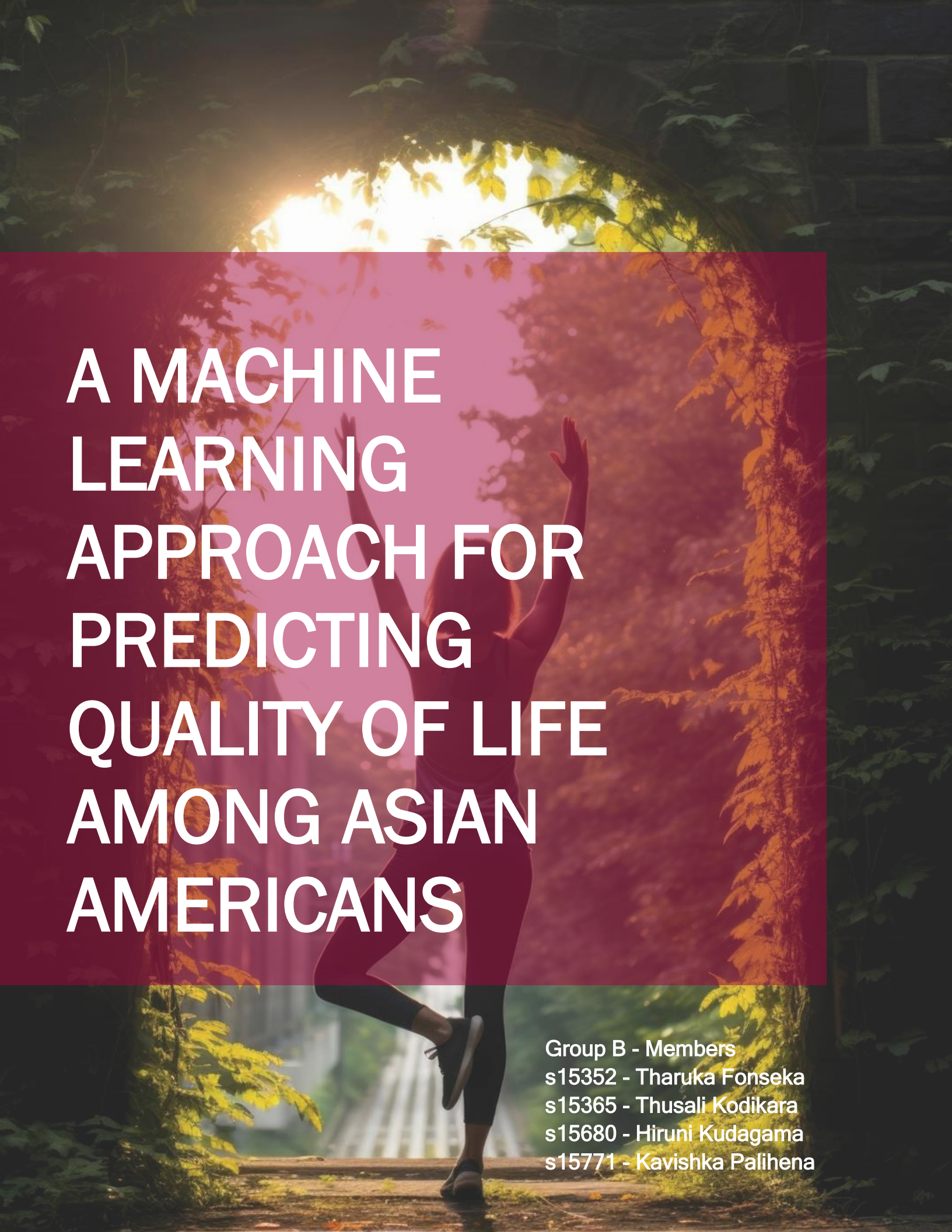


A MACHINE LEARNING APPROACH FOR PREDICTING QUALITY OF LIFE AMONG ASIAN AMERICANS

Group B - Members
s15352 - Tharuka Fonseka
s15365 - Thusali Kodikara
s15680 - Hiruni Kudagama
s15771 - Kavishka Palihena

Introduction

To assess individuals' well-being in society, various measures can be used, one of which is Quality of Life (QoL). QoL encompasses physical, mental, social, and environmental factors affecting a person's life. In this project, we analyze the factors related to the QoL of Asian Americans, the fastest-growing minority group in the United States, who face unique challenges. Identifying the factors significantly impacting their quality of life will help us understand these challenges and provide effective solutions.

Problem Statement

The objective of this project is to identify the main factors that affect the Quality of Life (QoL) of Asian Americans and to evaluate them using appropriate machine learning techniques.

Data Set

The dataset for this project was obtained from DATA.GOV and is based on a survey conducted by the government of Austin, Texas. It contains 2,609 rows and 231 columns, featuring both numerical and categorical variables. It contained 41 numerical variables and 190 categorical variables.At first glance, we identified 'Survey ID' as the dataset's primary key; due to its messiness, with a total of 2,609 observations and 231 variables spread across 7 sections as mentioned in the description file, we split it into 7 sections using 'Survey ID' as the index, enabling focused analysis of each section's variables and efficient extraction of insights without data loss. This enabled us to give individual attention to each variable.

Data Preprocessing

Prior to analyzing the data, preprocessing steps were necessary to ensure data quality and derive meaningful interpretations. These steps included handling missing values, encoding categorical variables, and reducing unnecessary features in each section.

Row Deletion : The dataset underwent a cleaning process where rows entirely filled with missing values were removed and the rows where the response variable 'Quality of Life' were missing were deleted from the dataset..**Column Deletion:** The proportion of missing values was calculated for each column in the dataset. To minimize the loss of information a threshold of 0.2 (20%) was set. Columns with more than 20% missing values were identified and removed from the dataset. Initially we graphically visualized the numerical variables and identified that there were few variables that we could convert into object data type as it contained level data. Also we didn't observe any outliers from the graphical analysis. Then for each section the value counts for each categorical variable were taken and carefully observed whether there were any abnormalities in the level. Indeed, we identified some category levels named '0', '2', '3', '4', and '11' instead of 'Yes', and 'No' for some columns. So those cells were properly replaced with missing values. Some other nominal categorical levels also had level names like '5', '6', '10', '11', and '33' that didn't give any meaning. They too were considered as missing. Another important finding was that in some columns there were some levels with abnormal level names which had high value counts. For example, -Status of Ownership [3] - 78 - Housing [5] - 53We figured out that these were not missing values but belong to the 'Other' category which had a text description in the dataset. So, these level names were modified as "Other".

Handling Missing Values

Missing values in categorical features were imputed using the mode, as it best represents the most common category in each variable. For numerical variables, missing values were imputed based on the data distribution—mean imputation was applied for symmetric distributions, and median imputation for skewed distributions. This approach preserved the central tendency of the data without introducing significant bias.

Duplicate and Index Handling No duplicate entries were found during inspection.

Merging Columns Combined the 3-1-1 columns and the 9-1-1 columns into a single column '3-1-1_9_1_1'. The resulting column marks '2' if an individual is aware of both the emergency services, and '1' if he/she only knows about one service, or if not, aware it marks '0'.After Applying Cramer's V heatmap for the categorical variables we observed a high correlation between variables that were placed consecutively. Specific columns were aggregated based on themes to create new composite variables to remove multicollinearity. For example,

- **Awareness of Services**: Marks respondents as "Aware" if they used any listed services, such as EMS Classes or Library Internet Access.
- **Civic Engagement**: Labels respondents as "Engaged" if they participated in activities like Public Meetings or City Elections.
- **Communication Methods**: Categorized into City-based, non-city-based Ethnic, and Non-City-based General Sources.
- **Mode of Transport**: Combined transportation types into a single categorical variable.

The newly created variables were kept, and the variables used to merge were then deleted from the dataset. All the separately preprocessed sections were then merged together by using the index as the 'Survey ID'. The resulting dataset altogether contained 148 variables.

Encoding Categorical Variables

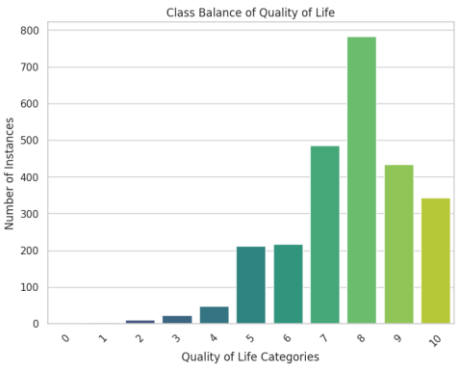
- **Label Encoding**: Basic categorical variables were label-encoded to convert text into numeric form.
- **Ordinal Encoding**: For ordinal variables, custom encoding was applied to maintain the inherent order. This was achieved by assigning numerical values based on the logical order of levels in each ordinal column.

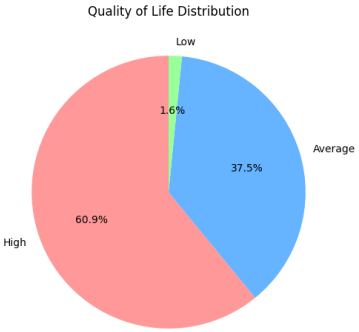
Dropping Irrelevant Features

- Columns where all values were unique were dropped, as they did not contribute meaningful patterns or variability to the analysis. Variables like 'Satisfied with Life 1' were also dropped.

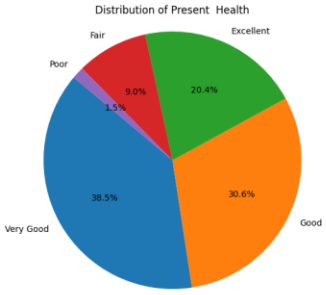
Descriptive Analysis

Univariate

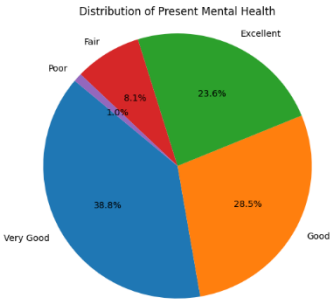




Initially the response variables had 11 categories where a majority of the observations were in the last few categories. After recategorizing them still the high category has the largest amount of observations.

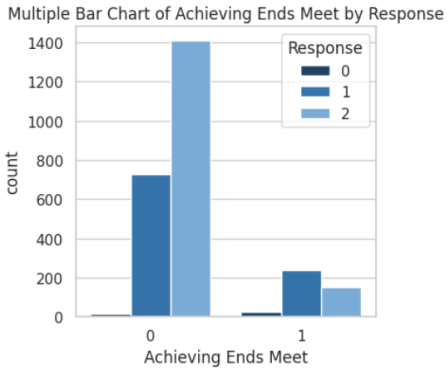


When considering this pie chart of the overall present health condition of individuals, most of them are in good and very good conditions. Only about 1.5% are having poor health.

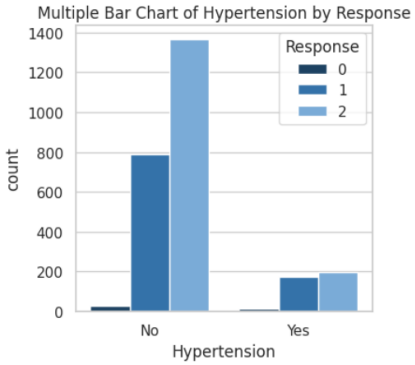


Mental/emotional health conditions also have a similar distribution. But the proportion with excellent mental health are higher than the ones having excellent overall health. The amount with

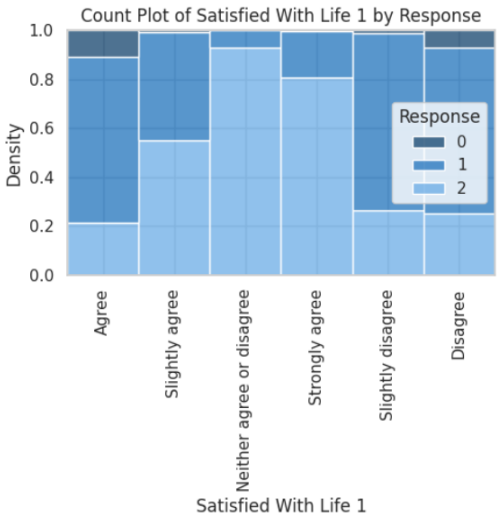
Bivariate Analysis



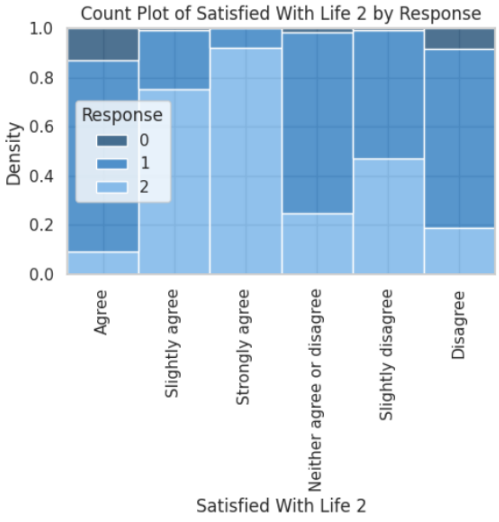
This plot represents whether the monthly household income is able to make ends meet (earn just enough money to live on) and the quality of life. Most houses do not have enough money for living but still have a high quality life.



When considering this plot regarding whether the individuals have hypertension, it shows that most people without hypertension are having high and average quality of lives. Yet from the ones who are having hypertension most of them have rated that they have a high quality life.

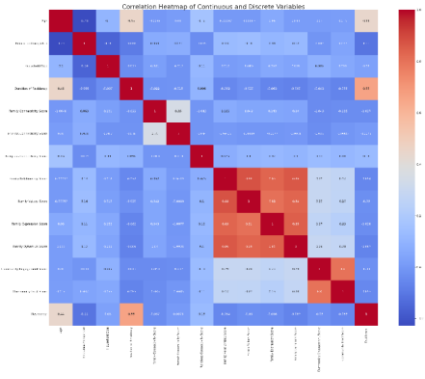


Individuals were asked the statement ‘In most ways my life is close to my ideal’, where they have rated it from agree to disagree. Individuals who strongly agree and neither agree or disagree are the ones who have mostly given a high rating for their quality of life. Ones who have agreed have average quality of life.



When asked to rate the statement ‘I am satisfied with my life’, most of the individuals who agree (strongly and slightly) have high quality lives. Even though some disagree with this statement, many of them have mentioned that they have high or average quality lives.

Multicollinearity To check for multicollinearity, we divided our data into two parts: categorical variables and numerical variables.**Numerical Variables:** We used the Pearson correlation coefficient to see how strongly two numerical variables are related. A heatmap was created to easily spot which variables have strong correlations.



We observed that several variables exhibited significant relationships with one another:

Family-Related Scores:

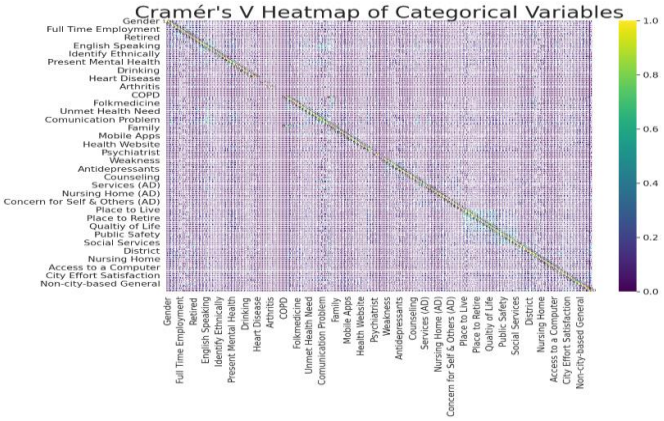
The **Family Relation Score**, **Family Value Score**, **Family Expression Score**, and **Family Dynamic Score** showed notably high multicollinearity

Community-Related Scores:

Similarly, the **Community Engagement Score** and **Community Trust Score** also exhibited significantly high multicollinearity.

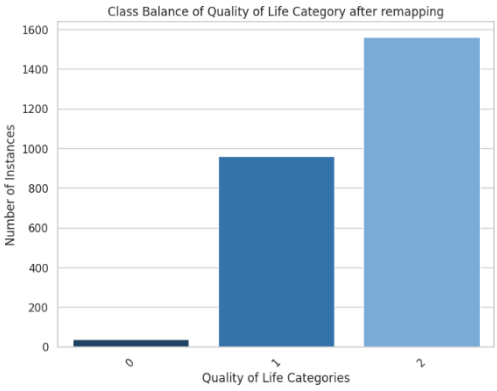
Categorical Variables: For categorical variables, we used Cramér's V to measure the strength of the relationship between pairs of variables. We also used a heatmap here to quickly see any strong connections among the categorical variables.

	Variable 1	Variable 2	Cramér's V
1620	English Speaking	Preferance	0.393035
12282	Raising Children	Qualtiy of Life	0.392265
13193	Qualtiy of Life	Raising Children	0.392265
1846	Familiarity with America	English Speaking	0.390995
1585	English Speaking	Familiarity with America	0.390995



The top five associations showed values between 0.3 and lower, indicating generally low associations among the variables. **Cramér's V Values**: The highest values were around 0.3, pointing to weak associations. Since there were no strong associations, we decided that further analysis of these categorical variables was unlikely to provide useful insights, so no additional analysis was conducted beyond the initial review.

Response Variable. The response variable, Quality of Life, originally had 11 levels. Some levels had very few observations, which impacted model performance. To fix this imbalance and make interpretation easier, we grouped Quality of Life into three categories based on terciles:**High Quality of Life (2)**: Levels 8–10.**Moderate Quality of Life (1)**: Levels 4–7**Low Quality of Life (0)**: Levels 0–3 This approach simplified the response variable into three clear groups, enhancing model performance and making quality-of-life outcomes easier to interpret.



Important results of advanced analysis

The data set was split into training (80% of original data) and test (20%) tests in the pre-processing stage and a few select models were fitted on the training data and tested on the test data.

Model Selection and Ensemble Techniques

To predict the quality of life among Asian Americans (low, moderate, high), we utilized various machine learning models, including individual classifiers and ensemble techniques, to enhance predictive accuracy:

Individual Classifiers: We employed various individual classifiers to predict Quality of Life, including Logistic Regression for its simplicity, Support Vector Machine for optimal class separation, Decision Trees for capturing non-linear relationships, and ensemble methods like Random Forest and XGBoost for improved accuracy. Other models included K-Nearest Neighbors for local patterns, AdaBoost and Gradient Boosting for enhancing performance, Multi-Layer Perceptron for complex relationships, and Naive Bayes for efficient multi-class classification.

Ensemble Techniques We employed a Stacking Classifier to enhance prediction accuracy for quality of life among Asian Americans. This approach integrates predictions from multiple algorithms, including Random Forest, LightGBM, and XGBoost, through a meta-

classifier.Trained with 5-fold cross-validation on the preprocessed dataset, the Stacking Classifier demonstrated superior performance compared to basic models. It effectively captured complex relationships while minimizing the risk of overfitting.

Evaluation Metrics Used: We evaluated our models using four key metrics such as Accuracy, Precision, Recall and F1 Score. These metrics helped us understand model performance and identify strengths and weaknesses in quality-of-life predictions.

Evaluation of Classification Models on Original Training Set Upon completion of preprocessing, we fitted the above models to the data. The table below presents the performance metrics obtained from these models without hyperparameter tuning (using default parameters) The **Gradient boosting** model emerged as the top performer with evaluation metrics, accuracy: 74.07%, precision: 72.6%, F1-score: 74.1% and recall 73.3%.

Original Data Set				
Model	**Accuracy**	**Precision**	**Recall**	**F1 score**
Logistic Regression	0.7232	0.717	0.723	0.72
DT	0.6784	0.682	0.678	0.68
RF	0.7407	0.726	0.741	0.733
XGB	0.729	0.72	0.729	0.724
ADB	0.694	0.679	0.694	0.685
GB	0.7212	0.709	0.721	0.715
MLP	0.694	0.684	0.694	0.689
SVM	0.7251	0.712	0.725	0.718
KNN	0.694	0.677	0.694	0.684
Naïve Bias	0.7251	0.712	0.725	0.718
Stacking Classifier	0.73	0.72	0.73	0.73

Handling Class Imbalance and Evaluation of Models on Data Set after applying different sampling techniques

We identified a significant class imbalance in our dataset, with 1,562 instances for High Quality (2), 963 for Moderate Quality (1), and only 40 for Low Quality (0). To mitigate this issue, we employed the Synthetic Minority Over-sampling Technique (SMOTE) and upsampling methods. After balancing the classes, we evaluated the same algorithms using the established performance metrics.

Quality of Life	Label	Count
High Quality	**2**	**1562**
Moderate Quality	**1**	**963**
Low Quality	**0**	**40**

Evaluation of Classification Models on Training Set After Applying Smote.

We fitted the same set of models to SMOTE data. The table below presents the performance metrics obtained from these models without hyperparameter tuning (using default parameters) on SMOTE data: The **Stacking Classifier model** emerged as the top performer with evaluation metrics, accuracy: 74.46%, precision: 73.8%, F1-score: 74.5% and recall: 74.1%.

SMOTE Data Set				
Model	**Accuracy**	**Precision**	**Recall**	**F1 score**
Logistic Regression	0.706	0.711	0.706	0.708
DT	0.6257	0.637	0.626	0.631
RF	0.7349	0.733	0.735	0.732
XGB	0.7446	0.738	0.745	0.741
ADB	0.6784	0.679	0.679	0.679
GB	0.725	0.716	0.725	0.721
MLP	0.704	0.699	0.704	0.701
SVM	0.6023	0.683	0.602	0.628
KNN	0.7115	0.711	0.712	0.71
Naïve Bias	0.6316	0.698	0.632	0.656
Stacking Classifier	**0.7446**	**0.738**	**0.745**	**0.741**

Evaluation of Classification Models on Training Set After Applying Oversampling.

Oversampling Data Set				
Model	**Accuracy**	**Precision**	**Recall**	**F1 score**
Logistic Regression	0.686	0.706	0.686	0.695
DT	0.6374	0.635	0.637	0.636
RF	0.71	0.695	0.71	0.71
XGB	0.7349	0.721	0.735	0.727
ADB	0.6862	0.689	0.686	0.687
GB	0.7193	0.719	0.719	0.718
MLP	0.708	0.696	0.708	0.701
SVM	0.641	0.654	0.641	0.647
KNN	0.715	0.723	0.715	0.717
Naïve Bias	0.17	0.807	0.175	0.265
Stacking Classifier	**0.73**	**0.72**	**0.73**	**0.72**

The Stacking Classifier model emerged as the top performer with evaluation metrics, accuracy: 73%, precision: 72%, F1-score: 72% and recall: 73%

We fitted the same set of models with hyperparameter tuning on three datasets: the original dataset, a dataset balanced using the Synthetic Minority Over-sampling Technique (SMOTE), and an oversampled dataset. This approach helped evaluate the models' performance and robustness across different data distributions, identifying the most effective configurations for each dataset.

Hyperparameter Tuning on Original Dataset

The table below presents the performance metrics obtained from these models with hyperparameter tuning:The **Stacking Classifier model** emerged as the top performer
with evaluation metrics,accuracy: 75%, precision: 75%, F1-score: 75% and recall: 74%.

Hyper Parameter tuning on Original Data Set				
Model	**Accuracy**	**Precision**	**Recall**	**F1 score**
Logistic Regression	0.737	0.721	0.737	0.728
DT	0.702	0.691	0.702	0.696
RF	0.725	0.71	0.725	0.716
XGB	0.731	0.716	0.731	0.723
ADB	0.7388	0.7222	0.739	0.727
GB	0.7349	0.722	0.735	0.728
MLP	0.692	0.7	0.692	0.694
SVM	0.7193	0.702	0.719	0.707
KNN	0.7193	0.706	0.719	0.712
Naïve Bias	0.1774	0.791	0.177	0.267
Stacking Classifier	**0.75**	**0.75**	**0.75**	**0.74**

Hyperparameter Tuning on SMOTE Dataset

Hyper Parameter tuning on SMOTE Data Set				
Model	**Accuracy**	**Precision**	**Recall**	**F1 score**
Logistic Regression	0.737	0.721	0.737	0.728
DT	0.7018	0.691	0.702	0.696
RF	0.7251	0.71	0.725	0.716
XGB	0.731	0.716	0.732	0.723
ADB	0.7388	0.722	0.739	0.727
GB	0.735	0.722	0.735	0.728
MLP	0.692	0.7	0.692	0.694
SVM	0.7193	0.719	0.707	0.702
KNN	0.7193	0.769	0.801	0.785
Naïve Bias	0.1774	0.791	0.177	0.267
Stacking Classifier	**0.73**	**0.72**	**0.73**	**0.72**

Ada boosting and Stacking model performance well under smote data set with hyper parameter tuning.

Hyperparameter Tuning on Upsampled Dataset

The table below presents the performance metrics obtained from these models with hyperparameter tuning:Gradient Boost method emerged as the best model among the models with
75% accuracy.
73.9% precision and
75% recall and
74.44% f1- score.

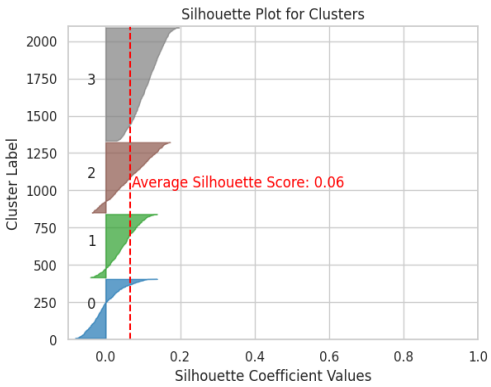
Hyper Parameter tuning on Oversampled Data Set				
Model	**Accuracy**	**Precision**	**Recall**	**F1 score**
Logistic Regression	0.686	0.706	0.686	0.695
DT	0.637	0.637	0.636	0.637
RF	0.721	0.708	0.721	0.714
XGB	0.734	0.48	0.48	0.479
ADB	0.6959	0.706	0.696	0.699
GB	**0.75**	**0.739**	**0.75**	**0.744**
MLP	0.6959	0.684	0.696	0.688
SVM	0.692	0.7	0.692	0.694
KNN	0.6998	0.542	0.433	0.18
Naïve Bias	0.1754	0.807	0.175	0.265
Stacking Classifier	0.74	0.73	0.74	0.72

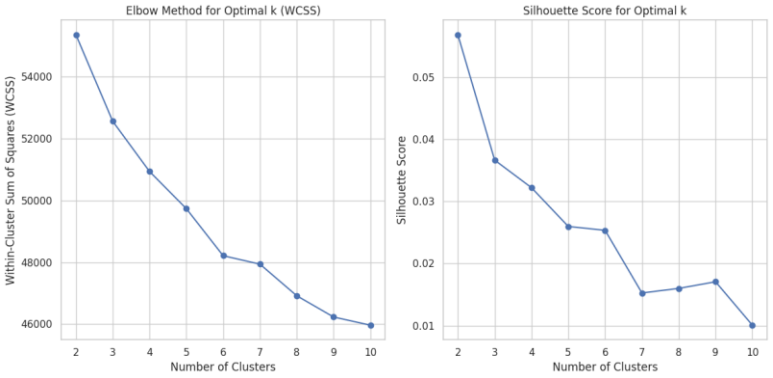
Clustering Analysis: Identification of Patterns in Mixed-Type Data

Given the mixed nature of our dataset with both categorical and numerical variables, we applied three clustering algorithms: K-Prototypes, K-Medoids, and KMeans. Each method involved a tailored approach to effectively handle the different data types. Below, we outline each algorithm, its configuration, and the evaluation metrics used to determine the optimal number of clusters..

K-Prototypes Clustering

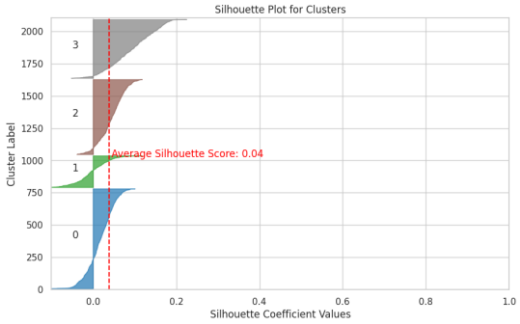
K-Prototypes Clustering is designed for mixed-type data, using Euclidean distance for numerical variables and Hamming distance for categorical variables. Categorical columns were converted to category types, and the 'Cao' method was used for cluster initialization. The algorithm minimizes total cost by combining both distance metrics, testing k values from 2 to 10. Evaluation metrics included the Within-Cluster Sum of Squares (WCSS) for elbow curve analysis and silhouette scores for cluster cohesion. Although 4 clusters were deemed optimal, an average silhouette score of around 0.06 indicated poor separation among them.

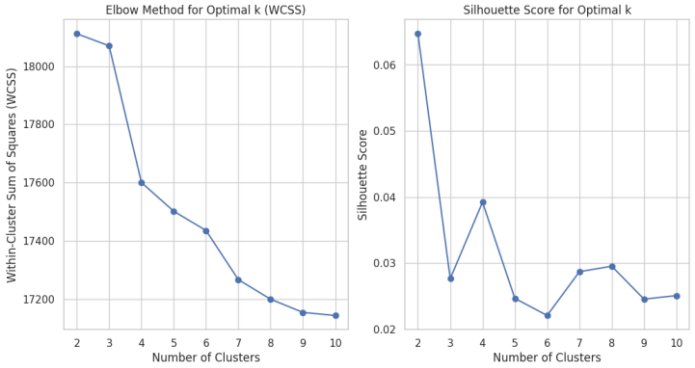




K-Medoids Clustering

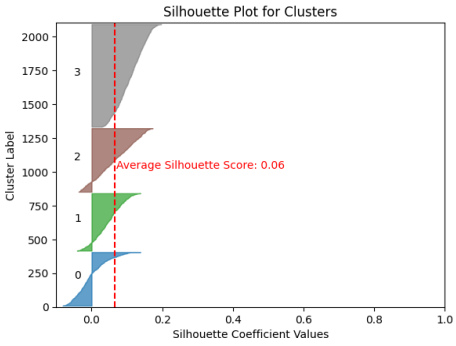
K-Medoids is similar to KMeans but less sensitive to outliers, using medoids instead of centroids for both categorical and numerical data. After preprocessing categorical variables, the algorithm minimized dissimilarity to the nearest medoid with k varying from 2 to 10. Evaluation metrics included Within-Cluster Sum of Squares (WCSS) for elbow analysis and silhouette scores for cluster quality. While 4 clusters were optimal, a near-zero silhouette score indicated poor distinctiveness, showing that points in different clusters were nearly as close as those within the same cluster.

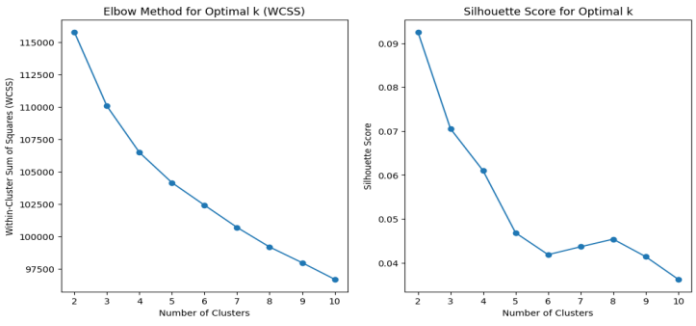




KMeans Clustering:

The KMeans algorithm was adapted for mixed-type data by combining Euclidean and Hamming distances. Categorical columns (12 to 138) were converted to category types, and a custom distance function was established. Hyperparameter tuning varied the number of clusters (k) from 2 to 10. Evaluation metrics included Within-Cluster Sum of Squares (WCSS) and silhouette scores, which indicated a lack of natural clustering structure and suggested that meaningful separations may not exist.





Interpretation of Low Silhouette Scores and Implications

Low silhouette scores indicate that the dataset lacks a meaningful clustering structure, suggesting that variables and observations do not form distinct groups. Consequently, clusters were not further analyzed, as there is no evidence of inherent groupings within the data.

Selection of Best Model

We define the best model as the one with the highest Accuracy, F1 Score, and Recall. Initially using 146 features, we selected the top 14 to 25 most important ones to improve performance and simplify the model. The performance metrics were as followUsing 146 features: Accuracy = 0.75, F1 Score = 0.75, Recall = 0.74,Using 14 features: Accuracy = 0.73, F1 Score = 0.71, Recall = 0.73Despite a slight decrease in performance with 14 features, we opted for this model due to its lower complexity and enhanced interpretability. The final model includes 14 key variables from a stacked ensemble fitted on the original dataset.

Discussion

The study identified key factors affecting Quality of Life (QoL) among Asian Americans, with findings showing that aspects like health status, financial stability, and civic engagement significantly impact QoL. Ensemble techniques, particularly the Stacking Classifier, provided the highest accuracy after tuning, outperforming individual classifiers. Balancing classes through SMOTE and oversampling improved prediction for underrepresented QoL categories.

Despite these successes, limitations include a smaller sample size for lower QoL categories and a dataset limited to Austin, Texas, which may affect generalizability. Additionally, potential biases in self-reported data should be considered. Nevertheless, these results offer valuable insights for community leaders and policymakers to target support areas that could improve QoL for Asian American communities.

Conclusion

This project successfully applied machine learning to predict QoL categories and identified influential factors for Asian Americans. The Stacking Classifier, aided by data balancing techniques, proved highly effective, highlighting the value of ensemble methods for complex, imbalanced datasets. Future research could expand this work with larger, more diverse samples or explore deep learning techniques for more nuanced predictions. Overall, these findings provide actionable insights that can help guide interventions aimed at enhancing QoL for Asian Americans.

About the Data Product

Link - https://black-river-065d1510f.5.azurestaticapps.net/
Code_link: https://github.com/HiruNavodya/Quality-of-Life-Prediction-using-Machine-Learning
Data_Product: https://github.com/tharuka7/ForntEnd_QOL.git
https://github.com/tharuka7/BackEnd_QOL.git

References:

https://ourworldindata.org/happiness-and-life-satisfaction

https://pmc.ncbi.nlm.nih.gov/articles/PMC10014008/

https://www.researchgate.net/publication/358123716_Predicting_Quality_of_Life_using_Machine_Learning_case_of_World_Happiness_Index

https://www.dfki.de/fileadmin/user_upload/import/12163_8.pdf