**Methodology**

To generate explanations for the notebook content following the CRISP-DM (Cross-Industry Standard Process for Data Mining) steps, I'll explain the process as observed from the provided content, maintaining simple academic language.

### 1. Business Understanding

This stage is about understanding the project objectives and requirements from a business perspective, then converting this knowledge into a data mining problem definition and a preliminary plan to achieve the objectives.

In the context of the provided notebook, the objective seems to be analysing data from a health survey (specifically focusing on HIV services) to understand service satisfaction levels, identify gaps in service provision, and improve health services based on client feedback. The business goal could be enhancing the quality of healthcare services, ensuring services meet the needs of the community, and increasing overall satisfaction among service recipients.

### 2. Data Understanding

The data understanding phase starts with an initial data collection and proceeds with activities to get familiar with the data, identify data quality issues, discover first insights into the data, or detect interesting subsets to form hypotheses for hidden information.

The notebook indicates the use of a dataset named `imonitor\_1703.csv`, which includes responses from a health survey with various attributes like service satisfaction, awareness of services, challenges in accessing services, etc. The dataset initially contains 46,549 rows and 85 columns, but after cleaning and preprocessing, the shape changes, indicating data quality and structure adjustments.

### 3. Data Preparation

Data preparation tasks are likely to be performed multiple times and not in any prescribed order. Tasks include table, record, and attribute selection, as well as transformation and cleaning of data for modelling tools.

The notebook shows extensive data preparation steps:

Removal of Unnecessary Columns: This step is crucial to streamline the analysis by eliminating data that might not contribute to the model's performance or could potentially introduce noise. For example, free-text fields requiring manual interpretation or fields with high cardinality can complicate the modelling process without adding value.

Renaming of Columns: Improving the readability and consistency of the dataset makes it easier for analysts and stakeholders to understand the data. Clear column names reduce the risk of misunderstandings or errors in data handling.

Dealing with Missing Data: High levels of missing data in certain columns can significantly impact the quality of the analysis. Removing or imputing these values ensures that the models are built on reliable and complete information, thereby improving their accuracy and reliability.

Data Type Conversion: Correcting data types, such as converting strings to datetime objects or categorizing rating scales, ensures that mathematical and logical operations during the modelling phase are applicable and meaningful. It also helps in using certain types of models that require numerical input.

Encoding of Categorical Variables: Machine learning models require numerical input, so categorical variables are encoded into a format that can be interpreted by these models. This step is essential for leveraging the predictive power of categorical data.

### 4. Modelling

In this phase, various modelling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques have specific requirements on the form of data. Therefore, going back to the data preparation phase is often necessary.

The notebook demonstrates the application of multiple machine learning models (CatBoost, LightGBM, XGBClassifier, RandomForestClassifier, LogisticRegression, and SVC) to predict service satisfaction levels. Trying out different algorithms helps in identifying the most effective model based on the problem at hand. Each algorithm has its strengths and weaknesses depending on the nature of the data and the specific task.

Performance Evaluation: The use of metrics like ROC AUC, accuracy, precision, recall, and F1 score allows for a comprehensive assessment of each model's performance. It helps in understanding not just the model's accuracy but also its ability to balance false positives and false negatives, which is crucial in healthcare applications.

Model Selection: Based on the comparative analysis, the best-performing model is selected. This ensures that the deployment phase uses the most effective tool available for making predictions, thereby maximizing the positive impact of the project.

### 5. Evaluation

At this stage in the project, you have built a model (or models) that appears to have high quality, from a data analysis perspective. Before proceeding to final deployment of the model, it is important to evaluate the model more thoroughly, and review the steps executed to construct the model, to be certain it properly achieves the business objectives.

The evaluation in the notebook involves assessing model performance on a test set and examining metrics like the classification report, ROC curves, Precision-Recall curves, and confusion matrices. This helps in understanding the model's generalizability and identifying any potential issues before deployment.

Assessment on Test Set: Evaluating the model on a separate test set provides an unbiased estimate of its performance on unseen data, which is a strong indicator of its generalizability.

Use of Various Metrics: Different metrics can provide insights into different aspects of model performance. For instance, precision and recall are particularly important in healthcare settings where the cost of false negatives might be very high.

ROC and Precision-Recall Curves: These allow for the evaluation of model performance across different threshold settings, offering a more nuanced view than single-number metrics.

Confusion Matrix: Provides a clear visualization of the model's predictions compared to the actual labels, helping identify patterns in misclassifications that could guide further refinement of the model.

### 6. Deployment

In the deployment phase, the cleaned and pre-processed dataset was exported for further use, such as exploratory data analysis (EDA) in tools like Power BI. This step is significant because EDA can uncover patterns, anomalies, and relationships in the data that might not be immediately apparent, providing valuable insights for business decisions and strategy formulation.

Additionally, the best-performing model and its evaluation report were integrated into a Power BI dashboard. This inclusion allows stakeholders to interact with the findings dynamically, understand the predictive performance of the model, and explore how different variables impact service satisfaction. This holistic approach ensures that the insights derived from the model are accessible and actionable, supporting informed decision-making. The dashboard serves as a powerful tool for presenting complex analytical outcomes in an intuitive and understandable manner, facilitating the practical application of these insights in improving health services.

**System Architecture and Implementation**

This chapter describes the architecture and implementation of a system designed to clean a dataset, run a machine learning model for classification prediction, and visualize the results. The system comprises three main components: a Deepnote notebook, a Supabase Postgres database, and a Power BI dashboard.

### System Overview

The system was architected to streamline the process of analysing healthcare survey data, from initial cleaning and processing through to predictive modelling and visualization. The end goal was to enable efficient insights into service satisfaction and identify areas for improvement.

### System Architecture

The system architecture was designed with modularity, scalability, and ease of use in mind. It consists of three primary components:

1. \*\*Deepnote Notebook:\*\* Utilized for data cleaning and execution of the machine learning model.

2. \*\*Supabase Postgres Database:\*\* Served as the storage mechanism for both the cleaned dataset and the outputs of the machine learning model.

3. \*\*Power BI Dashboard:\*\* Employed to visualize the cleaned data, model outputs, and conduct exploratory data analysis (EDA).

#### [Diagram Placeholder: System Architecture Diagram] Insert a diagram that visually represents the system architecture, indicating the flow of data between the Deepnote notebook, Supabase Postgres database, and Power BI dashboard.

### System Implementation Process

#### Deepnote Notebook

The Deepnote notebook was implemented to perform initial data cleaning tasks, including the removal of irrelevant columns, handling missing values, and encoding categorical variables. Following the cleaning process, the notebook executed a machine learning model to predict service satisfaction levels based on survey responses. The model selection process involved evaluating several classifiers, with the best-performing model based on accuracy, precision, recall, and F1 score being chosen for deployment.

Key steps in the notebook implementation included:

- Importing the dataset.

- Performing exploratory data analysis to understand the data's characteristics.

- Cleaning the data to prepare it for modelling.

- Splitting the data into training and testing sets.

- Training multiple machine learning models and selecting the best performer.

- Exporting the cleaned data and model predictions to the Supabase Postgres database.

#### Supabase Postgres Database

The Supabase Postgres database was configured to store the cleaned dataset and the outputs from the machine learning model. Two primary tables were created: one for the cleaned dataset and another for the model predictions. Database triggers and views were not utilized in this implementation; however, they were considered for future work to automate updates and enhance data management.

#### Power BI Dashboard

The Power BI dashboard was developed to provide interactive visualizations of the cleaned dataset and the results of the machine learning model. The dashboard included various charts and filters to enable stakeholders to explore the data from multiple angles, including service satisfaction levels, respondent demographics, and predictors of satisfaction.

Key features of the dashboard included:

- Interactive filters for selecting specific subsets of data.

- Visualizations for model accuracy and performance metrics.

- Charts depicting key findings from the exploratory data analysis.

### Integration Process

The integration process involved exporting the cleaned dataset and model predictions from the Deepnote notebook to the Supabase Postgres database using a direct API call. Once stored in the database, the data was then connected to Power BI through its Postgres database connector. This setup allowed for real-time updates to the dashboard as new data was processed and analyzed in the notebook.

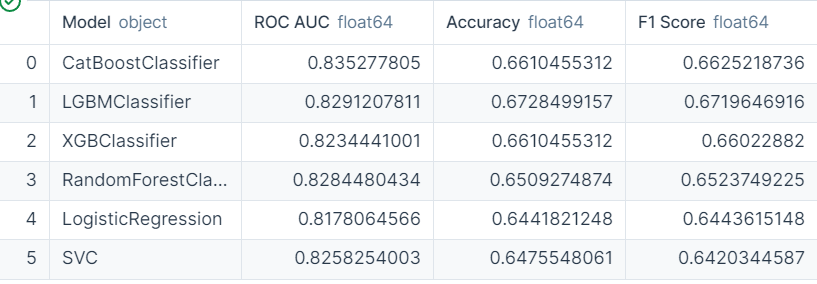
### Challenges Encountered in the Implementation

During the implementation of the system, several significant challenges were faced, most notably related to data interoperability and real-time data processing. A major obstacle encountered was the lack of interoperability between the iMonitor system, which houses the healthcare survey data, and the implemented system comprising the Deepnote notebook, Supabase Postgres database, and Power BI dashboard. The iMonitor system did not offer an API or direct integration feature for automatically exporting the data. This limitation hindered the possibility of creating a real-time application that could directly ingest data from the survey application, necessitating manual data export and import processes. To overcome this challenge, a manual process for exporting data from the iMonitor system and importing it into the Deepnote notebook was established. While this approach did not allow for real-time data updates, it ensured data availability for cleaning, analysis, and modelling. To mitigate the impact of this manual step and make the process as efficient as possible, detailed documentation was created, outlining the steps for data export and import, ensuring consistency and reducing the potential for errors.

**Results and Discussion**

### Machine learning Results

Model Comparisons



The table shows a comparison of several machine learning classifiers based on their Receiver Operating Characteristic Area Under Curve (ROC AUC), Accuracy, and F1 Score.

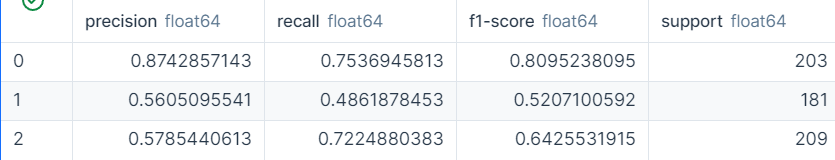
The CatBoostClassifier was identified as the best model due to its superior performance in the ROC AUC metric with a score of approximately 0.8353. The ROC AUC is a comprehensive measure of a model's performance that considers both the true positive rate and the false positive rate across different threshold levels. The CatBoost model's higher AUC value indicates a better balance between sensitivity (true positive rate) and specificity (false positive rate), suggesting it was more effective at distinguishing between the classes compared to the other models.

Although the Accuracy of the CatBoostClassifier is the same as that of the XGBClassifier at approximately 0.6610, the AUC score is a more balanced metric for classification models, especially in scenarios where the class distribution is imbalanced, or the cost of false positives differs from that of false negatives.

Additionally, the F1 Score, which is the harmonic mean of precision and recall, is highest for the CatBoostClassifier at approximately 0.6625. This indicates that it has the best balance of precision and recall compared to the other models, making it particularly useful if the cost of false positives and false negatives is high, as is often the case in many real-world applications.

### CatBoost Model Evaluation

#### Interpretation of Model Performance Metrics



The interpretation of the classification report provided suggests the following for each class in the model's predictions:

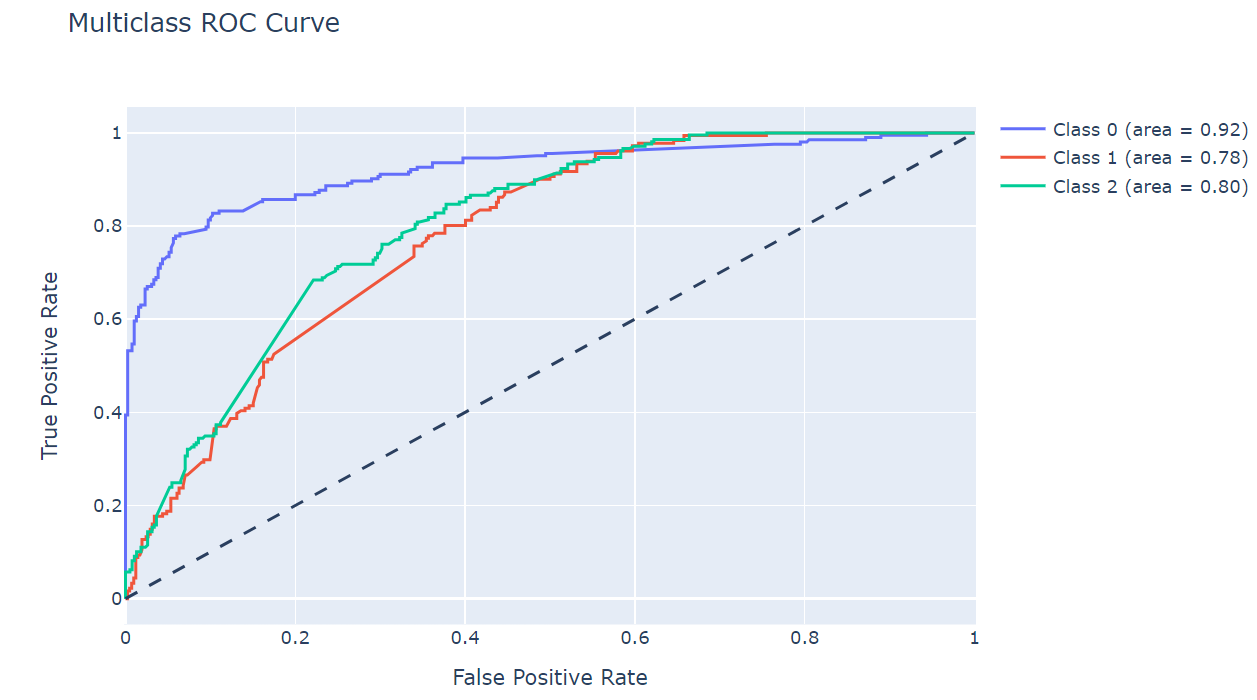
- \*\*Class 0\*\* had the highest precision of approximately 0.874, indicating that the model's predictions for this class were highly reliable; when the model predicted an instance to be Class 0, it was correct about 87.4% of the time. The recall for Class 0 was around 0.754, showing that the model was quite good—correctly identifying about 75.4% of the actual Class 0 instances. The F1-score, which combines precision and recall into a single metric, was highest for Class 0 at approximately 0.809. This suggests a strong overall predictive performance for Class 0, with a good balance between precision and recall. There were 203 instances of Class 0 in the dataset, which provided a substantial number of data points for the model to learn from.

- \*\*Class 1\*\* showed considerably lower precision and recall, at approximately 0.561 and 0.487, respectively. This means that the model was less precise in its predictions for Class 1 and missed a higher proportion of actual Class 1 instances. The F1-score for Class 1 was the lowest among the three classes at around 0.527, indicating a weaker predictive performance for this class compared to the others. There were 181 instances of Class 1, which was a reasonable sample size but might not have been sufficient for the model to learn as effectively as for Class 0.

- \*\*Class 2\*\* had a precision of approximately 0.579, which was better than Class 1 but still lower than Class 0. Its recall was about 0.722, meaning the model was relatively effective at capturing the true Class 2 instances. With an F1-score of around 0.643, Class 2's predictive performance was better than Class 1 but not as strong as Class 0. The number of Class 2 instances in the dataset was 209, the most among the classes, which likely helped in achieving a higher recall.

The model excelled in predicting Class 0, as evidenced by the high scores across all metrics. The model struggled more with Class 1, which had the lowest precision, recall, and F1-score, indicating a need for improvement in distinguishing this particular class. Class 2 was somewhat in the middle, with the model showing a good ability to identify true positives but still having room for improvement in precision. The support values indicated that the dataset was fairly balanced, which generally aids in model training. However, the differences in the model's predictive performance across the classes suggest that factors such as feature discriminability or class complexity may have influenced the outcomes.

#### ROC



The diagram displayed the ROC curves for three different classes, with each curve illustrating the trade-off between the true positive rate (TPR) and false positive rate (FPR) at various threshold settings.

For Class 0, the red curve exhibited an area under the curve (AUC) of 0.92, which indicated a very high level of distinction by the model. This meant that the model was highly capable of distinguishing between Class 0 and non-Class 0 instances. A high true positive rate was achieved while maintaining a low false positive rate, which is an indication of a strong predictive performance for this class.

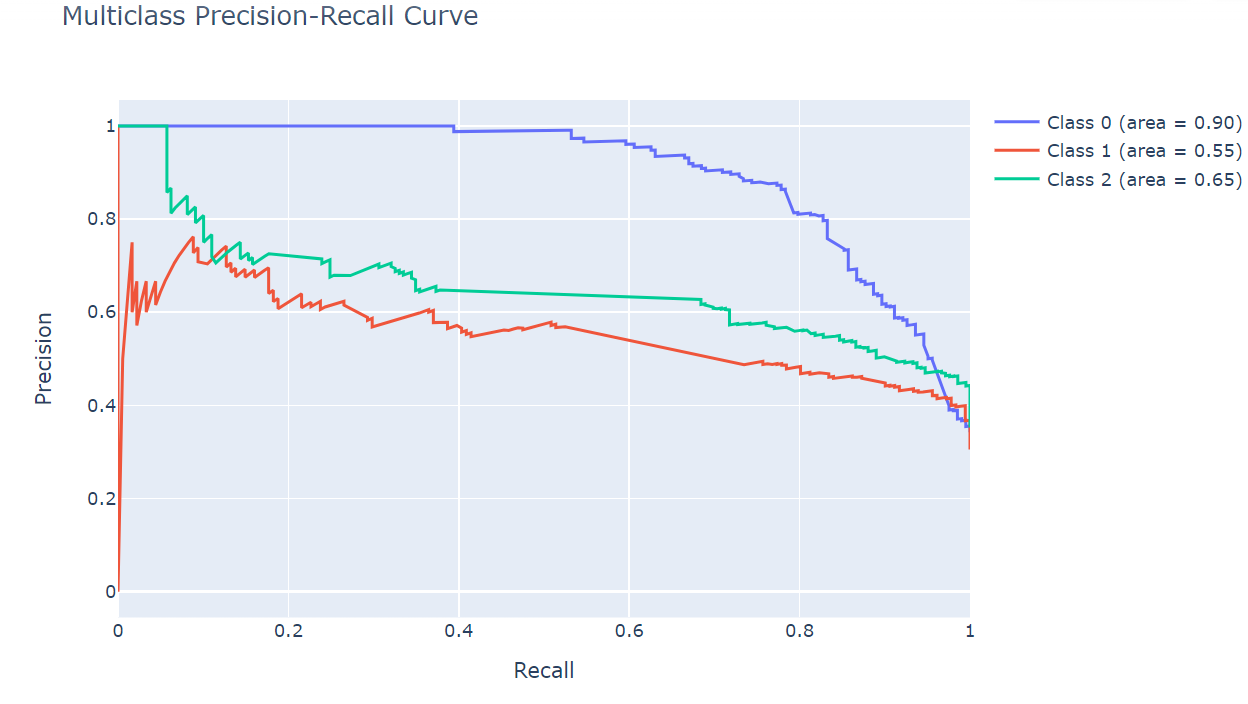
The green curve, representing Class 1, had an AUC of 0.78. Though this value was lower than that of Class 0, it still reflected a reasonably good performance of the model in classifying Class 1 instances. The model was less effective at discriminating Class 1 instances from non-Class 1 instances compared to Class 0, as evidenced by a lower true positive rate and a higher false positive rate.

Class 2, represented by the blue curve, showed an AUC of 0.80, placing its performance between Class 0 and Class 1. The model's ability to distinguish Class 2 instances from non-Class 2 instances was good, as indicated by the AUC score which was closer to 1 than to 0.5.

The ROC Curve is a useful tool for comparing the performance of a classifier across multiple classes. The closer the curve follows the left-hand border and then the top border of the ROC space, the more accurate the test. Conversely, a curve that comes close to the 45-degree diagonal of the ROC space is indicative of a less accurate test. In the case of the provided diagram, none of the classes approached the diagonal line, which represented the performance of a random guess, indicating that the model's predictions were better than random chance for all classes.

In summary, the ROC curve diagram from the model evaluation indicated that the classifier performed well, especially for Class 0, and offered a good balance between sensitivity and specificity for Class 1 and Class 2. This kind of performance is particularly valuable in applications where the cost of false positives and false negatives varies between classes, as it can inform the adjustment of threshold values to meet specific operational criteria.

### Precision-Recall Curve



Each line in the graph represents one of the classes (Class 0, Class 1, Class 2) and their respective precision-recall curves. These curves demonstrate the trade-off between precision (the y-axis) and recall (the x-axis) for the different threshold levels used in classifying instances into the respective classes.

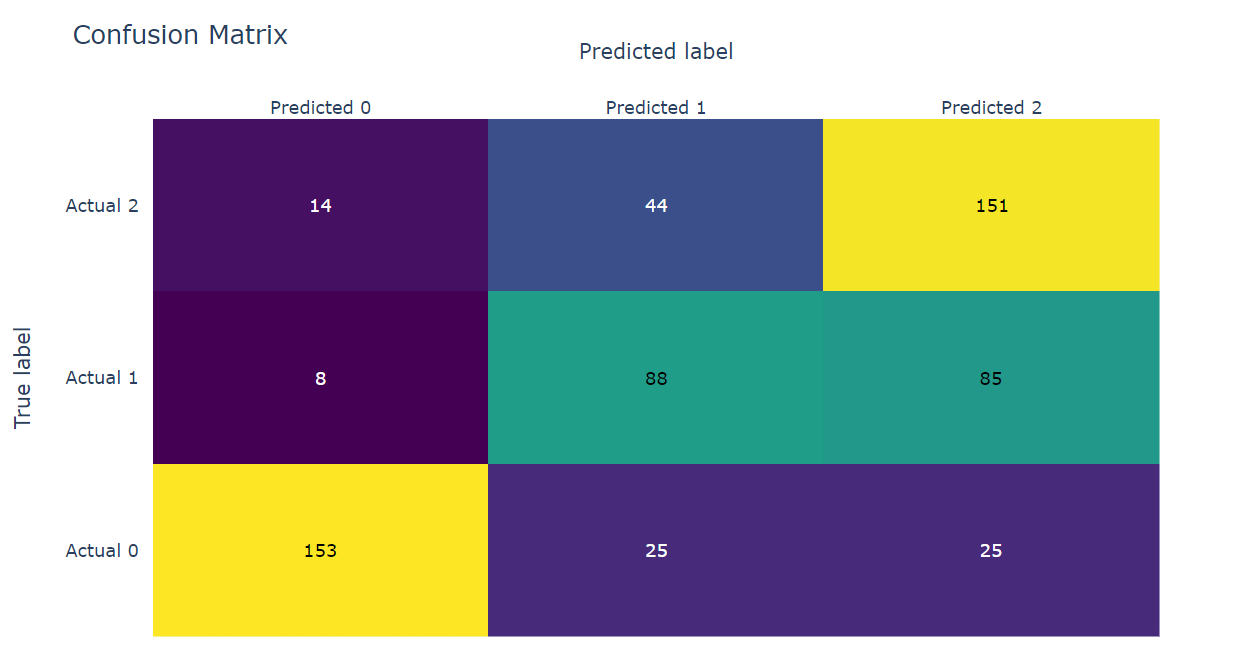
Class 0 (blue line): With an area under the curve (AUC) of 0.90, this class shows a high level of performance. It starts with a high precision, which indicates that when the model predicts an instance as Class 0, it is very likely to be correct. The recall starts high but decreases as precision decreases, suggesting that as the model becomes less stringent (lower threshold), it captures more true Class 0 instances but also makes more false positive errors.

Class 1 (red line): The AUC for Class 1 is 0.55, which is significantly lower than Class 0, indicating a moderate level of performance. The precision for Class 1 is lower than for Class 0 throughout most of the recall range, suggesting that the model has a harder time correctly identifying instances of Class 1 without making as many false positive errors.

Class 2 (green line): This class has an AUC of 0.65, which is better than Class 1 but not as high as Class 0. The precision for Class 2 is generally higher than for Class 1, indicating that the model is more confident in its predictions for Class 2 than for Class 1. The recall for Class 2 is higher than for Class 1 at lower levels of precision but then converges as the threshold is lowered.

The AUC for each class provides a single measure of performance across all threshold levels. In this case, Class 0’s model performance is quite strong, Class 2 is moderate, and Class 1 needs improvement. This graph can be interpreted as showing the ability of the model to distinguish between the classes with the understanding that for Class 1, the model is less capable of making correct positive predictions (precision) while correctly identifying all relevant instances (recall). In contrast, Class 0 is well-predicted, with both high precision and recall. Class 2 falls in between, with performance being better than Class 1 but not as high as Class 0.

#### Confusion Matrix Analysis



In this section, we present the evaluation of the classification model using a confusion matrix, which offers a detailed view of the model's predictive performance across the different classes. A confusion matrix juxtaposes the predicted class labels against the actual true class labels, providing a visual depiction of the model's accuracy and the types of errors it made.

### X.X.1 Interpretation of the Confusion Matrix

The rows of the matrix represent the actual classes of the dataset, and the columns represent the predicted classes as outputted by the model. The main diagonal of the matrix (from top left to bottom right) contains the counts of correct predictions, also known as the true positives for each class. These counts were 153 for Class 0, 88 for Class 1, and 151 for Class 2, which indicates that the model was most adept at accurately identifying instances of Class 0 and Class 2.

Off-diagonal elements represent misclassification errors. Notably, the model tended to misclassify between Class 1 and Class 2, with 44 instances of Class 2 being incorrectly predicted as Class 1. Similarly, 25 instances of Class 0 were incorrectly labeled as Class 2 by the model. The higher misclassification rates for Class 1 suggest that distinguishing features for this class may not be as distinct or well-learned by the model compared to features for Class 0 and Class 2.

##### X.X.2 Misclassification Analysis

Upon closer inspection of the confusion matrix, the misclassifications bring to light specific areas where the model's performance can be enhanced. For example, the model's confusion between Classes 1 and 2 may be indicative of overlapping feature spaces or insufficiently discriminative training data for these classes. It could also suggest a need for additional features that could better distinguish between these classes or an imbalanced dataset where Class 1 is underrepresented.

The relatively lower number of misclassifications for Class 0 highlights the model's effectiveness in recognizing this class, which could be attributed to more distinct class features or a more representative sample of Class 0 instances in the training set.

##### X.X.3 Strategies for Improvement

To address the misclassification issues identified in the confusion matrix, several strategies can be implemented:

1. \*\*Feature Engineering:\*\* Enhancing the feature set by adding new features or transforming existing ones could improve the model's ability to differentiate between Classes 1 and 2.

2. \*\*Resampling Techniques:\*\* If class imbalance is detected, resampling techniques such as oversampling the minority class or undersampling the majority class could be applied to provide a more balanced training set.

3. \*\*Model Tuning:\*\* Adjusting the hyperparameters of the model or experimenting with different algorithms might lead to better performance, particularly for the classes that the current model struggles with.

4. \*\*Post-Processing Adjustments:\*\* Threshold-moving or probability calibration could be considered to refine the decision boundary for the predictive model.

In summary, the confusion matrix has provided crucial insights into the model's strengths and weaknesses, guiding the focus toward targeted improvements that could result in a more accurate and reliable predictive performance across all classes.

### Feature Importance Analysis

An integral part of understanding the predictive model's behaviour is identifying which features have the most influence on the model's decisions. Feature importance analysis provides insights into the relative importance of each feature in the context of the model. The following bar chart illustrates the top 15 features ranked by their importance as determined by the machine learning model:

A graph with blue and white text

Description automatically generatedThe feature "ExpectedHIVServices\_ARTMedicine" was found to be the most influential in the model's predictions, indicating its strong predictive power for the outcome variable. The prominence of this feature underscores the importance of ART medicine availability in the context of HIV service provision and its impact on the model's classification decisions.

The features associated with GBV awareness measures such as "GBVAwarenessMeasures\_FacilitystafftrainedonGBV" and "GBVAwarenessMeasures\_PresenceofGBVDesk" were also highly significant, suggesting that initiatives related to GBV are key predictors in the dataset. This reflects the model's utilization of GBV-related variables as indicators for predicting outcomes, highlighting the interplay between GBV initiatives and service satisfaction or delivery.

Accessibility to services, indicated by features "AccessChallenges\_No" and "AccessChallenges\_Yes", also featured prominently, signifying the role of service accessibility in the model's predictions. The binary nature of these features implies that the model differentiates the impact of having or not having access challenges on the predicted outcome.

Awareness of services and rights, represented by "ServicesAwareness\_Yes" and "RightsAwareness\_Yes", were found to be important predictors. This may reflect the model's reliance on respondents' awareness levels to infer the likelihood of certain outcomes.

Lastly, operational features like "OperationDays\_Friday" and "OperationTimes\_8-5" suggest that the model considers the timing of services as influential, which could be related to the convenience and availability of services to respondents.

##### X.X.2 Implications of Feature Importance

The identified feature importances have important implications for strategic decision-making and policy development in the context of HIV service provision. They suggest areas that could be leveraged or improved to enhance service delivery and satisfaction.

By understanding which features are deemed most important by the model, stakeholders can focus their efforts on these areas to effect positive change in service delivery outcomes. For instance, strengthening ART medicine availability and GBV-related services could have a significant impact on overall service satisfaction.

##### X.X.3 Limitations of Feature Importance

While feature importance provides valuable insights, it is also subject to the limitations of the modelling approach. It is crucial to recognize that these importances are model-specific and may not capture causal relationships. Caution should be exercised when interpreting these results, and they should be considered in conjunction with domain knowledge and expert input.

### Comparison between Homa Bay and Kilifi

A screenshot of a computer

Description automatically generated

A graph with blue and white text

Description automatically generated

Based on the feature importance charts for Kilifi and Homa Bay, several observations can be made regarding the differences in factors that potentially contribute to the HIV prevalence rates between the two counties.

For Kilifi, which had the lower HIV prevalence rate, the top features include nutritional services and treatment interruption follow-up. These suggest a focus on consistent treatment and nutritional support within the HIV services provided. The emphasis on Partner Notification Support (PNS) indicates a robust contact tracing system that may help in controlling the spread of HIV. The presence of features such as GBV awareness measures and service denial suggests that awareness and support services are key aspects of the HIV response in Kilifi.

In Homa Bay, the features with the highest importance were related to Voluntary Medical Male Circumcision (VMMC) and Key Population (KP) services, which are targeted interventions known to reduce the risk of HIV transmission. The importance of presumptive TB screening implies a strong integrated response to TB and HIV, which is critical given the high rate of co-infection. Features such as confidentiality and counselling point towards a focus on the quality of care and privacy, which may improve the uptake of services.

The difference in feature importances between the two counties could be explained by the differing strategies and focus areas on their response to HIV. Kilifi's approach seems to be centred around nutrition and continued care, which might be effective in a setting with a lower prevalence rate, where the management of existing cases is crucial. On the other hand, Homa Bay's focus on prevention strategies, like VMMC and TB screening, could be a response to a higher prevalence rate, where reducing transmission is a priority.

Additionally, the importance of confidentiality in Homa Bay could relate to the social stigma associated with a higher prevalence rate, making privacy a significant concern for individuals seeking testing and treatment. The prominence of service denial as a feature in both counties could indicate that despite the prevalence rates, service access is a common issue, perhaps due to resource limitations or social barriers.

Overall, these feature importances suggest that the HIV response in both counties is tailored to their specific epidemiological and social contexts. Kilifi, with a lower prevalence, focuses on treatment adherence and support, while Homa Bay, with a higher prevalence, emphasizes preventive measures and integrated care approaches. Understanding these nuances is crucial for developing targeted interventions and allocating resources effectively.

Prevalence rates link: https://nsdcc.go.ke/kasf-areas-of-focus/

### Conclusion

**Conclusions, Recommendations, and Future Work**

### Conclusion

The system was successfully implemented, achieving the objective of providing a streamlined process for data analysis, from cleaning and modelling in a Deepnote notebook, through storage in a Supabase Postgres database, to visualization in a Power BI dashboard. This implementation not only facilitated a deeper understanding of the healthcare survey data but also laid the groundwork for ongoing improvements in service satisfaction analysis.

The challenges encountered during the implementation of this system underscored the importance of flexibility and adaptability in the face of technical limitations. By developing effective solutions to these challenges, the project successfully leveraged the available tools and technologies to achieve its objectives, albeit with some compromises on real-time data processing capabilities. Future enhancements will focus on exploring alternative solutions to achieve greater interoperability and automation, further streamlining the data analysis process.