NOVA IMS
Information Management School

BUDGETING FOR THE FUTURE

Data Mining II

Group 06

EDUARDO AMARAL | 20230355
JOANA OLIVEIRA | 20230384
JOÃO FREITAS | 20231472
RÚBEN MACHADO | 20230367
TERESA SANTOS | 20230380

TABLE OF CONTENTS

TABLE OF APPENDICES

ABSTRACT

Participatory Budgeting (PB) has emerged as a crucial strategy in municipal governance, fostering community engagement and equitable resource allocation. This report presents a data-driven approach undertaken by Mining City's council to enhance its budgeting process.

Our project aims to improve Mining's City's budgeting process by better understanding residents' demographics and lifestyles. The main hypothesis was that predictive modeling could identify critical variables and forecast community preferences, thereby informing more effective PB decisions.

Feature selection methods, including Recursive Feature Elimination, LASSO, ANOVA and Random Forest, were employed to identify the most impactful variables. We then evaluated six machine learning algorithms (Decision Tree, Logistic Regression, Random Forest, Gradient Boosting, K-Nearest Neighbors, and a Stacking Ensemble) using cross-validation and parameter tuning to optimize performance. The top-performing model, a stacking ensemble combining Random Forest, Gradient Boosting, Decision Tree and KNN with a Logistic Regression meta-learner, achieved 0.778 on validation score.

This data-driven approach confirmed our hypothesis, demonstrating that predictive modeling can significantly enhance PB processes by aligning resource allocation with community preferences. The project concluded that leveraging data analytics in municipal decision-making fosters inclusivity, transparency, and trust, ultimately improving governance and community well-being. Through these insights, Mining City aims to create a more responsive and effective PB process, ensuring that every resident's voice is heard and valued.

INTRODUCTION

In the constantly changing context of municipal governance, participatory budgeting has emerged as an essential strategy for increasing community engagement and ensuring fair distribution of resources. At the center of this approach is the ability of residents to actively determine the distribution of public funds, which improves transparency, responsibility, and trust within local government organizations. Recognizing the importance of participatory budgeting, Mining City's council goes on a data-driven journey to strengthen its decision-making processes.

In this context, the city council of Mining City seeks to improve the effectiveness of its budgeting process. To achieve this, the council has embarked on a data collection initiative to understand its residents' demographics and lifestyles better. This project, undertaken in collaboration with data scientists from NOVA IMS, aims to leverage predictive modeling techniques to identify critical variables and forecast community preferences. With this, the council aims to make informed decisions that reflect the needs of its population.

It involves several key steps: identifying relevant variables, performing multiclass classification to predict lifestyle preferences, and providing actionable insights to enhance resource allocation.

In order to gain some context regarding the topic of participatory budgeting, we did some research and found the article "How Participatory Budgeting Can Improve Governance & Well-Being" [(1)] by People Powered. It indicates that PB can lead to improved governance and community well-being. For instance, PB processes in Brazilian municipalities have been associated with lower infant mortality rates and increased spending on health and education, reflecting significant improvements in community health and well-being.

In the United States, cities like New York and Chicago have seen PB fostering civic engagement and trust in government. PB allows residents to have a direct say in how public funds are allocated, which can empower communities and enhance social cohesion. For example, New York City's PB process has led to increased investments in schools and public housing, addressing community priorities and improving living conditions.

Based on these findings, we expect our data mining project to show positive correlations between PB participation and various indicators of well-being, such as health consciousness, financial wellness, and social engagement. Our project aims to predict lifestyle outcomes using citizen-related attributes, and previous research suggests that well-designed PB processes can lead to better overall lifestyle outcomes.

[(1)] https://www.peoplepowered.org/news-content/how-participatory-budgeting-can-improve-governance-well-being - accessed on June 3rd, 2024

DATA EXPLORATION

Variables and Attributes

The dataset consists of 21 variables (Appendix 1) and 90165 rows related to the demographics and lifestyles of Mining City residents.

We will only decide what variables are relevant after collecting statistical information about the dataset, seeing the distribution of the data and performing feature selection methods.

Correlations

To give a precise analysis for the participatory budget, we have decided to make a heatmap (Appendix 2) with all the variables to check the correlation between them. This will also help us choosing which variables might need to be removed because of strong correlations (we defined a threshold of 0.8 for high correlations) and which ones may be combined into a new variable. We also noticed that 'overall_well_being' and 'financial_wellness_index' have perfect correlations, which will be dealt with in the feature selection part.

Statistics and Distribution of the Data

The initial phase involved collecting statistical information about the dataset. This was crucial for understanding the data's structure, identifying outliers, and highlighting its distribution. Methods like .info(), .describe(), count plots (Appendix 3), histograms (Appendix 4) and scatter plots (Appendix 5) were used to provide this initial overview. This foundational analysis gave us a good understanding of the data, setting the stage for more advanced mining techniques and insightful interpretations later in the project.

- The only country and city in the dataset are Data Land and Mining City, respectively. Therefore, these are redundant variables, and we removed them from the dataset.
- We created some new variables (gender and age) that can give us more insightful information than the ones we had previously (title and date of birth) and removed the latter.
- Some variables have values that are clearly wrong, like negative values or values outside the variable's range. These incoherences, as well as other incoherences we find during data preprocessing, will be fixed later.

DATA PREPROCESSING

We had to treat both the training data and the test data in order to ensure data quality and consistency, leading to an accurate predictive model.

Duplicate Values

We discovered five rows of duplicated data on the training set and decided to remove them since it might lead to overrepresentation of certain data points.

There were no duplicated rows on the test dataset.

Incoherences

Both datasets were treated to ensure coherence and reliability before further analysis.

- Unrealistic values were replaced with the mean. On the variable “age”, we decided to replace one specific record by hand (from 2023 to 2005) assuming that it was an imputation error;
- Converted negative values to their absolute value;
- Used upper clipping on variables that had a range of possible values.

Outliers

We used box plots, scatter plots and histograms to visualize the outliers. Our initial approach was to calculate the IQR threshold but quickly realized it was not a good approach since we were going to lose a very large percentage of our data if we simply removed all the values beyond that value.

- On almost all the variables, we converted the outliers to NaN and then imputed them when we imputed the rest of the missing values of that variable;
- We used IQR for the variable ‘social_media_influence_score’ since it provided us an acceptable number for the amount of outliers;
- We used an alteration of the IQR (used 2.5 instead of 1.5 on the formula) on the variable ‘entertainment_engagement_factor’ as it also provided us with an acceptable number for the amount of outliers.

Missing Values

We detected missing values across 14 variables in the training dataset. (Appendix 6)

The treatment of the missing values involved making decisions on what methods would best suit each variable to best maintain the dataset's integrity.

- For variables with a lot of missing values we used k-Nearest Neighbors;
- For variables with a low number of missing values we used the median (since the distributions are not normal, the median provides a value that is less affected by outliers than the mean);
- For the variable ‘tech_savviness_score’, we used the mean because the distribution resembles a normal distribution.

We detected missing values across the same 14 variables in the test dataset and treated them using the same instance of imputer fit to training data.

Data Skewness

In order to ensure the construction of a more accurate, interpretable, and reliable predictive model, the data was subjected to a process of skewness treatment. This was achieved through the application of log transformations to seven variables that exhibited a non-normal distribution. (Appendix 7)

Feature Engineering

In addition to the existing variables (age and gender) created at the beginning of the notebook, we introduced three new variables called 'health_index', ‘investment_profile’ and ‘age_bins_encoded’. These variables combine information from already existing variables:

- 'health_index' combines 'health_consciousness_rating', 'stress_management_score', and 'avg_weekly_exercise_hours' (Appendix 8)
- ‘investment_profile’ combines ‘investment_portfolio_value’, ‘investment_risk_apetite’ and ‘investment_risk_tolerance’ (Appendix 9)
- ‘age_bins_encoded’ only uses our already existing variable ‘age’ and creates 5 bins for it (Appendix 10)

To create 'health_index' and ‘investment_profile’, we performed data binning on each of the original variables, dividing them into five categories: Very Low, Low, Medium, High and Very High. For each original variable, a Very Low value was encoded as 0, a Low value as 1, a Medium value as 2, a High value as 3 and a Very High value was encoded as 4.

By summing the encoded values from the three variables related to each of the new ones, we created the 'health_index' and ‘investment_profile’ variables, whose values range from 5 to 15. These new variables allow us to extract and analyze more meaningful insights from our dataset by synthesizing information from the existing variables into single, comprehensive measures.

A similar process was used using Primary Component Analysis (PCA) to try and reduce the dimensionality of our dataset. We grouped similar features and performed PCA to create a new variable capturing a majority of the variance of these.

MULTICLASS CLASSIFICATION

Feature Selection

We split our dataset into 3 distinct dataframes: X_original (containing all the original features), X_feature_engineering (containing the original features, but replacing the features used in logarithmic transformations and binning with the respective outputs), and X_PCA (containing the features derived from PCA of different sets of features). We ran every model showing F1 score for each dataset to see which one performed best, and saw that X_feature_engineering was the one that worked best.

Feature selection reduces the number of features, enhancing computational efficiency, reducing overfitting, and improving interpretability. We used RFE (Recursive Feature Elimination), LASSO (Least Absolute Shrinkage and Selection Operator), ANOVA (Analysis of Variance) and Random Forest to select the most impactful features.

- Recursive Feature Elimination works by iteratively removing the least important features based on a model's performance, enhancing accuracy by reducing overfitting and improving interpretability. We selected 8 features so that we could have a reasonable number of variables to compare with the other feature selection method. (Appendix 11)
- The Lasso method works by performing variable selection and regularization, shrinking some coefficients to zero. This approach helps identify the most important features, making the model more interpretable. (Appendix 12)
- The ANOVA method is used to determine if there are any significant differences between the means of two or more independent groups and it can be leveraged to determine the significance of individual features in relation to the target variable. ANOVA assumes that the variables are normally distributed, but we were still able to use it because of the log transformations we did to the variables. In the provided code, ANOVA is employed as a feature selection technique to select the top 8 features based on their significance in predicting the target variable. (Appendix 13)
- Random Forest is an ensemble learning method that can be used to perform feature selection. It measures the importance of each feature by looking at how much the accuracy or impurity decreases when the feature is included in the model and the features that contribute more to reducing error are deemed more important. We performed Random Forest three times, one for each set of variables ('X_original', 'X_feature_engineering' and 'X_PCA').

To determine which features to use in the modeling phase, we combined the results of the 4 methods to select a subset of features that consistently shows importance across all of them. This means, select features that were labeled True on REF and ANOVA, had non-zero coefficients on the LASSO and showed up on the Random Forest bar chart. We ended up selecting 13 features and stored them in the "selected_columns" list.

Modeling

To determine the most accurate and effective model for our project, we implemented and evaluated six different machine learning algorithms. By comparing the performance of each model, we aimed to identify the best approach for our specific dataset and prediction needs. The models we considered include the Decision Tree Classifier, Logistic Regression, Random Forest Classifier, Gradient Boosting Classifier, K-Nearest Neighbors (KNN) and a Stacking ensemble.

Cross Validation

We ran all the models using cross-validation to ensure that our performance metrics are robust and to minimize the risk of overfitting. Cross-Validation allows us to assess how well our models generalize to an independent dataset, providing a more reliable estimate of their predictive performance. We used K-Fold Cross Validation, that involves partitioning the dataset into k (in our project k=5) equal-sized subsets.

Scaling

From all the scaling methods available we decided to use MinMax Scaling because it preserves the shape of the dataset. It transforms each feature into a given range, which is crucial for predictive algorithms that are sensitive to the variance in data and can be biased towards higher magnitude features. This normalization ensures that each feature has an equal opportunity to influence the predictive model, thereby enabling more balanced and interpretable results.

The scaling was done inside cross validation. This ensures that the scaling is conducted independently for each fold, preventing data leakage, ensuring realistic performance estimates and maintaining consistency.

Decision Tree Classifier

A Decision Tree Classifier is a simple model that splits the dataset into subsets based on the value of input features, creating a tree-like model of decisions. Our Decision Tree Classifier was configured using a parameter grid to find the best combination of hyperparameters. For each iteration of the random search loop, a set of hyperparameters was randomly selected from the grid. These parameters were then used to configure the model, which was subsequently trained and evaluated using K-Fold Cross-Validation. Through the observation of the graph, the best combination of parameters was 'entropy', 100, 150, 40, 'balanced' as it had the highest validation score. (Appendix 14)

Logistic Regression

Logistic Regression is a statistical technique for binary classification tasks. It models the probability of an instance belonging to a particular class by fitting a logistic function to the observed data. The model was trained using Logistic Regression, which fitted the model. To assess the model's performance and generalization ability, we employed k-folds cross-validation. The combination of parameters with the highest validation score was 'solver': saga, 'penalty': l1, 'C': 0.2. (Appendix 15)

Random Forest Classifier

A Random Forest Classifier is an ensemble learning method that builds multiple decision trees during training and merges their outputs to improve accuracy and prevent overfitting. The criterion parameter measures the quality of the split, entropy is the chosen criterion, which calculates the information gain achieved by splitting a node based on a particular feature. The 'n_estimators' parameter specifies the number of decision trees to be included in the random forest, in this case we chose 140. The 'max_depth'=20 indicates the maximum depth of each individual tree. The 'min_samples_split'=6 and 'min_samples_leaf'=6 indicates the minimum number of samples required to split a node and required to be at a leaf node, respectively. The 'class_weight'=balanced helps the model pay more attention to the minority classes. Finally, with the 'random_state' parameter we ensure that the randomness used to create the decision trees is consistent across different runs of the model. Once trained, we used the model to make predictions on a separate validation dataset, with the predicted labels stored in a variable.

(Appendix 16)

Gradient Boosting Classifier

Gradient Boosting Classifier is an ensemble learning method that builds models sequentially, with each new model attempting to correct the errors made by the previous ones. It uses gradient descent to minimize the loss function, focusing on the most difficult cases to improve the overall performance. We decided to test Gradient Boosting Classifier with 110 estimators, a maximum depth of 4 and with two different learning rates (0.4 and 0.6), that allowed us to determine the optimal one based on its performance on the validation data. By systematically evaluating the model's performance with the learning rates through cross-validation, we aimed to identify the rate that yields the best balance between bias and variance, ultimately improving the classifier's predictive capabilities.

(Appendix 17)

K-Nearest Neighbors

K-Nearest Neighbors is an instance-based classifier that makes predictions by comparing new data points to specific instances from the training dataset, leveraging the intuition that similar instances are likely to share similar outcomes.

We started by identifying the optimal number of neighbors to select through a Grid Search systematically explores all possible combinations of hyperparameters within a predefined grid. Our grid has only one hyperparameter, correspondent to the different values for neighbors. This systematic approach provides a comprehensive overview of the hyperparameter space but can be computationally expensive, testing values of k in a list with values between 5 and 33. The grid search revealed that the optimal number of neighbors was 25. We then trained the model using this optimal parameter on our training dataset and evaluated their performance using k-fold cross-validation. (Appendix 18)

Stacking Classifier

The Stacking Classifier is an ensemble learning technique that combines the predictions of multiple base models with a meta-learner. In our implementation, we defined a list of base learners, which included Random Forest Classifier, Gradient Boosting Classifier, Decision Tree Classifier and K-Nearest Neighbors. These base learners were then combined using a Stacking Classifier along with a final estimator, which was a Logistic Regression model. The base learners were trained on the training data, and their predictions were used as input features for the final estimator.

Train Score: 0.87
Validation Score: 0.78
(Appendix 19)

CONCLUSION

The project aimed to use predictive modeling techniques to enhance the effectiveness of PB and understand the preferences and lifestyles of Mining City's residents. By employing data preprocessing techniques, feature engineering and selection, and multiclass classification, we were able to use predictive modeling accurately.

Splitting the data helps to create a model that is capable of learning from the data provided during the training phase, as well as generalizing and predicting on unseen data (test set). Comparing the outcomes of several models, it was identified that the Stacking Classifier model performed the highest F1 score of 0.77944 on the test set, indicating its effectiveness in predicting residents' lifestyle types relevant to PB.

This approach can complement traditional methods of citizen engagement by allowing the council to anticipate the needs and preferences of residents, demonstrating a strong correlation with various well-being indicators.

Although we used diverse feature selection techniques to select the essential variables for the final model, there may have been other relevant variables that were not included, potentially limiting the scope and depth of the model.

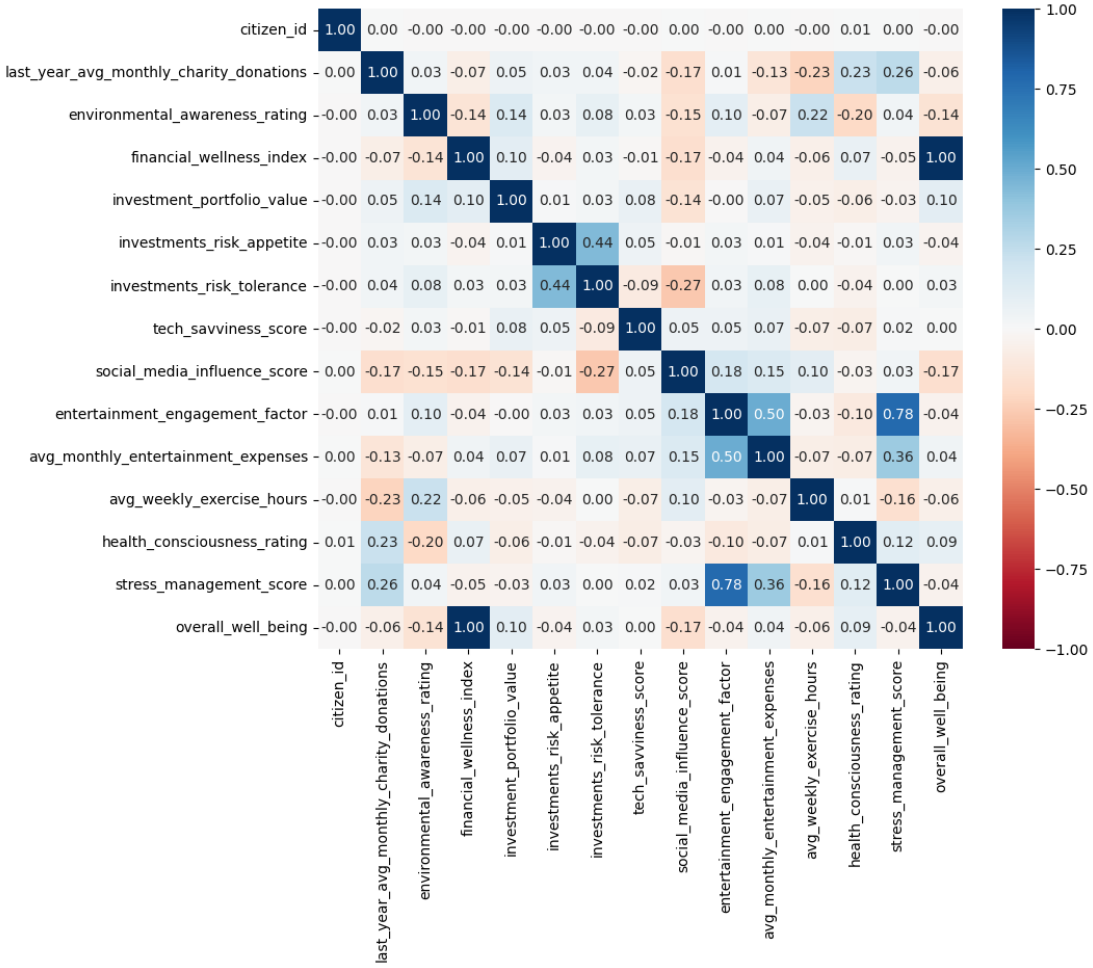
For the predicted lifestyle, we checked the number and percentage of each category to compare and understand the differences with the train dataset. Our final model predicted more people for the Adventure Seeker lifestyle (20.65%), followed by Investor (20.38%), Health-Conscious (20.08%), Fitness Enthusiast (19.53%) and the least predicted category is Travel Enthusiast (19.35%). There's a slight difference compared to the distribution of lifestyles in the train data. The most represented category is Health-Conscious (20.11%), followed by Investor (20.03%), Fitness Enthusiast (20%), Adventure Seeker (19.97%) and the least represented category is Travel Enthusiast (19.90%). As we don't have the actual lifestyles for the test data, just the predicted ones, we can't assess where our final model failed. (Appendix 20, Appendix 21 and Appendix 22)

In a nutshell, the project highlighted the importance of data mining techniques in governance, suggesting that such methodologies can lead to fairer public spending. By using data-driven insights, we ensured that Mining City can create a more inclusive participatory budgeting process, ensuring that every voice is heard.

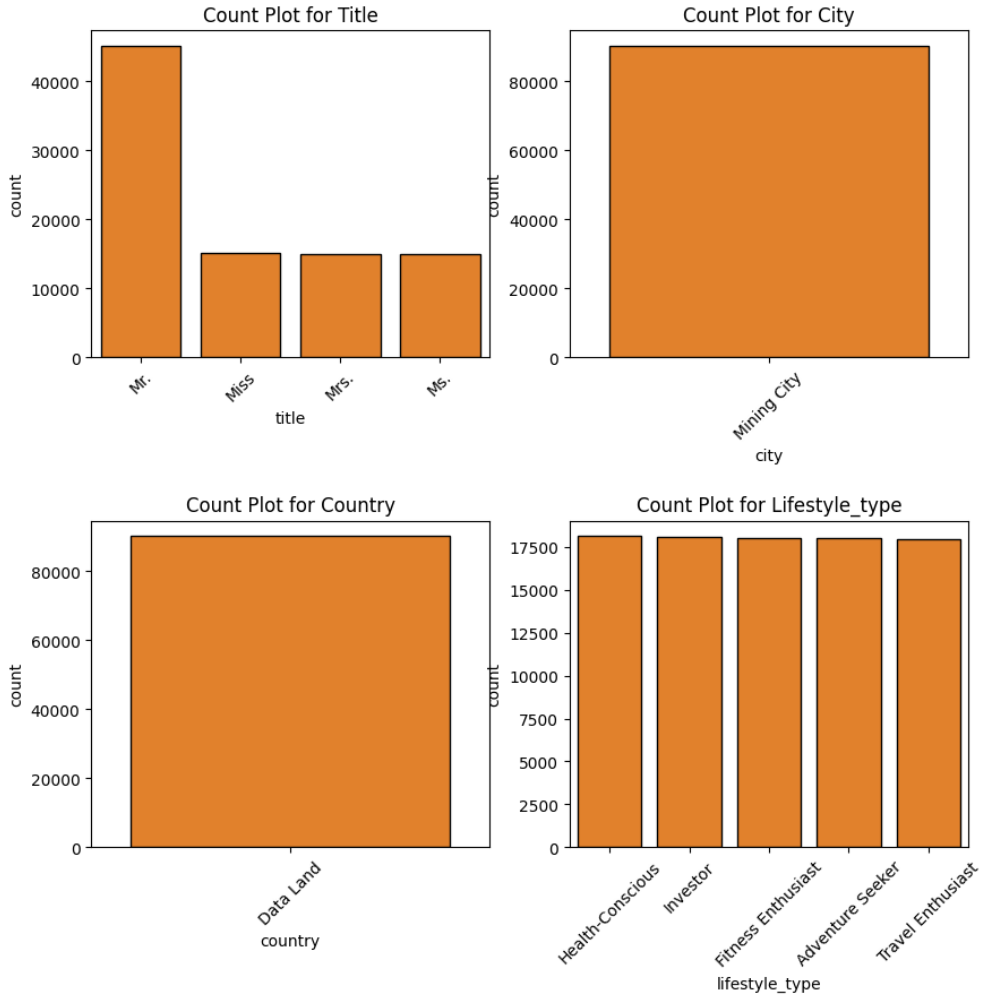
APPENDICES

Variable	Description
citizen_id	Unique identifier of the citizen.
Name	First name of each citizen.
Title	Title of each citizen.
date_of_birth	Date of birth of each citizen.
city	Name of citizen's city.
country	Name of citizen's country.
last_year_avg_monthly_charity_donations	The average of monthly charitable donations made by each citizen in the last year.
environmental_awareness_rating	A rating [0, 10] of each individual's awareness of and engagement with environmental issues.
financial_wellness_index	An index indicating each citizen's overall financial health.
investment_portfolio_value	The value, in thousands of units of currency, of each citizen's investment portfolio.
investments_risk_appetite	A measure of each individual's willingness to take risks in their investments.
investments_risk_tolerance	A measure of each individual's tolerance for risk in their investment choices.
tech_savviness_score	A score representing each citizen's proficiency and comfort with technology.
social_media_influence_score	A score representing each citizen's influence and activity on social media platforms.
entertainment_engagement_factor	A score representing each citizen's engagement with entertainment activities.
avg_monthly_entertainment_expenses	The monthly expenditure on entertainment for each citizen, in units of currency.
avg_weekly_exercise_hours	The average number of hours each citizen spends on exercise weekly.
health_consciousness_rating	A rating [0, 10] of each citizen's awareness and proactive behavior towards their health.
stress_management_score	A score indicating how effectively each citizen manages stress.
overall_well_being	A score indicating each citizen overall status.
lifestyle_type	A categorization of the predominant lifestyle choice for each citizen (Target Variable).

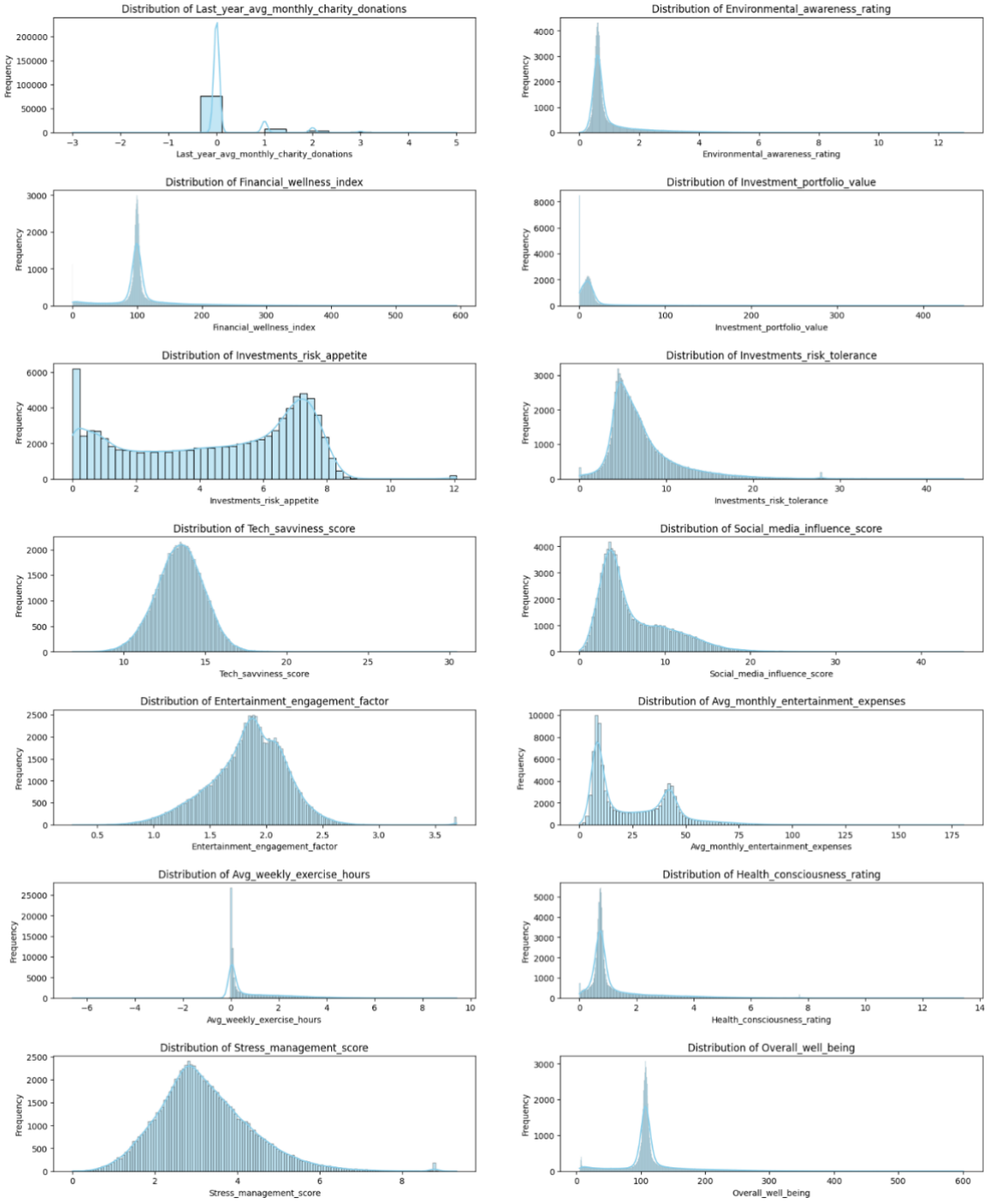
Appendix 1 – Variables Description



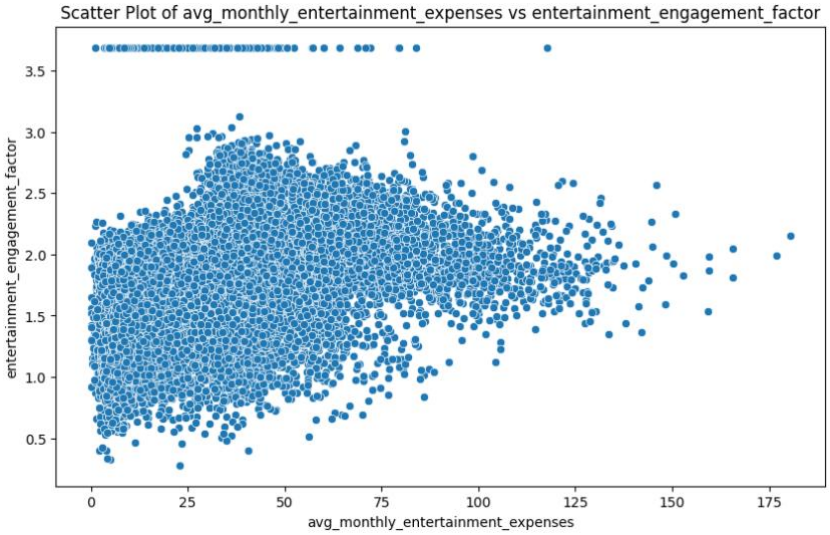
Appendix 2 – Variables Heatmap (Correlation Matrix)

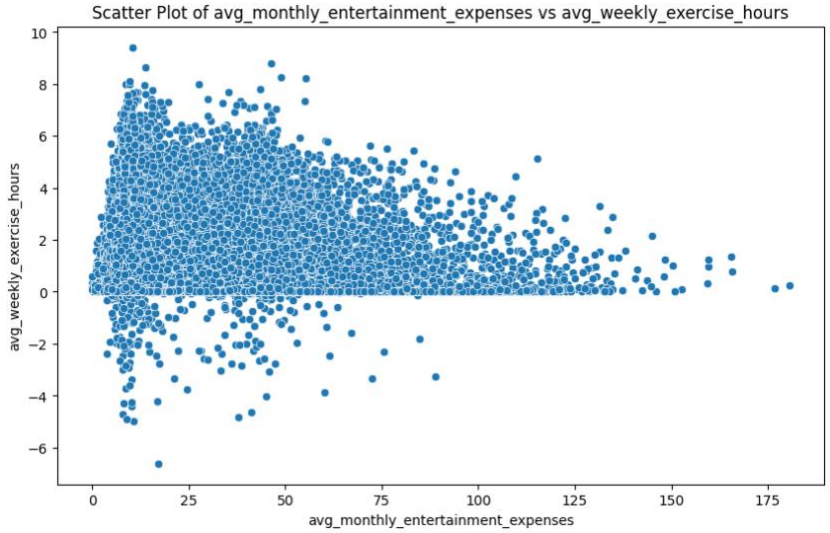


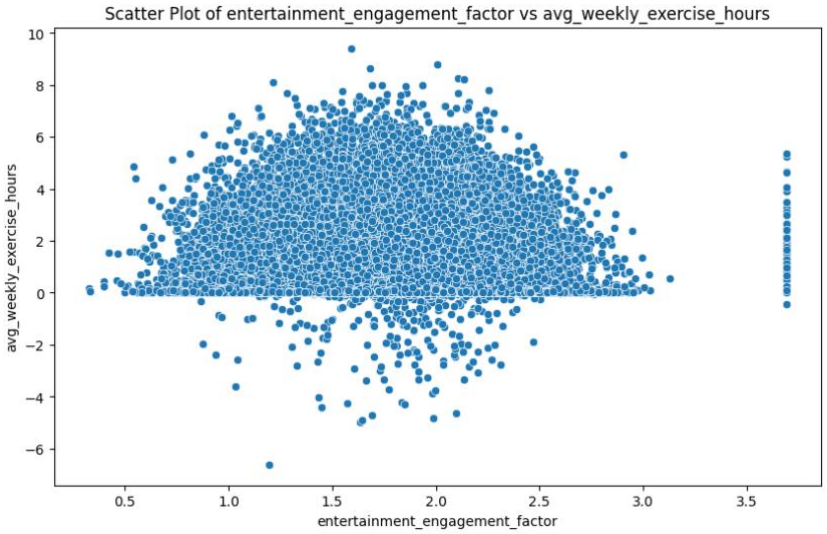
Appendix 3 – Count Plots for Categorical Variables



Appendix 4 – Histograms for Numerical Variables



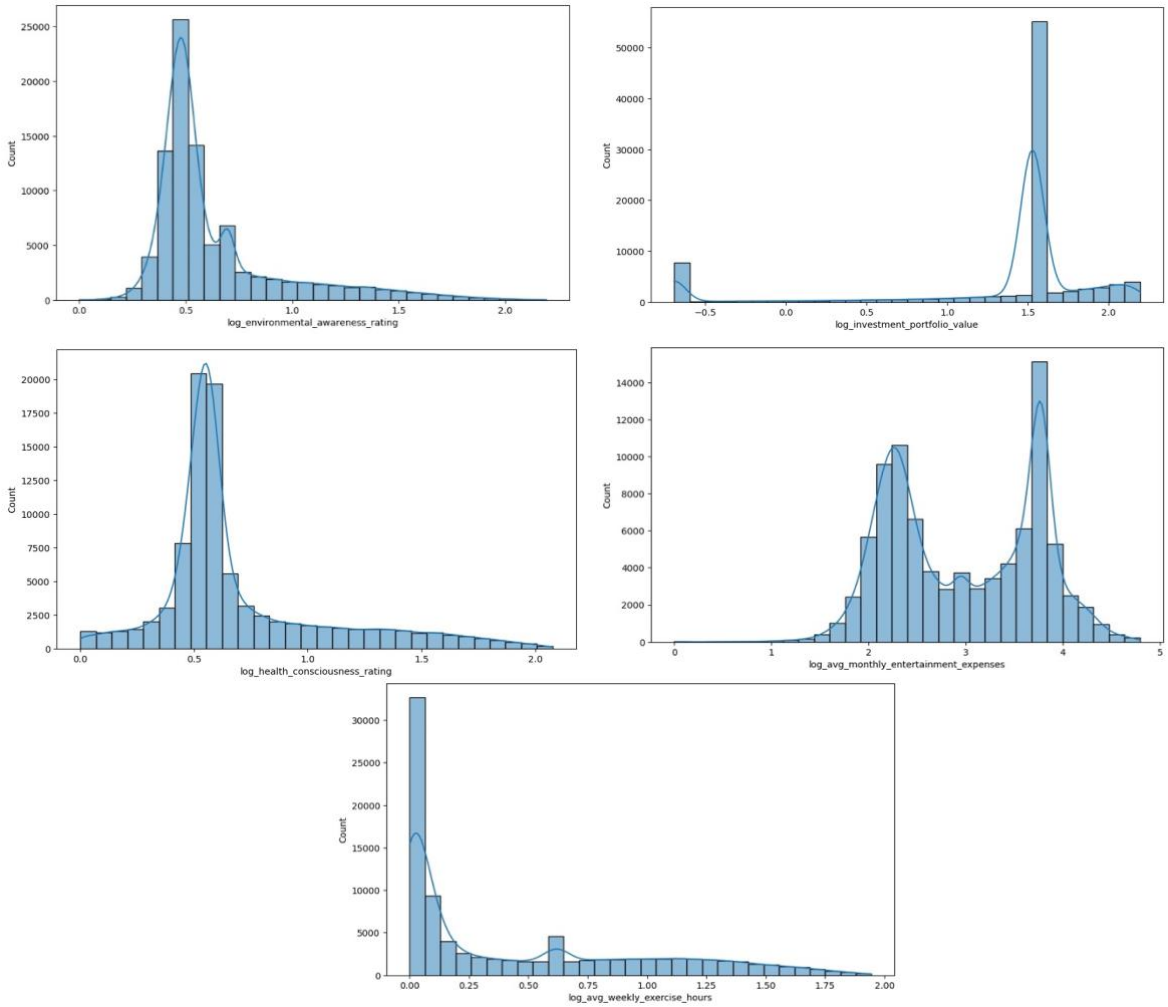




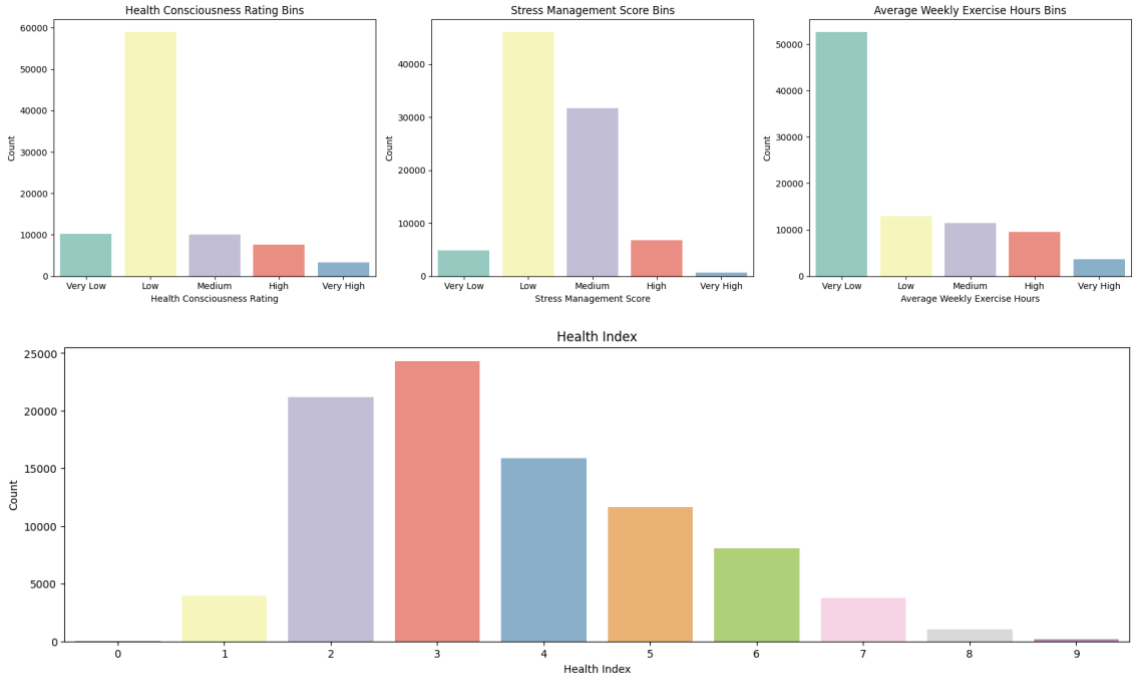
Appendix 5 – Scatter Plots for Average Monthly Entertainment Expenses, Entertainment Engagement Factor and Average Weekly Exercise Hours

	Missing Values	% of Total Values
avg_weekly_exercise_hours	1546	4.0
stress_management_score	1546	4.0
last_year_avg_monthly_charity_donations	1353	3.5
social_media_influence_score	1353	3.5
entertainment_engagement_factor	1353	3.5
investment_portfolio_value	1159	3.0
environmental_awareness_rating	966	2.5
financial_wellness_index	966	2.5
tech_savviness_score	966	2.5
health_consciousness_rating	966	2.5
investments_risk_appetite	580	1.5
investments_risk_tolerance	580	1.5
avg_monthly_entertainment_expenses	580	1.5
overall_well_being	580	1.5

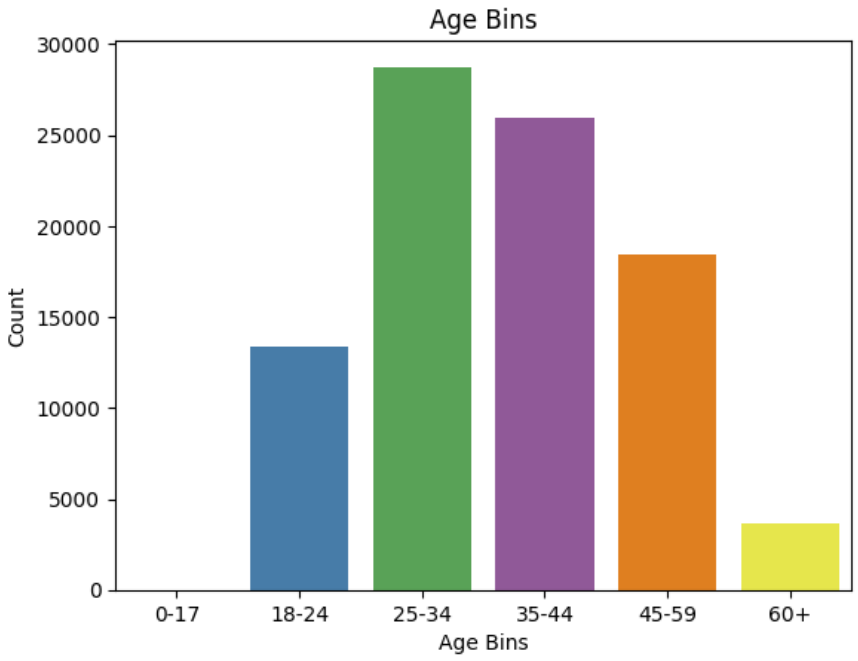
Appendix 6 – Table of Missing Values



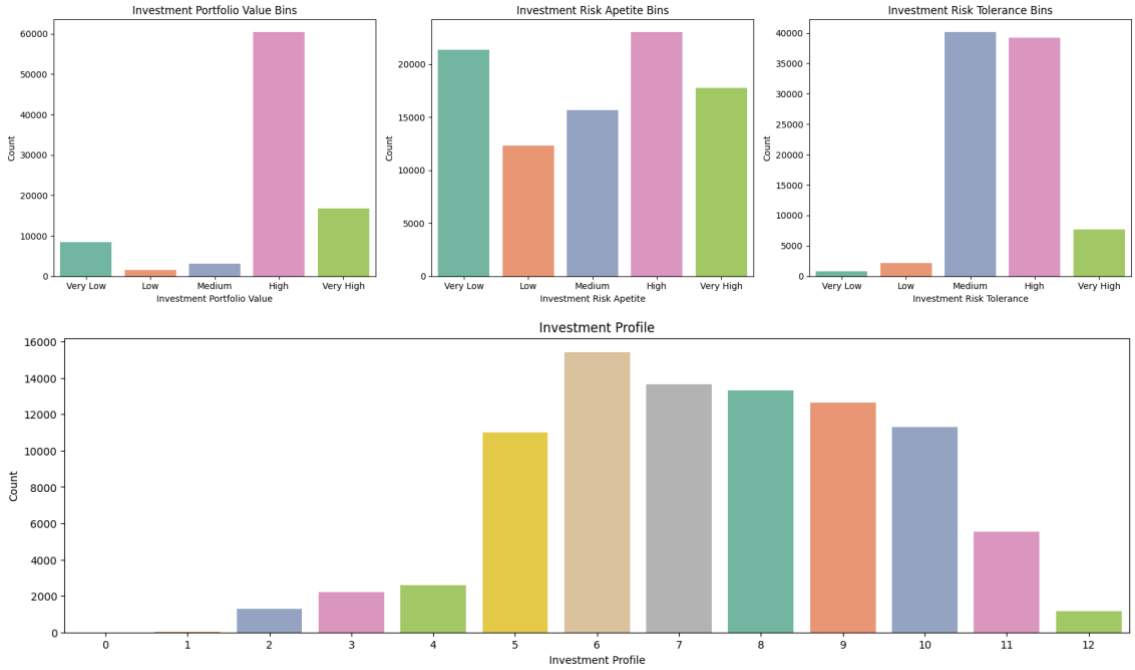
Appendix 7 – Histograms for Log Variables



Appendix 8 – Distribution of the Health Index Variable and its Components



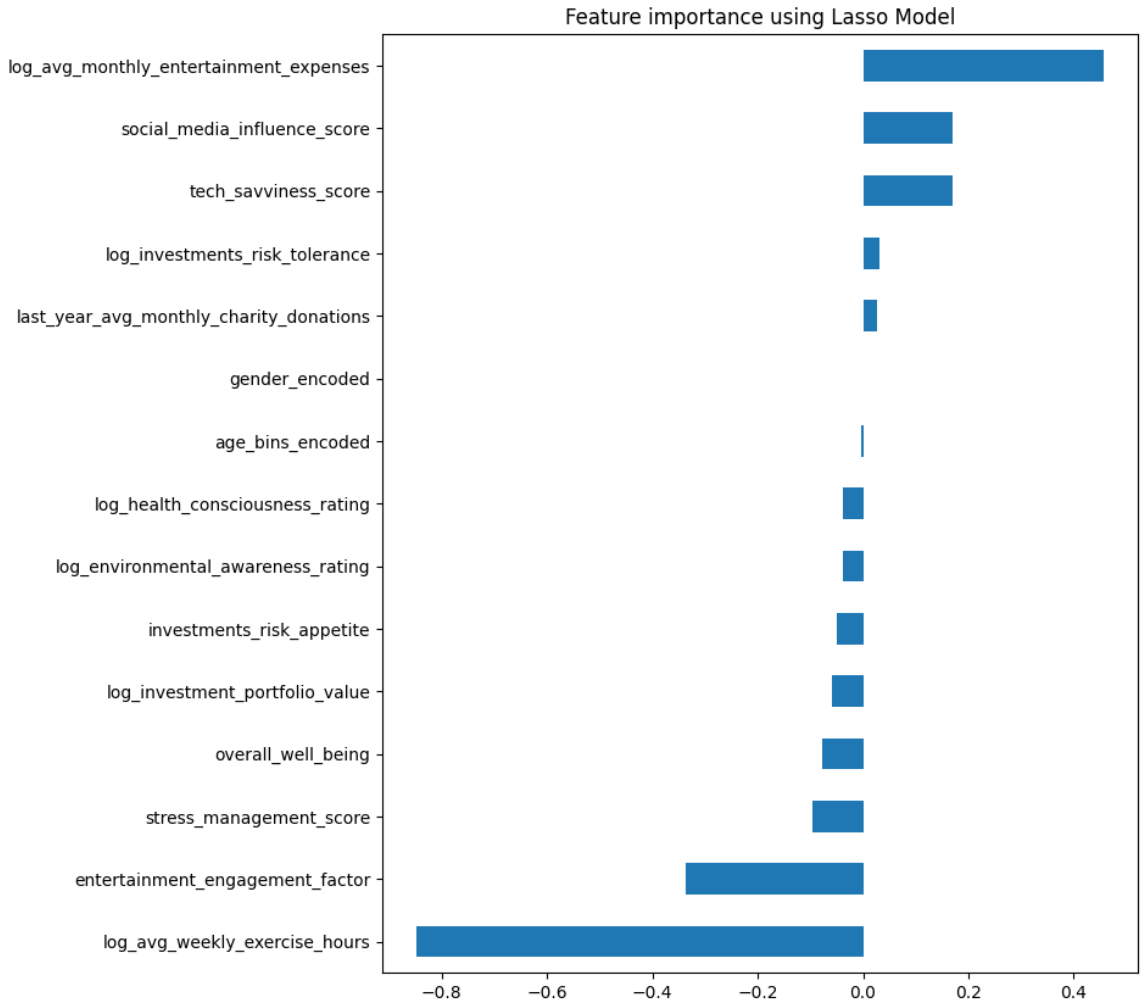
Appendix 9 – Age Bins Distribution



Appendix 10 – Distribution of the Investment Profile Variable and its Components

```
log_environmental_awareness_rating          True
log_investment_portfolio_value             False
log_investments_risk_tolerance              True
log_avg_monthly_entertainment_expenses      True
log_avg_weekly_exercise_hours               True
log_health_consciousness_rating             True
last_year_avg_monthly_charity_donations    False
investments_risk_appetite                  False
tech_savviness_score                       False
social_media_influence_score                True
entertainment_engagement_factor             True
stress_management_score                    False
overall_well_being                          True
gender_encoded                             False
age_bins_encoded                           False
dtype: bool
```

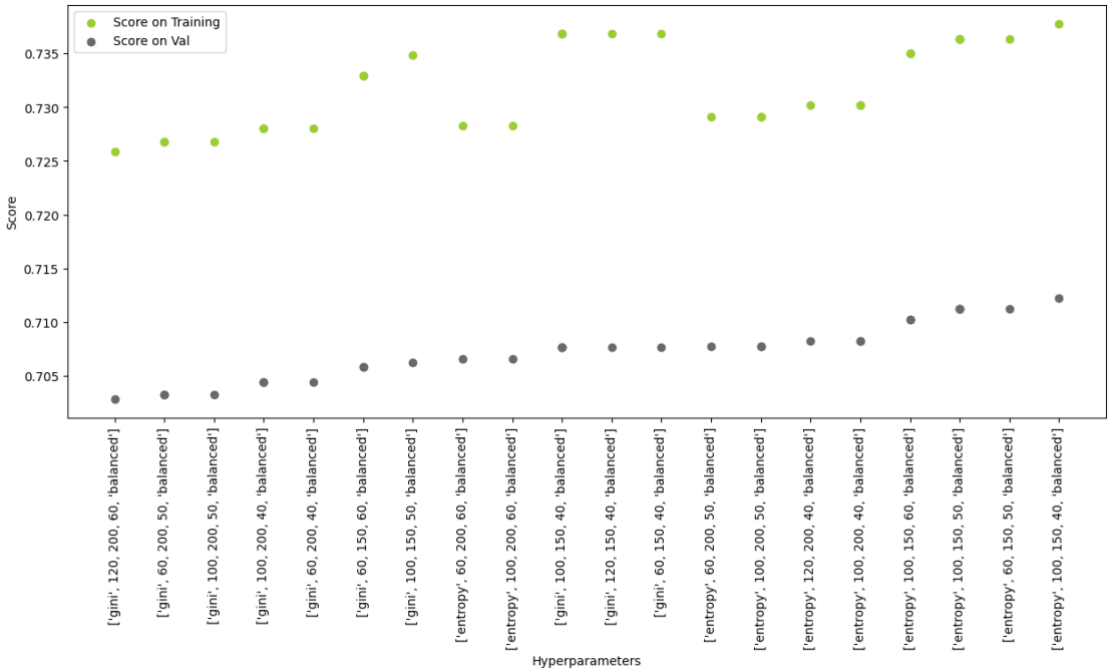
Appendix 11 – RFE Feature Selection Method Output



Appendix 12 – Lasso Regression Feature Selection Method Output

```
log_environmental_awareness_rating          True
log_investment_portfolio_value              True
log_investments_risk_tolerance             False
log_avg_monthly_entertainment_expenses      True
log_avg_weekly_exercise_hours               True
log_health_consciousness_rating             True
last_year_avg_monthly_charity_donations     True
investments_risk_appetite                  False
tech_savviness_score                       False
social_media_influence_score                True
entertainment_engagement_factor            False
stress_management_score                     True
overall_well_being                         False
gender_encoded                             False
age_bins_encoded                           False
dtype: bool
```

Appendix 13 – ANOVA Feature Selection Method Output



Appendix 14 – Decision Tree Classifier

```
Best Validation F1 Score: 0.6580620361825947
Corresponding Train F1 Score: 0.6518414475965089
Best Parameters: {'solver': 'saga', 'penalty': 'l1', 'C': 0.2}
```

Appendix 15 – Logistic Regression Score

```
Train F1 Score: 0.8768920663948346
Validation F1 Score: 0.7673673792428692
```

Appendix 16 – Random Forest Classifier Score

```
Learning Rate: 0.4, n_estimators: 110, max_depth: 4
Train F1 Score: 0.842
Validation F1 Score: 0.771
-------------------------------
Learning Rate: 0.6, n_estimators: 110, max_depth: 4
Train F1 Score: 0.859
Validation F1 Score: 0.764
-------------------------------
```

Appendix 17 – Gradient Boosting Classifier Score

```
Train F1 Score: 0.7327035003864472
Validation F1 Score: 0.7140200018897909
```

Appendix 18 – KNN Score

```
Train Score: 0.871
Validation Score: 0.778
```

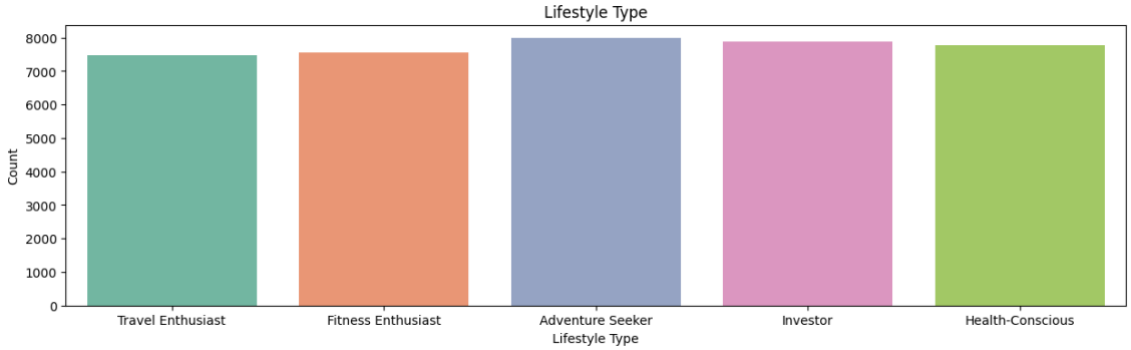
Appendix 19 – Stacking Classifier Score

lifestyle_type	Count	Percentage
Adventure Seeker	18001	19.965617
Fitness Enthusiast	18033	20.001109
Health-Conscious	18130	20.108696
Investor	18058	20.028838
Travel Enthusiast	17938	19.895741

Appendix 20 – Count and Percentage of train_data set

lifestyle_type	Count	Percentage
Adventure Seeker	7980	20.650571
Fitness Enthusiast	7548	19.532645
Health-Conscious	7763	20.089020
Investor	7875	20.378853
Travel Enthusiast	7477	19.348912

Appendix 21 – Count and Percentage of predict_test set



Appendix 22 – Bar Plot of Predicted lifestyle_type Variable