar an object is to its own cluster compared to other clusters. A higher silhouette score indicates better-defined clusters, with a score close to +1 suggesting a perfect match to its own cluster and distinct separation from other clusters. This metric is crucial as it helps us determine the optimal number of clusters for our model, ensuring that we use the most effective clustering configuration for both analysis and visualization purposes. Below is a screenshot showing the evaluation of these scores which shows us the best K=3.

```
    # Print the silhouette score for the current k
    print(f"Silhouette Score for k={k}: {score:.4f}")
```

```
Silhouette Score for k=3: 0.2458
Silhouette Score for k=4: 0.1741
Silhouette Score for k=5: 0.1579
Silhouette Score for k=6: 0.1603
Silhouette Score for k=7: 0.1490
```

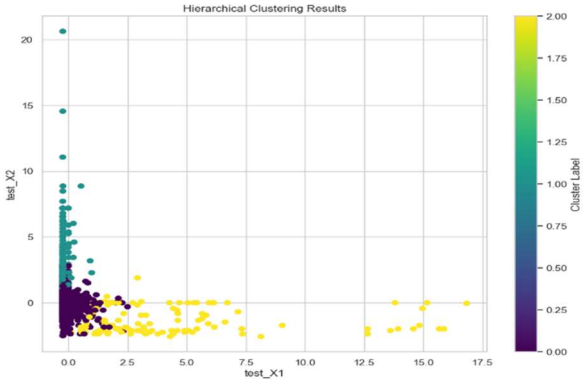
Predictions of Kmeans clusters

The KMeans clustering model was employed with three (3) designated clusters to analyze and categorize the dataset. After fitting the model to the data, the resulting cluster assignments were visually represented through a scatter plot. This visualization provides a clear depiction of how the data points, corresponding to the 'test_x1' and 'test_x2' features, are distributed across the three distinct clusters. Below is the scatter plot that demonstrates the spatial segregation of the clusters based on these features:



5.5.2 Hierachal Clustering

Hierarchical clustering was applied to the dataset using the Agglomerative Clustering function from scikit-learn. After fitting the model, cluster labels were extracted and used to create a scatter plot for visual analysis. Below is the visualization of the hierarchical clustering, which clearly shows three distinct clusters, each represented by a unique color corresponding to their cluster label. This plot highlights the effective segregation among the groups, particularly along the 'test_X2' axis, indicating a successful clustering outcome. The choice of three clusters was validated by the clear demarcation seen in the visualization, suggesting that the hierarchical clustering effectively captured the inherent grouping within the data based on the provided features. Below is the scatter plot that demonstrates the spatial segregation of the clusters based on these features.



5.6 Analysis of Formed Clusters

Starting with an assessment of how specific features—test_X1 and test_X2—are clustered in the supervised learning model, particularly focusing on visualizing the relationship between these features to understand their interaction, and contrasting these observations with the clusters formed through unsupervised learning methods, here are some insights;

KMeans Clustering: Clustered the data into groups based on the inherent similarities in test_X1 and test_X2. This technique formed clusters that mainly separated based on the test_X1 values while showing a varied range of test_X2 values within each cluster.

Hierarchical Clustering: Demonstrated a more stratified grouping, with clusters forming across a gradient of test_X2 values. Notably, data points with the highest test_X2 values clustered tightly at the lower range of test_X1, suggesting a strong inverse relationship between these variables in high-density areas.

5.7 Comparative Analysis with Supervised Learning Outcomes

Alignment with Supervised Outcomes: Clustering results showed remarkable alignment despite the imperfections with the supervised labels. Notably, clusters identified in both KMeans and hierarchical methods did not perfectly correspond to the predefined risk categories (low, moderate, high). For instance, some high-risk areas identified in supervised learning (based on test_X2 values) were dispersed across multiple clusters in unsupervised techniques.

Relationship between test_X1 and test_X2: Clustering effectively highlighted an inverse relationship, especially visible in hierarchical clustering, where higher test_X2 values tended to correlate with lower test_X1 values, aligning somewhat with the risk distribution seen in supervised learning.

5.8 Effectiveness of Clustering Models

The clustering models employed—KMeans and hierarchical clustering—demonstrated substantial effectiveness in discerning natural groupings from the data, independent of any predefined labels. This capability is particularly valuable in exploratory data analysis, enabling the detection of inherent structures and patterns that are not immediately obvious. By uncovering these groupings, the models facilitate a deeper understanding of the dataset's characteristics, revealing dynamics that might otherwise remain obscured in label-constrained analyses. Such insights are instrumental in forming hypotheses about data relationships and guiding further analytical endeavors.

6.0 REFERENCES

Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.

Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.). Springer.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: with Applications in R. Springer Science & Business Media.

Provost, F., & Fawcett, T. (2013). Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media.

Shalev-Shwartz, S., & Ben-David, S. (2014). Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press.

Tan, P.-N., Steinbach, M., & Kumar, V. (2005). Introduction to Data Mining. Pearson Education.