**Yarmouk University**



**IntelliShield: AI for Substance Misuse Prevention**

**College of Information Technology and Computer**

**Science, Department of Data Science and Artificial Intelligence**

**Prepared by:**

**Duaa Sawalmeh Sarah Bani Khalaf**

**Supervisor:**

**Dr. Enas Khashashneh**

**Second Semester 2025**

**Abstract:**

In today’s data-driven world, leveraging artificial intelligence (AI) to address societal issues holds great promise. This project introduces IntelliShield, an AI-powered framework designed to predict youth susceptibility to drug experimentation—specifically cannabis—based on behavioral and demographic features such as age, personality traits, and openness to experience. Using supervised machine learning techniques including XGBOOST and LightGBM, combined with hyperparameter tuning (RandomSearchCV) and class balancing strategies, the model achieved promising results in identifying at-risk individuals. By enabling early and accurate risk detection, this system empowers policymakers and health organizations to proactively intervene, supporting prevention efforts and contributing to the protection of future generations.

**Table of Contents:**

| **No.** | **Section** | **Page** |
| --- | --- | --- |
| **1** | **Project Introduction** | **4** |
| **2** | **Dataset Description** | **5** |
| **3** | **Project Objectives** | **7** |
| **4** | **Project Overview** | **8** |
| **5** | **Related Work** | **9** |
| **6** | **Methodology** | **22** |
| **7** | **Data Analysis / Preprocessing / Modeling** | **25** |
| **8** | **Web Interface / Deployment** | **101** |
| **9** | **Discussion** | **103** |
| **10** | **Conclusion** | **105** |
| **11** | **References** | **106** |
| **12** | **Appendices** | **107** |

**Project Introduction**

In a world increasingly shaped by data and intelligent systems, IntelliShield stands as a powerful example of how Artificial Intelligence can be harnessed for real-world impact. This project aims to leverage machine learning techniques to address one of society’s most pressing challenges — substance misuse.

IntelliShield was developed with a clear mission: to transform our AI knowledge into meaningful action. By analyzing behavioral and psychological patterns, the model intelligently predicts the likelihood of cannabis misuse, enabling early intervention and informed decision-making. Our approach is rooted in ethical responsibility, social awareness, and the belief that AI should serve as a shield — protecting individuals, empowering communities, and ultimately saving lives.

This project is not just a technical exercise; it’s a reflection of how we, as future AI professionals, can contribute to a safer and more conscious society. Through careful model selection, performance optimization, and transparent evaluation, IntelliShield demonstrates that when AI meets purpose, the results can be life-changing.

**Dataset Description**

**Name:** NSDUH Trimmed Dataset (National Survey on Drug Use and Health – Trimmed Version)  
**Source:** U.S. Department of Health and Human Services – [SAMHSA](https://www.samhsa.gov/data/data-we-collect/nsduh-national-survey-drug-use-and-health/datafiles)  
**Year:** ~2015–2016 (cleaned version commonly used for educational purposes)  
Shape: 56,276 records × 12 features

**Feature Descriptions:**

| Feature | Description |
| --- | --- |
| id | Unique identifier for each participant (not used in modeling). |
| age | Age category (1–17), where: 1 = 12–13 years, ..., 17 = 70+ years. |
| gender | Gender: 1 = Male, 2 = Female. |
| education | Education level (1–11):  1–7 = Less than high school,  8 = High school graduate,  9–10 = Some college/associate degree,  11 = College+ |
| neuroticism | Emotional instability score, aggregated from 12 items (each scored 1–5). |
| extraversion | Sociability/energy trait score, aggregated. |
| openness | Openness to new experiences, creativity score, aggregated. |
| agreeableness | Compassion/cooperativeness score, aggregated. |
| conscientiousness | Self-discipline/organization score, aggregated. |
| impulsiveness | Tendency to act without thinking (scale: 1–11; higher = more impulsive). |
| sensation | Sensation seeking (scale: 1–11; higher = more thrill-seeking). |
| cannabis | Target variable – cannabis (weed) use:  1 = Yes,  2 = No,  94 = Doesn’t know,  97 = Refused to answer. |

**Notes:**

* The five personality traits are based on standardized psychological instruments such as the BFI or NEO-FFI.
* Impulsiveness and Sensation Seeking are separate behavioral traits, measured on a different 1–11 scale.
* Target values (94, 97) represent ambiguous responses and should be removed or handled before training classification models.

**Data Access:**  
You can access the dataset from the official [SAMHSA NSDUH data files page](https://www.samhsa.gov/data/data-we-collect/nsduh-national-survey-drug-use-and-health/datafiles).

**Project Objectives:**

* **Empower decision-makers** — Equip parents, educators, and policymakers with insights into behavioral and psychological risk factors associated with youth substance misuse to enable timely and informed intervention.
* **Drive awareness through data** — Offer a robust, data-driven foundation for developing more targeted and impactful awareness campaigns tailored to at-risk demographics.
* **Support early intervention** — Utilize predictive modeling to identify high-risk individuals early, enabling proactive support and reducing the likelihood of substance dependency.
* **Develop scalable AI tools** — Build an adaptable and scalable machine learning model that can be integrated into broader health and education systems to support long-term preventive strategies.

**Project Overview**

In an era where the challenges faced by youth are increasingly complex and multifaceted, the ability to predict and prevent substance misuse has never been more critical. IntelliShield harnesses the power of artificial intelligence to build an intelligent, data-driven solution aimed at identifying young individuals at risk of cannabis experimentation.

This project integrates advanced supervised machine learning algorithms, including XGBOOST and LightGBM, enriched with meticulous hyperparameter tuning and data balancing techniques to ensure robust and reliable predictions. By focusing on key behavioral and demographic features — such as age, personality traits, and openness to experience — IntelliShield provides nuanced risk assessments tailored to individual profiles.

Deployed as an interactive and scalable platform, IntelliShield not only delivers instant risk evaluation but also serves as a vital decision-support tool for parents, educators, and policymakers. Its proactive nature enables timely interventions that can curb the progression toward substance dependence, ultimately fostering healthier communities.

Through this innovative blend of AI technology and social responsibility, IntelliShield exemplifies how intelligent systems can be applied beyond theory to create tangible, positive impact—transforming data into protection, prevention, and hope for future generations.

**Related work:**

1. **Machine Learning-Based Prediction of Substance Use in Adolescents in Three Independent Worldwide Cohorts: Algorithm Development and Validation Study.**

This research aims to develop a machine learning (ML) model that can predict adolescent substance use across three national datasets from South Korea, the United States, and Norway. The models were trained using the Korea Youth Risk Behavior Web-Based Survey (KYRBS) (number of participants: 1,098,641) and tested using the Youth Risk Behavior Survey (YRBS) from the United States (2,511,916 participants) and the Ungdata from Norway (700,660 participants).

**Techniques Used:**

Several machine learning models were tested, with XGBoost being the best-performing.

Importance analysis techniques such as SHapley Additive exPlanation (SHAP) were used to identify the most important predictive factors.

**Problems Faced and Solutions:**

Main Challenge: The cultural and social environments differ between the three countries, making generalization of the model difficult.

Solution: Using three different national datasets and testing the model on them to ensure generalizability.

Model accuracy varies across countries, with stronger performance in the US and Korean data compared to the Norwegian data.

Solution: Analyze the most influential factors in each dataset separately to improve interpretation and adapt to the local context. Model **Accuracy and Results:**

On the KYRBS (South Korea) dataset, XGBoost achieved:

AUROC: 80.61%

Precision: 30.42%

Sensitivity: 31.30%

Specificity: 99.16%

F1-score: 30.85%

On YRBS (United States):

AUROC: 79.30%

Precision: 68.37%

On Ungdata (Norway):

AUROC: 76.39%

Precision: 12.74%

The most influential factors in predicting drug use:

Smoking status (most influential factor)

Body mass index (BMI)

Suicidal ideation

Alcohol consumption

Sadness and hopelessness

**Conclusion:**

The research demonstrates that machine learning models, especially XGBoost, have the ability to predict Adolescent substance use has been accurately identified across different cultures. These models could be used in the future to develop more targeted preventive intervention programs, focusing on the factors most influential in addictive behavior.

1. **Machine-learning approaches to substance-abuse research: emerging trends and their implications**

This paper reviews the latest trends in the use of machine learning techniques to study substance use disorders, focusing on emerging applications and their implications for future medical research and practice.

**Techniques Used:**

Machine learning (ML) has been applied to substance use disorder data in several contexts, such as:

Predicting current use

Assessing risk for future use

Predicting treatment success

A wide range of data has been used, including:

Physiological measurements

Longitudinal questionnaires

Treatment records and medical outcomes

National surveys

Social media

**Problems Researchers Faced and Solutions:**

Some models were inaccurate due to limited data or weak validation methods.

Solution: The need for larger and more diverse datasets, as well as adopting standardized standards for data recording and analysis.

Uneven performance of models, with some models demonstrating high predictive accuracy while others performed poorly.

Solution: Relying on deep neural network models and applying them to multimodal unstructured data to increase accuracy.

**Research Results and Model Accuracy:**

Some models demonstrated high accuracy in predicting drug use when using physiological and behavioral data.

Machine learning applications have shown promising potential in the clinical diagnosis of substance use disorders.

Further rigorous testing is needed to ensure the reliability of the results.

**Conclusion:**

The research confirms that the use of machine learning in the study of substance use disorders has great potential, but it still faces challenges related to data quality and validation standards. Improved data recording, increased data volume, and the adoption of advanced deep learning models are expected to improve the accuracy of predictions and expand their clinical applications.

1. **Substance use prediction using artifcial intelligence techniques**

This study applies advanced deep learning (DL) models, particularly Long Short-Term Memory (LSTM) networks, to predict substance use trends. It explores how different types of LSTM models—standard LSTM, Bi-directional LSTM (BiLSTM), and Recursive LSTM—are integrated with SHAP (Shapley Additive Explanations) analysis to identify and explain the factors influencing substance use. The goal is to enhance public health policies and intervention strategies by providing more accurate and actionable predictions.

**Techniques Used:**

1. **LSTM Models**: These deep learning models are ideal for handling sequential data and understanding temporal dependencies, making them suitable for tracking substance use trends. The study used:
   * **Standard LSTM**: Basic form of LSTM for sequence prediction.
   * **BiLSTM**: This model considers both forward and backward sequences, which enhances the prediction capacity by capturing a broader context of the data.
   * **Recursive LSTM**: A variation of LSTM that allows for deeper learning of sequential patterns in data.
2. **SHAP Analysis**: This technique is used to provide a clear understanding of the contribution of individual features to the predictions, making the model’s decisions more interpretable.
3. **SMOTE (Synthetic Minority Over-sampling Technique)**: To address class imbalance in the dataset, especially with rare substance use cases (like ecstasy), SMOTE was applied to artificially increase the representation of minority classes in the data.

**Challenges Faced:**

1. **Class Imbalance**: One of the major challenges in the study was the extreme imbalance in the dataset, particularly for rare substances like ecstasy. Traditional ML models struggle to make accurate predictions when the target variable has very few instances.
   * **Solution**: SMOTE was applied to resample the minority classes, helping to balance the dataset and improve model performance.
2. **Data Complexity**: While the LSTM models perform well with complex sequential data, the dataset used in this study (survey-based with simpler features) didn’t fully leverage the potential of LSTM’s capabilities.
   * **Solution**: Despite this, the study showed that LSTM models can still be useful in simpler data contexts, although traditional ML models performed similarly.
3. **Interpretability**: With deep learning models, understanding the exact reasons behind predictions can be difficult.
   * **Solution**: SHAP analysis was integrated to provide insight into the significance of each feature, enhancing the interpretability of the model’s decisions.
4. **Overfitting**: The use of resampling methods like SMOTE introduces the potential risk of overfitting, especially when the synthetic data might not fully represent the minority class in real-world scenarios.
   * **Solution**: The study acknowledges this challenge and highlights the need for careful application of SMOTE to avoid overfitting.

**Accuracy and Results:**

1. **Prediction Accuracy**: Both the LSTM models and traditional ML models achieved high prediction accuracies, approximately 99%. However, in certain cases, traditional ML models slightly outperformed LSTM models due to the simpler nature of the data.
   * The high accuracy indicates that the feature selection process and data preprocessing methods were effective.
2. **Performance of LSTM Models**: BiLSTM and standard LSTM models performed well, especially in detecting complex patterns in substance use behaviors. The BiLSTM model, by incorporating both past and future context, was particularly effective.
3. **Impact of SMOTE**: SMOTE improved the model's ability to predict minority cases like ecstasy, but it also came with limitations, particularly concerning the risk of overfitting the minority class and potentially oversimplifying data complexity.

**Contributions and Future Directions:**

* **Advancement in Drug Research and Public Health Policy**: The study introduces a novel way of predicting substance use trends using LSTM models, providing actionable insights for public health policies aimed at prevention and early intervention.
* **Future Research**: The study encourages further research into using LSTM models with more complex and non-linear datasets. Combining feature selection methods (like Recursive Feature Elimination and Random Forests) with LSTM could lead to more accurate predictions and better feature refinement.
* **Integration of ML and DL**: This research contributes to the integration of machine learning (ML) and deep learning (DL) in social science and public health, paving the way for more effective and data-driven strategies in addressing substance abuse.

**Conclusion:**

In conclusion, while LSTM models didn’t significantly outperform traditional machine learning models in this study, they represent a promising tool for substance use prediction, particularly in more complex datasets. The study also emphasizes the role of advanced computational techniques in overcoming challenges like data imbalance and improving the interpretability of prediction models. The use of SHAP analysis and SMOTE, along with LSTM, provides a comprehensive approach for understanding and predicting substance use behaviors, with significant potential for future applications in public health policy and intervention planning.

1. **Prediction of addiction to drugs and alcohol using machine learning: A case study on Bangladeshi population**

Aims to predict the risk of drug and alcohol addiction using machine learning techniques. Researchers collected data from 510 people, revealing that 209 people suffered from addiction due to the influence of friends, while 98 people cited curiosity as a reason for addiction. After collecting the data, they used several machine learning algorithms to analyze the data and predict the risk of addiction.

**Techniques Used**

**Machine Learning Algorithms:**

The researchers used nine algorithms to train models on the data:

k-NN (k-Nearest Neighbors)

SVM (Support Vector Machine)

Logistic Regression

Naïve Bayes

Random Forest

CART (Classification and Regression Trees)

AdaBoost

MLP (Multi-Layer Perceptron)

GBM (Gradient Boosting Machine)

**PCA (Principal Component Analysis) Analysis Technique:**

PCA was used to reduce the dimensionality of the data and facilitate the machine learning process by reducing the number of input features, which helps improve the performance of some models.

**Problems Faced by the Researchers and How They Solved Them**

**Feature Selection:**

The researchers faced a challenge in selecting appropriate features that could significantly impact addiction prediction. Data was collected from addicted and non-addicted individuals based on social, psychological, and familial factors. 23 data attributes were identified, including age, marital status, level of self-care, sleep disturbances, family problems, and others.

Solution: These attributes were selected after consulting with physicians and researchers in the field, and based on relevant scientific articles and reviews.

Challenges in Improving Performance Using PCA:

**Some models did not achieve significant performance improvements after applying PCA, while others achieved significant improvements.** For example, k-NN saw its accuracy drop from 96.8% to 82.29% after using PCA.

Solution: The impact of PCA on each individual model was reviewed, and the performance differences were analyzed to determine which models benefited most from this technique.

**Dealing with Unprocessed Data:**

Before applying PCA or any other data processing, the accuracy of some models was low (such as SVM, which achieved 59.01%).

Solution: The data was appropriately preprocessed, and PCA was applied to improve performance, which showed significant improvements in some models.

**Model Accuracy and Results Achieved**

**Model Accuracy Before and After PCA:**

Before applying PCA:

k-NN achieved the best accuracy of 96.8%.

SVM achieved 93.75% accuracy.

Logistic Regression achieved 84.37% accuracy.

CART achieved 50% accuracy.

After applying PCA:

Logistic Regression achieved the best accuracy of 97.91%.

SVM achieved 95.83% accuracy.

k-NN achieved 82.29% accuracy.

CART saw its performance deteriorate to 59.37% accuracy.

**Model Performance Based on Other Criteria:**

Logistic Regression showed the best performance in terms of accuracy (97.91%), but in terms of overall performance, CART was considered the best in some criteria, such as sensitivity (100%), but with a decline in the accuracy of other models.

SVM also showed good results, with an accuracy of 95.83% and good performance in sensitivity and specificity. Comparative Analysis with Previous Work

The results of this research were compared with previous studies in similar fields, such as predicting smoking behavior, diagnosing tobacco-related diseases, and predicting alcohol disorders.

The accuracy of the model in this research (97.91%) was high compared to other studies, such as:

Zhang et al. achieved 84.11% using XGBoost.

Kumari et al. achieved 98.7% in predicting alcohol addiction using artificial neural networks.

Compared to other studies, this study demonstrated high accuracy using simple techniques, enhancing the effectiveness of the methods used.

1. **Analysis of substance use and its outcomes by machine learning**

The aim of this research was to study the development of substance use disorder (SUD) among children and young people and how to predict the future development of this problem using a set of psychological and health characteristics. Artificial intelligence and machine learning (ML) techniques were used to extract patterns and identify factors predictive of the development of substance use.

**1. Research Results:**

Detection of current SUD and prediction of future SUD using the SUS score:

Detection: SUD was accurately detected (>80% AUROC) across all visits.

Future Prediction: Predictive ability was intermediate between ages 12 and 14 (AUROC = 0.61) and gradually improved with age, reaching (AUROC = 0.80) at age 19.

Substance Use Severity Models:

Models with two, three, and four paths were tested. The single-path model performed best, although it showed poorer BIC fit. The single-path model had 27.3% of participants in the high-intensity trajectory and 72.7% in the low-intensity trajectory.

Characteristics Associated with High-Intensity Trajectory:

Approximately 30 psychological and health characteristics were identified as optimal predictors of high-intensity trajectory.

The most prominent characteristics included aggressive behavior, pain (e.g., stomach pain), poor physical fitness, and irregular sleep habits.

Comparison of Machine Learning Models:

Random Forest and Naïve Bayes models demonstrated superiority over other models, demonstrating high accuracy and increasing predictive power with age.

**2. Techniques Used:**

Machine Learning:

Machine learning techniques such as Random Forest (RF) and Naïve Bayes were used to predict whether participants were in the high- or low-intensity trajectories of drug use.

Automated Feature Selection (AFS) using the RF algorithm was used to select 30 psychological and health characteristics from a large set.

Trajectory Analysis:

Trajectory analysis was applied to divide participants into different trajectories based on the severity of drug use.

**3. Problems faced by researchers and how they were resolved:**

The problem of trajectory balance:

There was a discrepancy in group sizes in the three- and four-trajectory models. This problem was resolved by choosing the two-trajectory model because it had a more balanced group size and was more suitable for prediction.

Flaws in initial selection:

When data were collected at an early age (10-12 years), no differences were found between groups due to the initiation of drug use. However, the actual occurrence of abnormal behaviors such as aggression and truancy was predicted at older ages (between 12-14 and 16 years).

Challenge in selecting features:

There is a challenge in selecting the optimal features to predict the future development of SUD. The RF algorithm was used to select the top 30 features.

Challenge in model accuracy:

It was difficult to accurately predict the future using a complex and divergent dataset. However, the researchers were able to achieve high model accuracy across different age groups, with prediction accuracy reaching 93% at age 22.

**4. Model Accuracy and Results:**

Accuracy:

The accuracy of the models used to predict trajectory was initially good (AUC = 0.71) and increased with age (AUC = 0.93 at age 22). The RF and Naïve Bayes algorithms demonstrated high predictive power with low variance in accuracy.

SUD Prediction:

The AUROC (area under the operating curve) index, which measures the accuracy of the models compared to SUD, achieved similar results (0.86 for intensity of use and 0.85 for frequency of use and number of substances used).

1. **Machine-learning approaches to substance-abuse research: emerging trends and their implications**

This review aims to provide an overview of some recent trends in the application of machine learning (ML) techniques to substance use disorder (SUD) data, exploring future challenges and opportunities in research and practice.

**Recent Findings:**

ML techniques have been applied to SUD data for a wide range of predictive applications, such as:

**Detecting current substance use.**

Assessing future risk.

Predicting treatment success.

Models used in this field include diverse machine learning techniques and multiple datasets, such as:

Physiological measurements.

Longitudinal surveys.

Treatment outcomes.

National surveys.

Medical records.

Social media.

**Abstract:**

The continued application of ML techniques to SUD data shows great promise in improving the accuracy of predictions. Some models demonstrate high accuracy, particularly in predicting current substance use using physiological and behavioral measurements, suggesting their potential for future clinical diagnostic applications. However, results are not always consistent, with some models performing poorly or even being subject to chance. This is often due to a lack of data and/or weak validation methods.

To advance this field, research on larger, multi-modal data, along with standardization in data recording methods and accurate testing protocols, is expected to be the ideal path. The use of deep neural network models in multi-modal unstructured data applications will play a pivotal role in advancing this field.

**Future Technologies and Trends:**

Multi-modal data (a combination of physiological and behavioral measurements, medical records, and others) can significantly improve predictions.

Deep neural networks are promising tools that can be used to analyze unstructured data that is difficult to process using traditional techniques.

Future Challenges:

Increasing data volume and the use of multi-modal data.

Improving model validation methods to avoid problems of poor performance.

Standardizing recording methods to obtain results that are generalizable across multiple studies.

**Methodology**

This study employed a structured machine learning pipeline to predict cannabis use using the NSDUH trimmed dataset. The dataset, containing 56,276 entries and 12 features, includes demographic information, Big Five personality traits, impulsiveness, sensation-seeking behavior, and self-reported cannabis use. Prior to modeling, records with ambiguous target labels (i.e., 94 = "Don’t know" and 97 = "Refused") were removed, yielding a final dataset of 56,245 samples.

The methodology followed four main stages: data preprocessing, feature selection, modeling, and evaluation.

**1. Data Preprocessing**

Although the dataset did not contain evident outliers, Winsorization was applied to numeric features as a precautionary measure to ensure robustness in case of unexpected user input during future deployment. MinMax scaling was subsequently used to normalize the data. Given the class imbalance (32,031 non-users vs. 24,214 users), the SMOTE technique (Synthetic Minority Oversampling Technique) was employed to oversample the minority class in the training set.

**2. Feature Selection**

Feature importance was first evaluated using H2O’s Gradient Boosting classifier. However, removing the least important features based on this method led to a decline in model performance. A second analysis using XGBoost yielded similar importance rankings. Based on these insights, the top five features—age, conscientiousness, extraversion, neuroticism, and openness—were selected for further experimentation. This subset maintained high predictive performance while reducing model complexity.

**3. Model Training and Tuning**

A range of classification models was trained using both the full feature set and the selected top five features:

Logistic Regression

Decision Tree

Random Forest

Gradient Boosting (tuned with GridSearchCV)

XGBoost

LightGBM

CatBoost

K-Nearest Neighbors

Naive Bayes

Support Vector Machine (SVM)

Multi-layer Perceptron (MLP)

Each model was evaluated using the F1 score due to the class imbalance. Stratified splitting was used to divide the data into 80% training and 20% testing sets while maintaining class proportions.

**4. Evaluation and Comparative Analysis**

Among all models, XGBoost achieved the best overall performance, especially when using all available features. It consistently outperformed other classifiers in terms of F1 score on both training and testing sets, indicating strong generalization without significant overfitting.

**Data Analysis /Preprocessing/ Modeling**

**Exploratory Data Analysis (EDA)**

**1. Data Cleaning and Initial Inspection**

* The ID feature was dropped since it serves no predictive purpose.
* The dataset contains 56,276 samples and 11 numerical features (all integer type, int64).
* No missing values or duplicates were found, which indicates the dataset is clean and ready for analysis without imputation or deduplication steps.
* The target variable cannabis originally contained values:
  + 1 = Yes (uses cannabis)
  + 2 = No (does not use cannabis)
  + 94 = Don’t know
  + 97 = Refused  
    Rows with values 94 and 97 were removed to maintain binary classification integrity. The target was then mapped to {1: Yes, 0: No}.

**2. Dataset Composition**

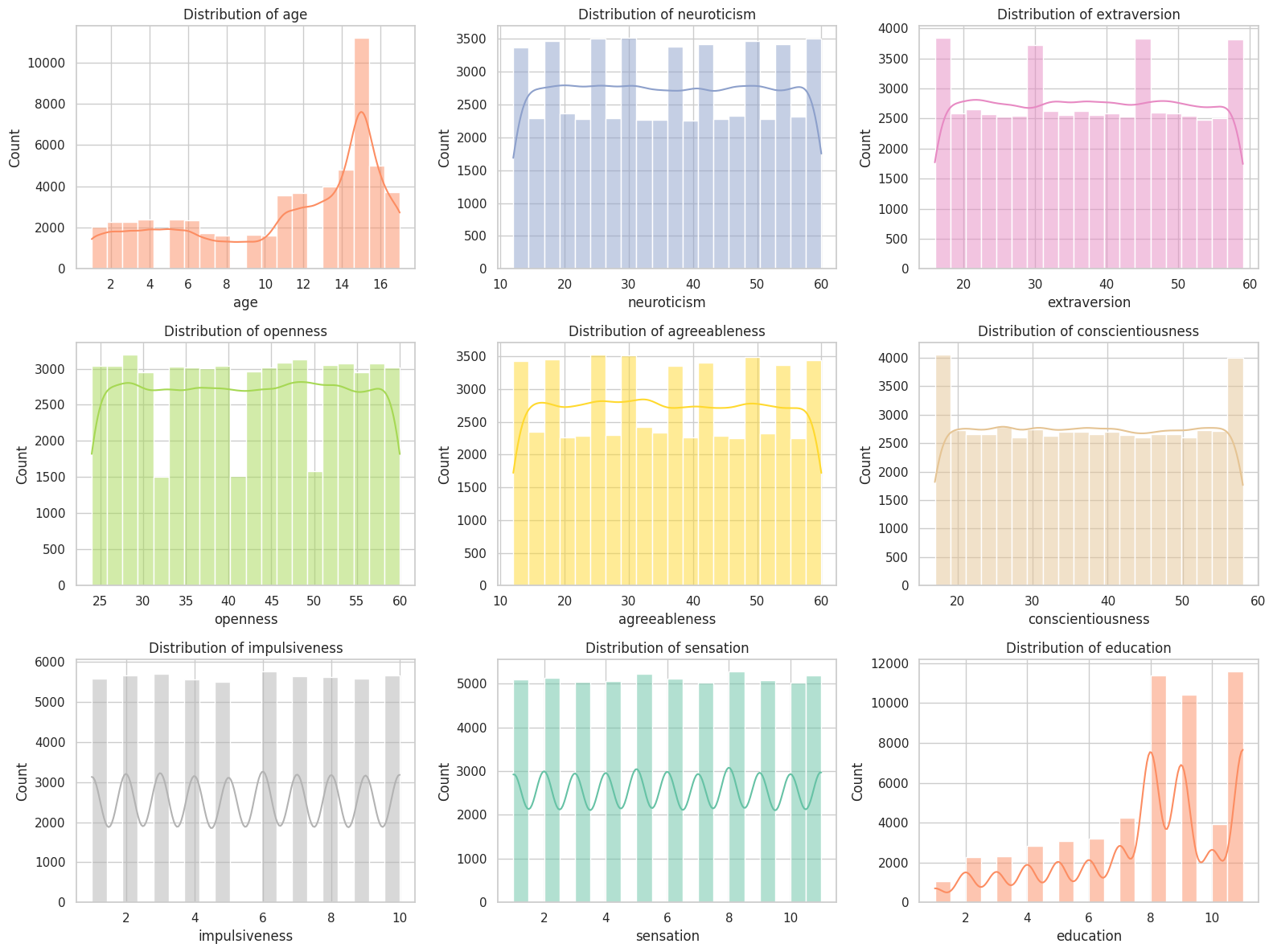
* Features include demographics (age, gender, education), personality traits (neuroticism, extraversion, openness, agreeableness, conscientiousness), and behavioral traits (impulsiveness, sensation seeking).
* Age groups are encoded categorically from 1 to 17 representing ranges (e.g., 1 = 12-13 years, 17 = 70+ years).
* Gender encoded as 1 = Male, 2 = Female.
* Education ranges from 1 (lowest) to 11 (college graduate or higher).

**3. Statistical Summary**

* Age: Mean value around 11, indicating most participants are in the 40-44 age bracket.
* Gender: Nearly balanced with 48% males and 52% females.
* Education: Mean level is about 8, indicating majority completed high school, but distribution is slightly skewed toward lower education.
* Personality traits: Scores are aggregated from Likert scale questionnaires and range moderately (e.g., neuroticism mean ~36 on a scale of 12-60).
* Behavioral traits: Impulsiveness and sensation seeking scores indicate moderate to above-average tendencies.

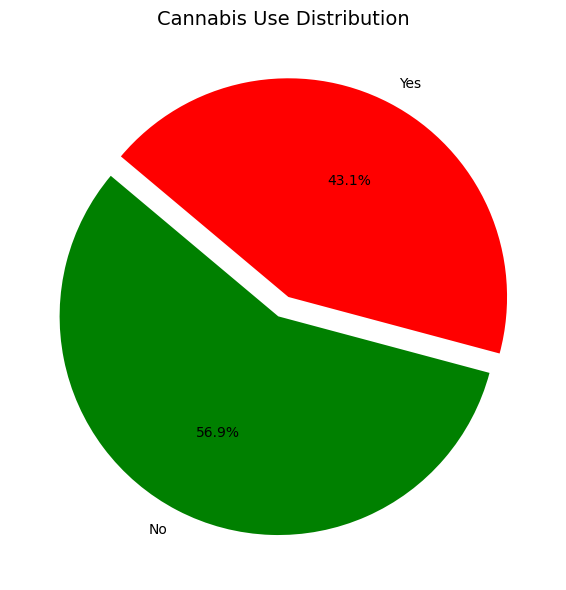
**4. Data Distributions**

* Histograms and KDE plots reveal:



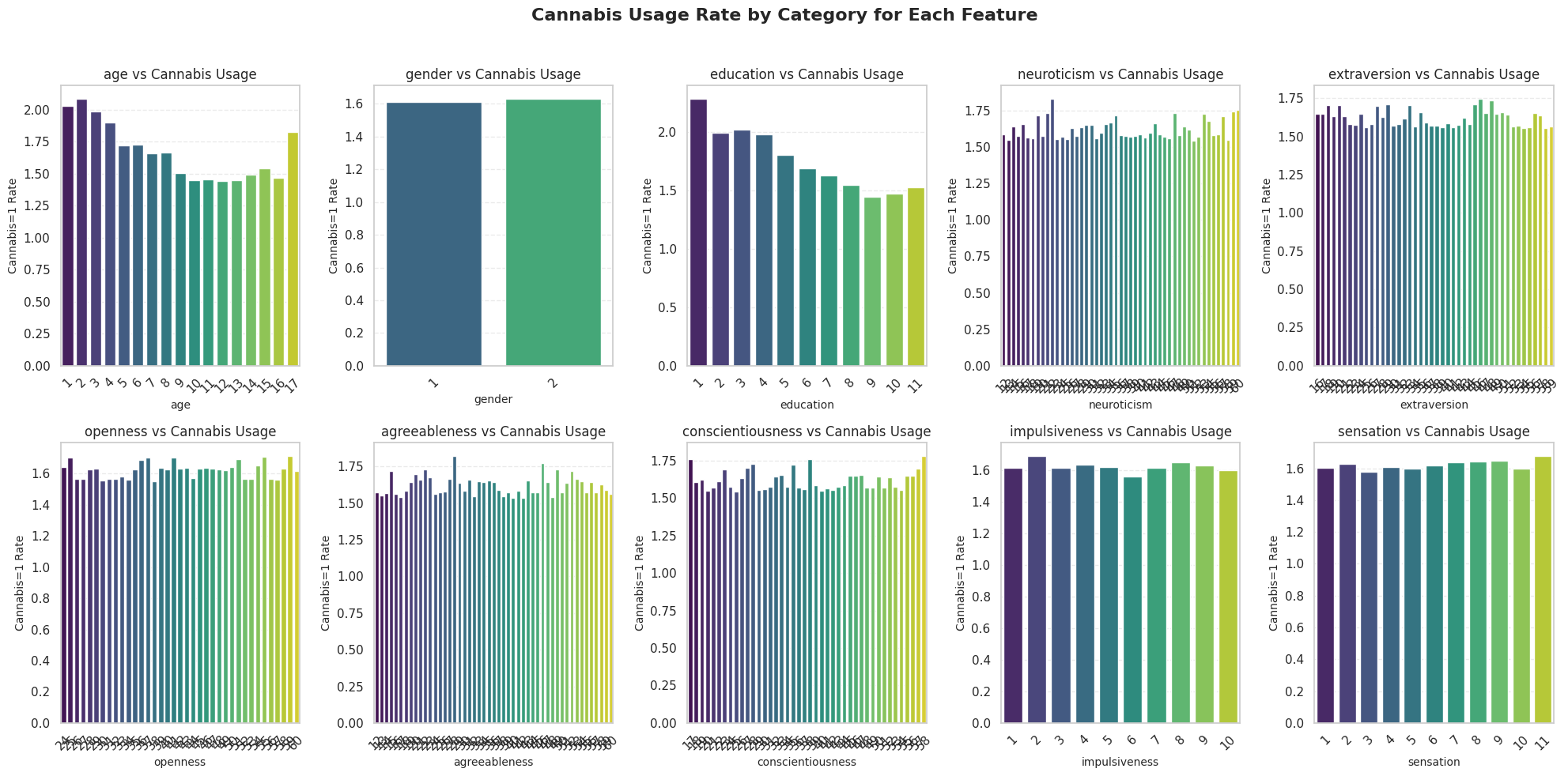
* + Age distribution is right-skewed, with more participants clustered in older age groups.
  + Personality traits show relatively uniform or slightly varied distributions, with some periodic spikes, indicating diverse emotional and behavioral profiles.
  + Education tends toward higher values, reflecting a relatively educated sample.

1. **Target Variable Distribution**



* Post-filtering, the target variable shows:
  + 43.1% of participants report cannabis use.
  + 56.9% report no use.
* The target is somewhat imbalanced .

1. **Cannabis Usage Rate Across Demographic and Personality Features.**



The figure displays the Cannabis=1 usage rate by category for each feature. Below are the general observations for each subplot:

1. Age vs Cannabis Usage

Younger age categories (e.g., 1–3) exhibit higher usage rates.

There is a declining trend in usage with increasing age, with a slight uptick at the highest age bin.

2. Gender vs Cannabis Usage

Cannabis usage appears fairly similar across genders, with only a minor difference in usage rates between the two categories.

3. Education vs Cannabis Usage

Higher cannabis usage is seen in lower education levels.

There's a declining trend in usage as education level increases.

4. Neuroticism vs Cannabis Usage

Slight increase in usage with higher neuroticism.

The usage rate is relatively stable across the spectrum with some fluctuation.

5. Extraversion vs Cannabis Usage

No strong trend but slight variation across levels.

Usage appears moderately high across most of the range.

6. Openness vs Cannabis Usage

Generally high and stable usage across all openness scores.

A mild increasing trend toward higher openness levels.

7. Agreeableness vs Cannabis Usage

Minor fluctuations with no clear pattern.

Overall, moderate usage rate.

8. Conscientiousness vs Cannabis Usage

Usage is relatively stable, with a small increasing trend.

Moderate to high rates throughout.

9. Impulsiveness vs Cannabis Usage

Cannabis usage is positively correlated with impulsiveness.

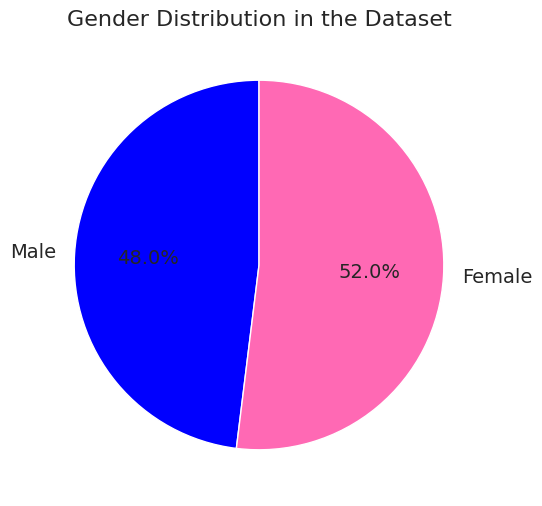
Higher impulsiveness levels show higher usage rates.

10. Sensation vs Cannabis Usage

Clear increasing trend: as sensation-seeking increases, so does cannabis usage.

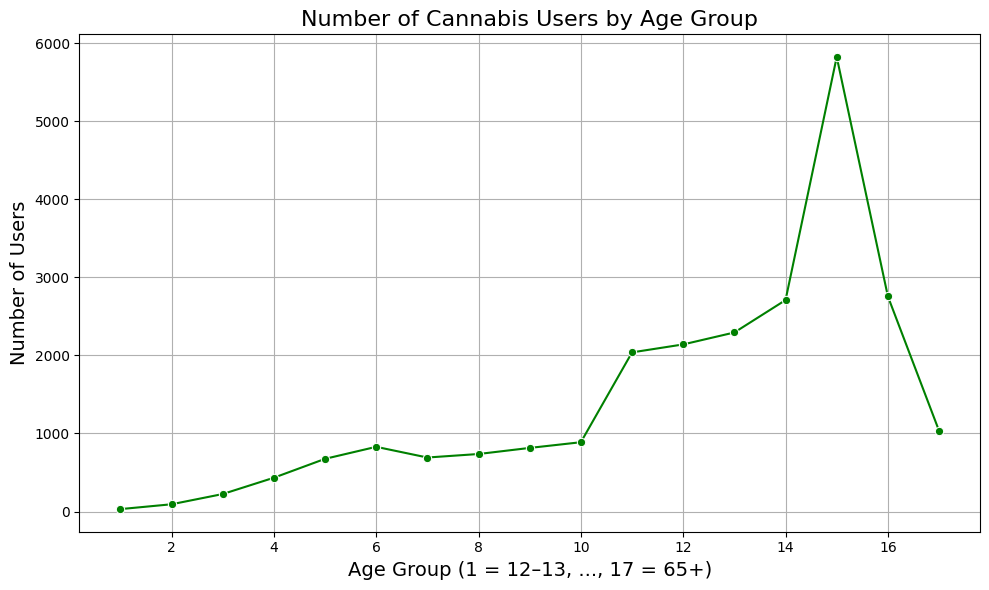
This may indicate a strong behavioral link between sensation-seeking and substance use.

**7.Gender and Cannabis Use**

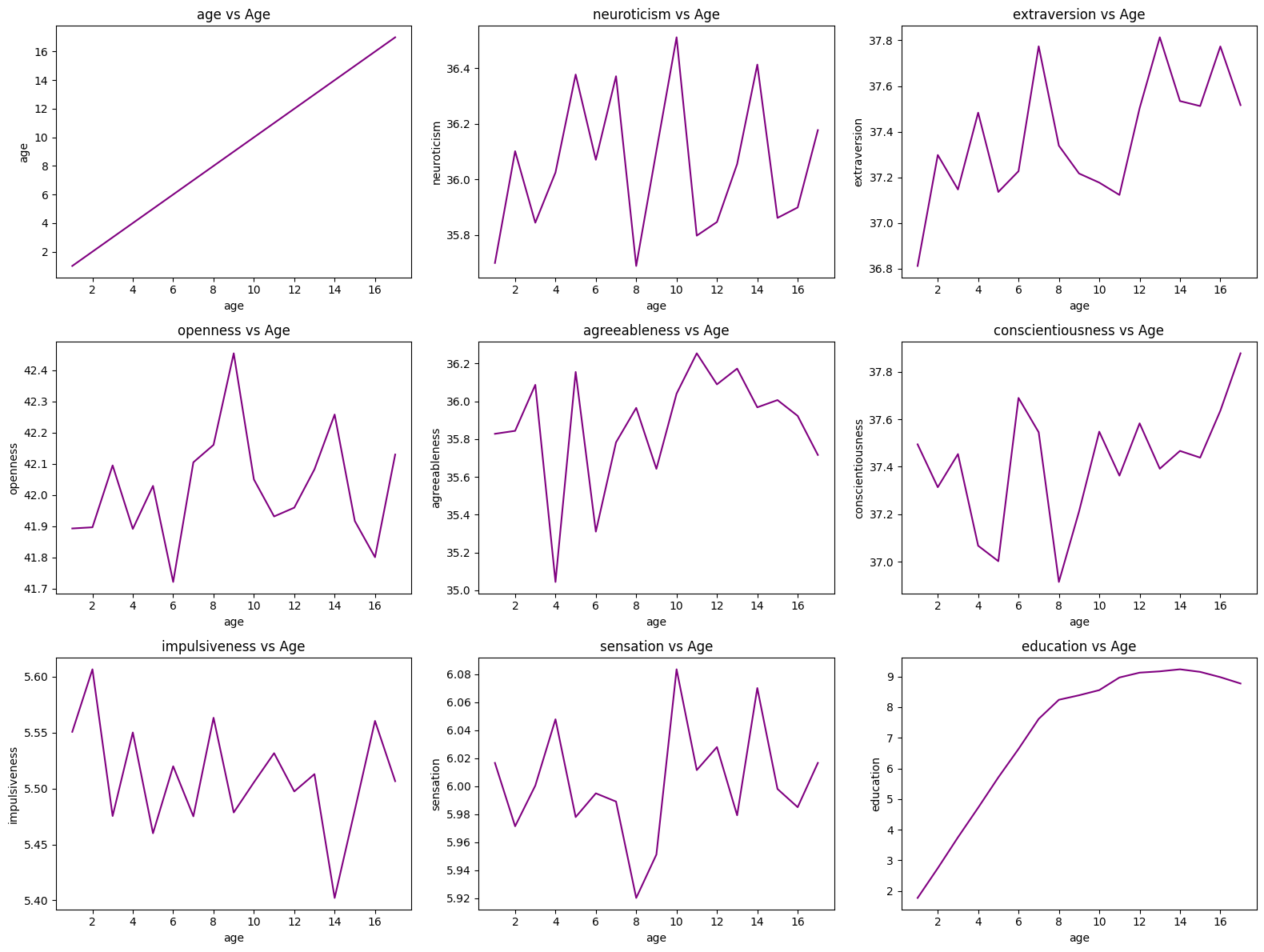


* Gender distribution is balanced overall.

**8.Age and Cannabis Use**

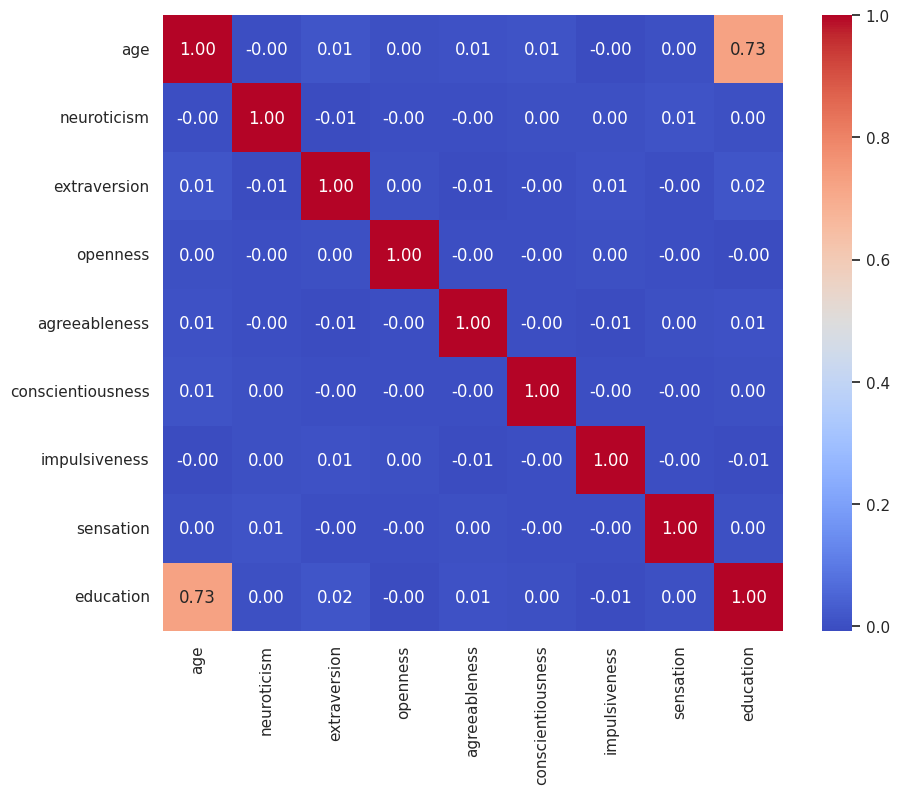


* The number of cannabis users increases with age ,confirming that older age groups have higher likelihood of cannabis use.



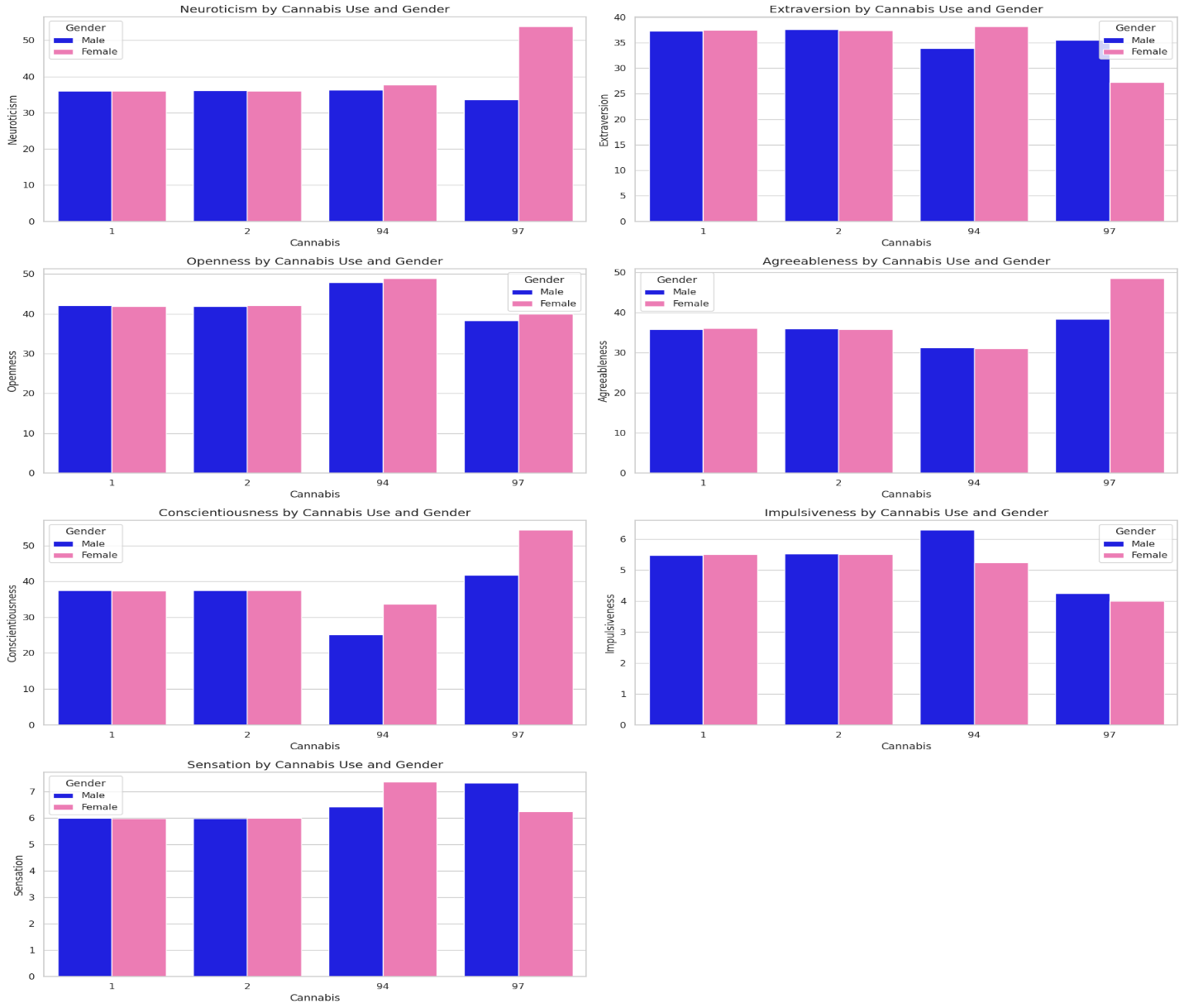
* Line plots show that personality traits fluctuate with age but no consistent linear trend emerges for most variables, except education which increases with age up to a point before plateauing.

**9.Correlation Analysis**



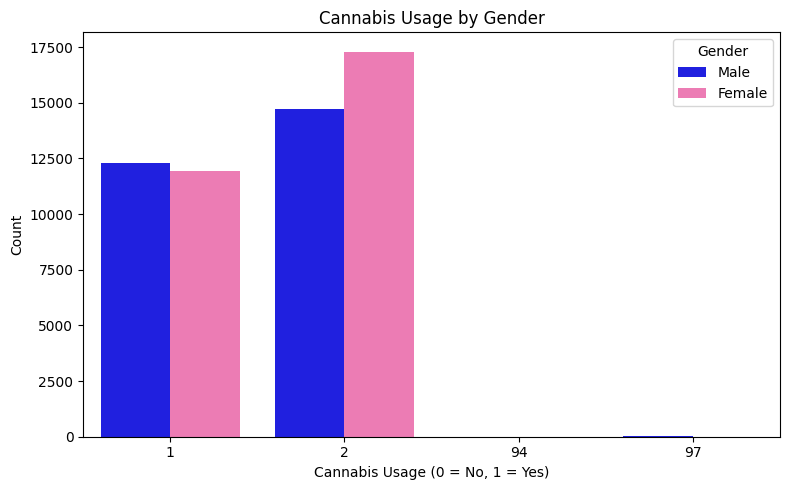
* Pearson correlation matrix reveals:
  + A strong positive correlation (r = 0.73) between age and education.
  + All other features show weak correlations (ranging between -0.01 and 0.02), suggesting independence between most variables.
* No multicollinearity issues were detected, making all features viable for modeling.

**10.Psychosocial Traits by Cannabis Use and Gender**



* Multi-faceted bar plots show:
  + Females score consistently higher on neuroticism and agreeableness.
  + Cannabis users exhibit slightly higher extraversion, openness, and sensation seeking.
  + Conscientiousness tends to be lower among cannabis users.
  + Impulsiveness varies with cannabis use, with some spikes among certain groups.

**11. Cannabis Usage by Gender**



A slightly higher number of females than males reported not using cannabis (if label 2 corresponds to 0 = No).

Among cannabis users (1 = Yes), males and females are nearly equal, with a slight edge possibly toward males.

12. Summary and Implications

* The dataset is well-prepared, clean, and rich with meaningful features representing demographics, personality, and behavior.
* Data distributions and correlations highlight important relationships, particularly age’s association with education and cannabis use.
* Gender and psychosocial traits exhibit expected patterns that could influence predictive modeling.
* Although no major data quality issues were detected, the dataset is imbalanced in terms of the target variable distribution, which will be addressed during preprocessing. Additionally, two outlier values in the target variable were identified and removed as part of the preprocessing steps to ensure more reliable model training.

**Data Preprocessing**

This part details the preprocessing steps applied to the dataset in preparation for training machine learning models aimed at predicting cannabis use. The preprocessing pipeline addresses data cleaning, target encoding, outlier treatment, feature scaling, and class imbalance handling. Each step is motivated by ensuring data quality, model robustness, and accurate predictive performance.

**Step 1: Target Variable Cleaning**

**Objective:**  
Ensure the target variable (cannabis) contains only clear and unambiguous class labels to avoid noise that could degrade model performance.

**Process:**

* The dataset initially contained the following values for the target variable:

| **Value** | **Meaning** | **Count** |
| --- | --- | --- |
| 1 | Cannabis user | 24214 |
| 2 | Cannabis non-user | 32031 |
| 94 | Unknown response | 15 |
| 97 | Refused to answer | 16 |

* Rows with values 94 (Unknown) and 97 (Refused to answer) were removed. These ambiguous responses could introduce noise and mislead the model.

**Result:**  
The cleaned target variable includes only:

* 1 for users
* 2 for non-users

**Step 2: Train-Test Split and Target Encoding**

**Objective:**  
Prepare data subsets for training and evaluation while maintaining consistent class distribution, and convert target labels to a binary format suitable for classification algorithms.

**Process:**

* The dataset was split into training (80%) and testing (20%) sets using stratified sampling on the target variable. This maintains the class ratio across both sets.
* Target labels were encoded as follows:
  + Original 1 (users) → 0
  + Original 2 (non-users) → 1

**Rationale:**  
Encoding to binary labels (0/1) aligns with the standard input for binary classification models and simplifies interpretation.

**Step 3: Numeric Feature Selection and Outlier Analysis**

**Objective:**  
Identify and prepare relevant numeric features for model input, ensuring that extreme outliers do not distort the model’s learning.

**Process:**

* Selected numeric features:['age', 'neuroticism', 'extraversion', 'openness', 'agreeableness', 'conscientiousness', 'impulsiveness', 'sensation']
* Outlier detection was conducted using the Interquartile Range (IQR) method with a factor of 1.5.
* No significant outliers were found in the training data, but a Winsorization step was included as a safeguard.

**Rationale:**  
Outliers can heavily skew model training. Including a Winsorizer ensures any potential extreme values introduced later (e.g., through data updates or errors) are capped to reasonable bounds.

**Step 4: Scaling and Outlier Treatment Pipeline**

**Objective:**  
Standardize feature scales to improve model convergence and ensure consistent numeric ranges across all features.

**Process:**

* Constructed a pipeline combining:
  + **Winsorizer:** Caps numeric feature values within the IQR boundaries.
  + **MinMaxScaler:** Scales features to the [0, 1] range.
* The pipeline was fitted on the training data and then applied to both training and test datasets.
* The fitted pipeline was serialized (saved) for consistent reuse.

**Rationale:**

* Winsorization prevents extreme values from distorting scaling and model training.
* MinMax scaling normalizes the data range, improving gradient-based model training efficiency.
* Applying the pipeline ensures consistent preprocessing for both training and test data.

**Step 5: Addressing Class Imbalance with SMOTE**

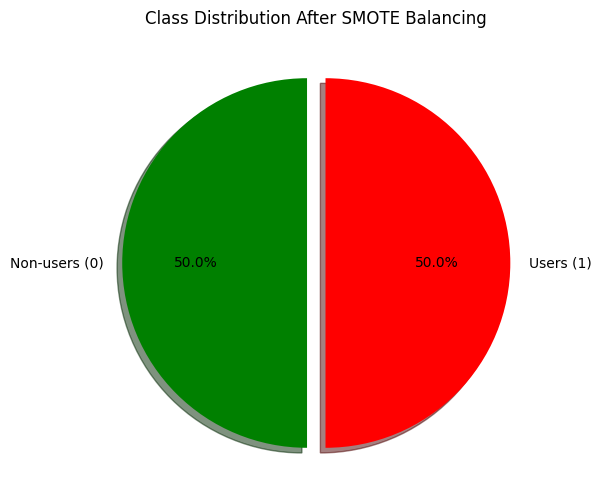
**Objective:**  
Balance the class distribution in the training set to prevent bias toward the majority class and improve model generalization.

**Process:**

* Initial class distribution in the training data was imbalanced:

| **Class** | **Count** |
| --- | --- |
| Users (0) | 25625 |
| Non-users (1) | 19371 |

* Applied **SMOTE** (Synthetic Minority Over-sampling Technique) *only on the training set* after scaling to generate synthetic samples of the minority class (non-users).
* Post-SMOTE, classes were perfectly balanced:



| **Class** | **Count** |
| --- | --- |
| Users (0) | 25625 |
| Non-users (1) | 25625 |

* Visualized the balanced distribution via a pie chart for verification.

**Rationale:**

* Training on imbalanced data can cause the model to favor the majority class, reducing predictive performance on the minority class.
* SMOTE generates realistic synthetic samples, helping the model learn a more balanced decision boundary.
* Applying SMOTE after scaling ensures synthetic data is generated in the scaled feature space, preserving feature distributions.

**Summary of the Processing Pipeline Order**

1. **Clean target variable:** Remove ambiguous target entries to ensure label clarity.
2. **Split data and encode target:** Stratified split to maintain distribution, encode targets for modeling.
3. **Select numeric features and detect outliers:** Identify relevant features and assess data quality.
4. **Build and apply pipeline (Winsorizer + MinMaxScaler):** Handle potential outliers and normalize features.
5. **Apply SMOTE on training data:** Balance classes to improve model fairness and accuracy.

Each step builds upon the previous to ensure the data fed into the model is clean, consistent, scaled, and balanced — prerequisites for robust and reliable machine learning.

**Modeling:**

**Experiment 1 – Baseline Modeling with All Features**

**Models used:**

**1.** **LogisticRegression:**  
Used as a simple baseline model for binary classification tasks. It is fast, interpretable, and effective when the relationship between features and the target is approximately linear.

Results:

**-Training metrics:**

Accuracy: 0.6271

Precision: 0.6613

Recall: 0.5212

F1 Score: 0.5829

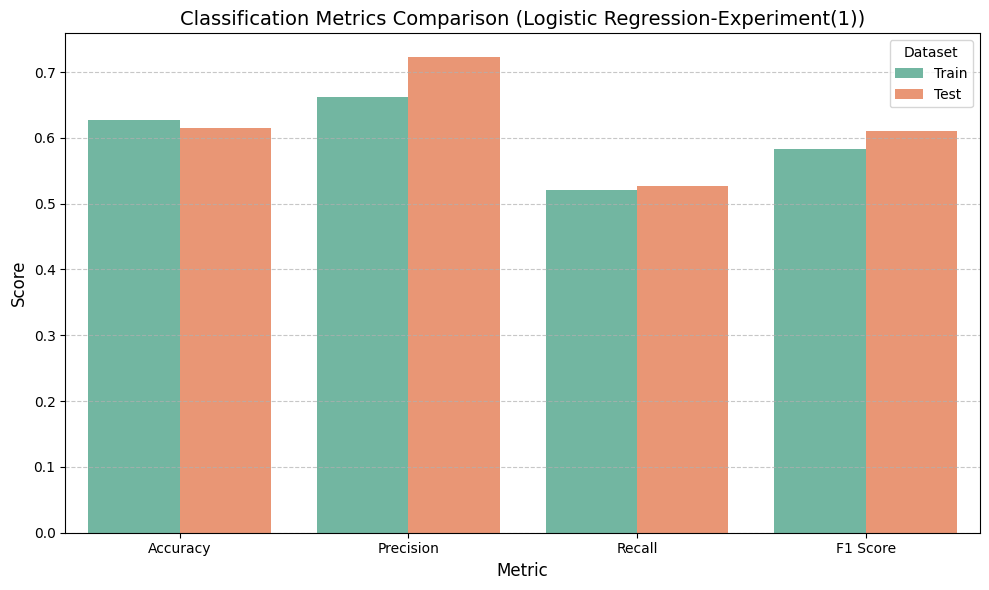
**-Testing metrics:**

Accuracy: 0.6156

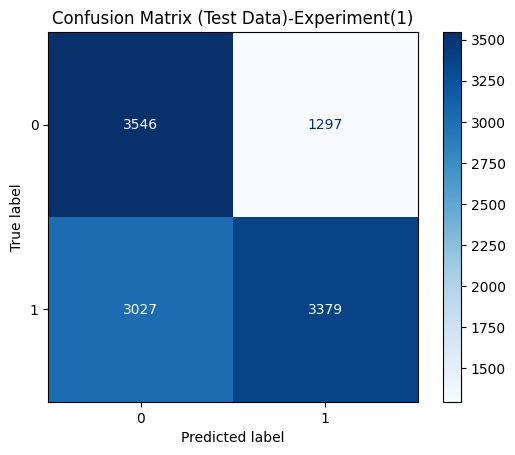
Precision: 0.7226

Recall: 0.5275

F1 Score: 0.6098



The Logistic Regression model achieves solid performance across all metrics, showing good balance between precision and recall, and stable behavior on both train and test sets.



The confusion matrix reveals that the model performs better in identifying class 0 than class 1, with a considerable number of false negatives that may impact sensitivity.

**We used the default parameters for LogisticRegression.**

**2. Decision Tree Regression:**

Chosen for its simplicity and ability to model non-linear relationships and feature interactions. It also provides easily interpretable decision rules.

Results:

**-Training metrics:**

Accuracy: 0.6271

Precision: 0.6613

Recall: 0.5212

F1 Score: 0.5829

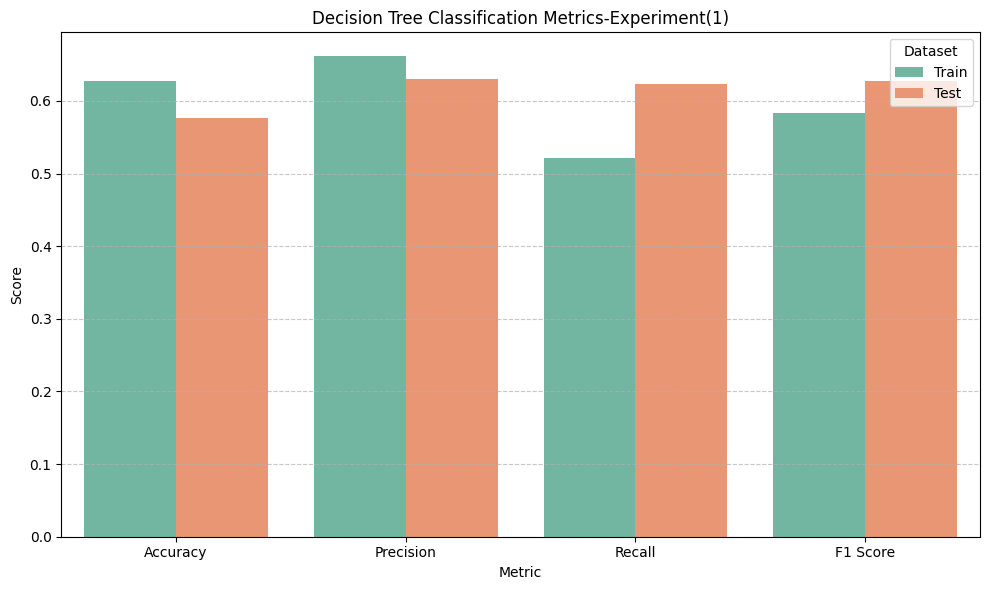
**-Testing metrics:**

Accuracy: 0.5769

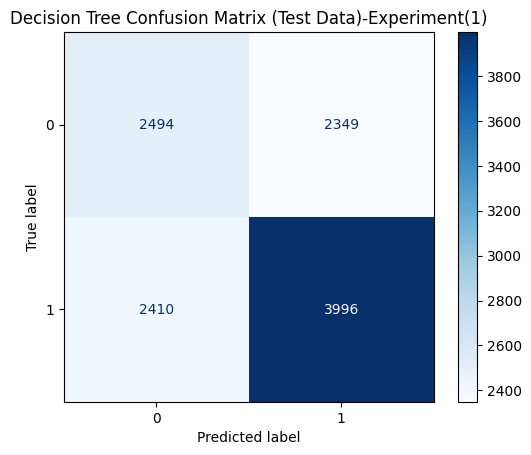
Precision: 0.6298

Recall: 0.6238

F1 Score: 0.6268



The model shows good precision and decent accuracy, but the recall on the training set is lower, indicating it may miss some relevant cases during learning.



The Decision Tree model has a relatively balanced prediction, but a notable number of misclassifications still exists, especially in predicting class 0 and class 1 correctly.

**We used the default parameters for Decision Tree**

**3.** **Random Forest**

An ensemble method that reduces overfitting compared to single decision trees by averaging multiple trees. It improves generalization and handles high-dimensional data well.

Results:

**-Training metrics:**

Accuracy: 1.0000

Precision: 1.0000

Recall: 1.0000

F1 Score: 1.0000

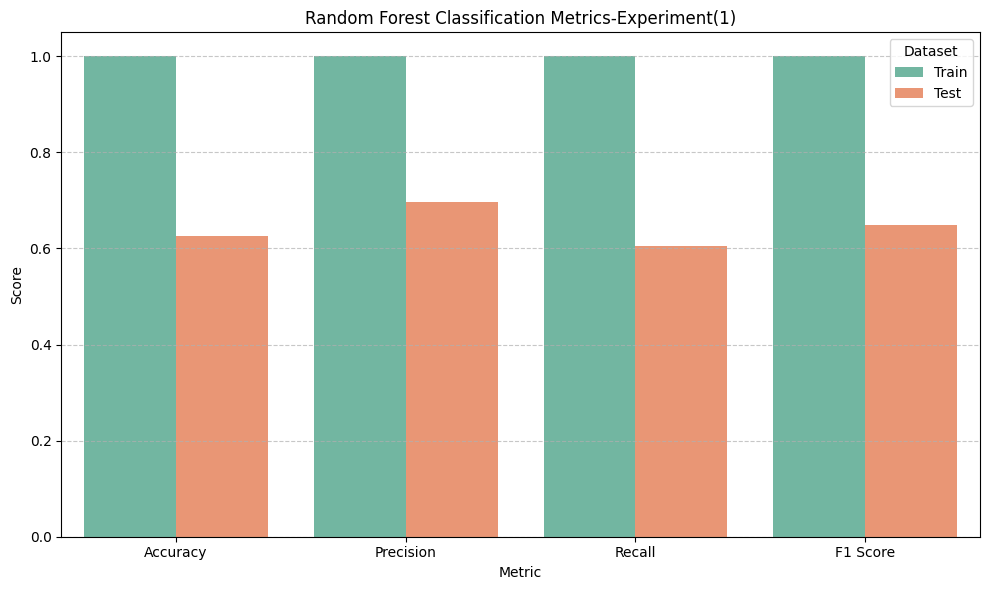
**-Testing metrics:**

Accuracy: 0.6253

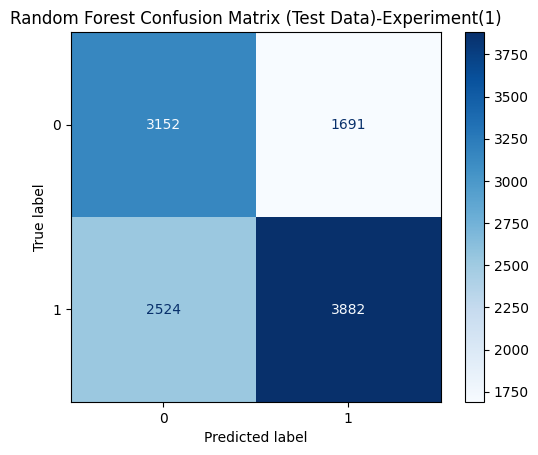
Precision: 0.6966

Recall: 0.6060

F1 Score: 0.6481



The model performs exceptionally well on training data, but there's a visible performance drop on test data, which may suggest overfitting.



The Random Forest model effectively identifies class 1, but shows notable confusion when predicting class 0, leading to a higher number of false positives.

**We used the default parameters for Random Forest**

**4.** **Gradient Boosting**

Selected for its strong predictive performance by sequentially building weak learners to correct errors from previous models. Effective for capturing complex patterns in the data.

Results:

**-Training metrics:**

Accuracy: 0.6796

Precision: 0.7536

Recall: 0.5336

F1 Score: 0.6248

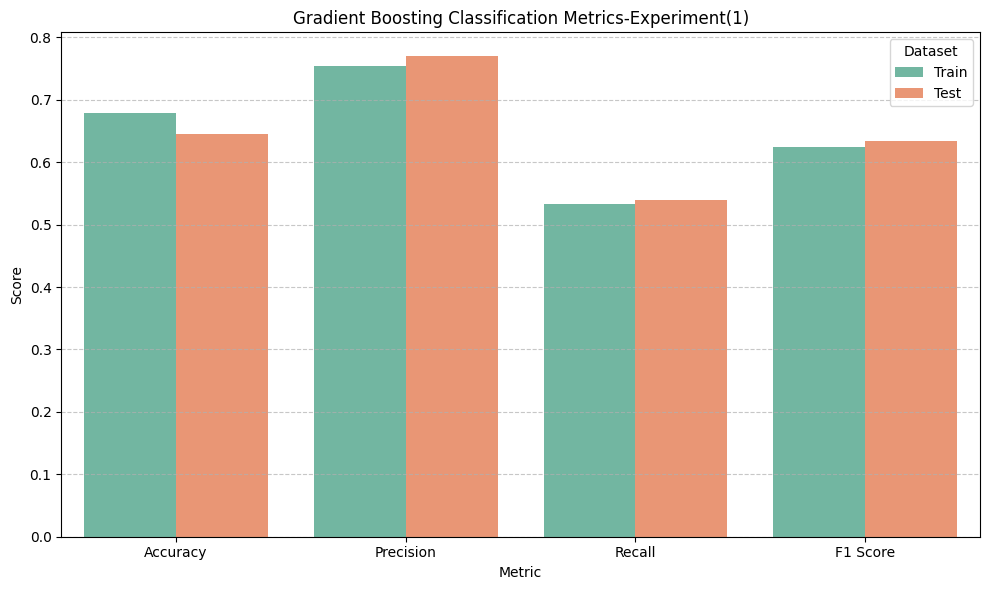
**-Testing metrics:**

Accuracy: 0.6456

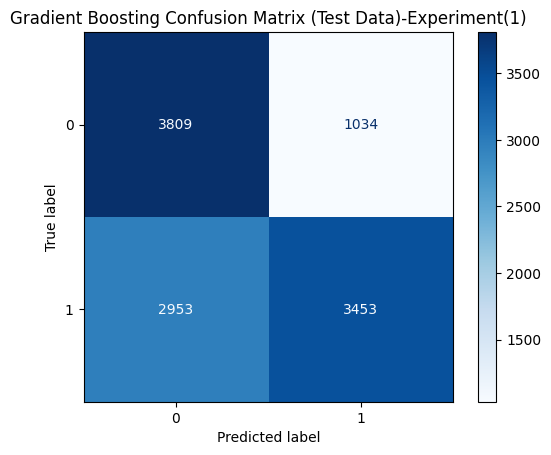
Precision: 0.7696

Recall: 0.5390

F1 Score: 0.6340



The model achieves high precision, indicating confidence in its positive predictions, while maintaining balanced performance across both training and testing sets.



The Gradient Boosting model shows strong performance in detecting class 0, but struggles more with class 1, as indicated by the high number of false negatives.

**We used the default parameters for Gradient Boosting**

**5.XGBOOST**  
An optimized implementation of gradient boosting that offers faster training and better regularization, making it suitable for large datasets and preventing overfitting.

Results:

**-Training metrics:**

Accuracy: 0.7728

Precision: 0.7950

Recall: 0.7351

F1 Score: 0.7639

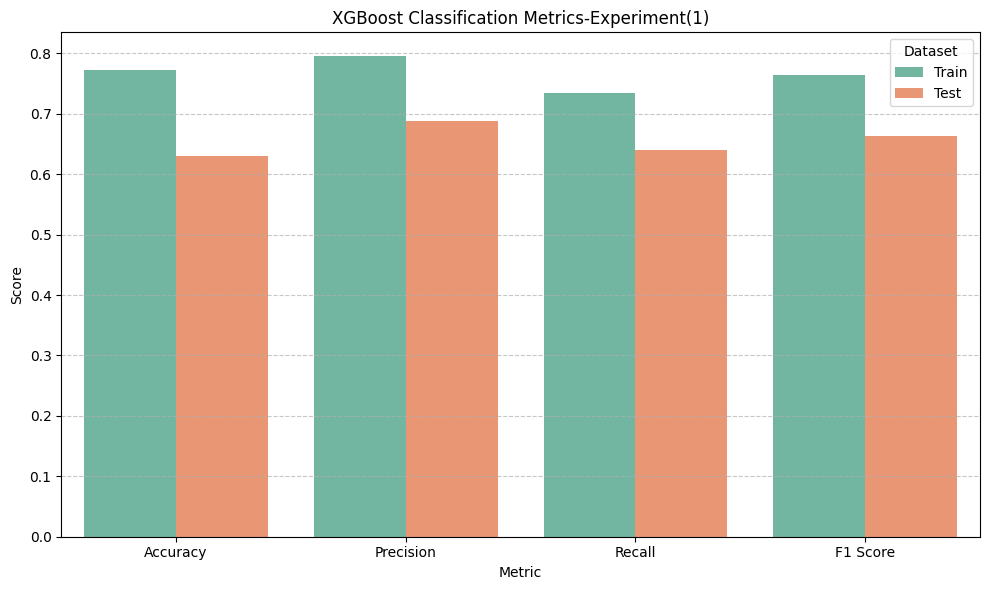
**-Testing metrics:**

Accuracy: 0.6300

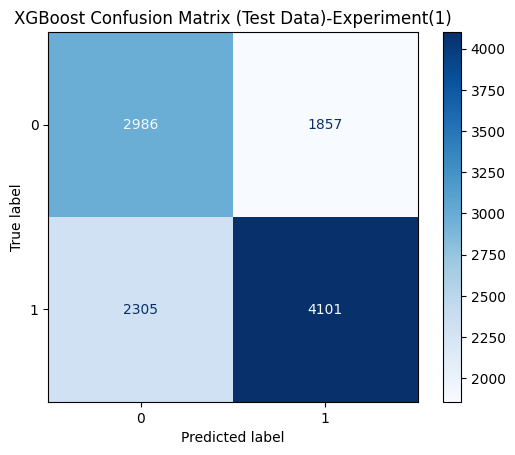
Precision: 0.6883

Recall: 0.6402

F1 Score: 0.6634



XGBoost shows excellent training performance, with a slight drop in test metrics, suggesting some overfitting, yet it still maintains competitive results across all evaluation scores.



The XGBoost model performs well overall, especially in identifying class 1, but has a noticeable number of false positives when predicting class 1 from class 0.

**We used the default parameters for XGBOOST.**

**6.** **LightGBM**

Chosen for its efficiency and scalability, especially with large datasets. It uses histogram-based algorithms and leaf-wise tree growth for faster and more accurate training.

Results:

**-Training metrics:**

Accuracy: 0.7104

Precision: 0.7552

Recall: 0.6228

F1 Score: 0.6826

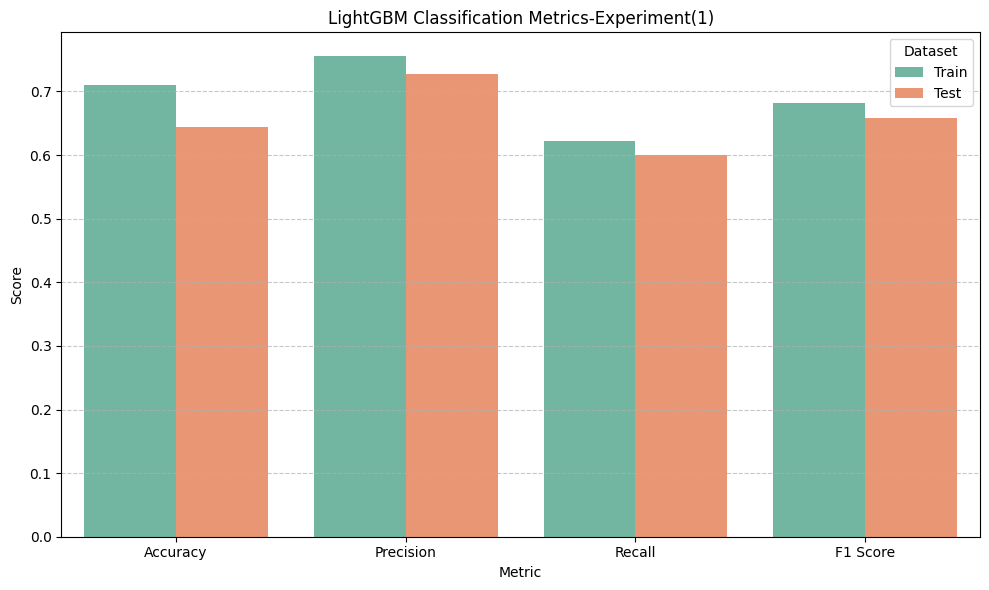
**-Testing metrics:**

Accuracy: 0.6441

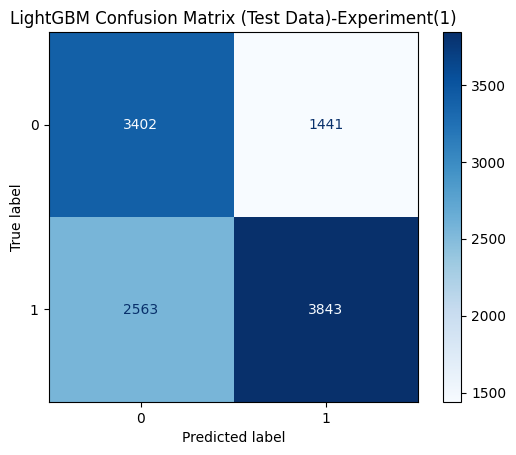
Precision: 0.7273

Recall: 0.5999

F1 Score: 0.6575



LightGBM maintains consistent and strong performance across accuracy, precision, recall, and F1-score, with minimal performance drop from training to test sets—indicating a well-generalized model.



The LightGBM model demonstrates solid classification performance, with relatively balanced results and a good ability to correctly identify both classes, though some misclassification still exists.

**We used the default parameters for LIGHTGBM.**

**7.** **CatBoost**  
Used because it handles categorical features natively and reduces the need for extensive preprocessing. It also prevents overfitting through ordered boosting.

Results:

**-Training metrics:**

Accuracy: 0.7571

Precision: 0.7893

Recall: 0.7015

F1 Score: 0.7428

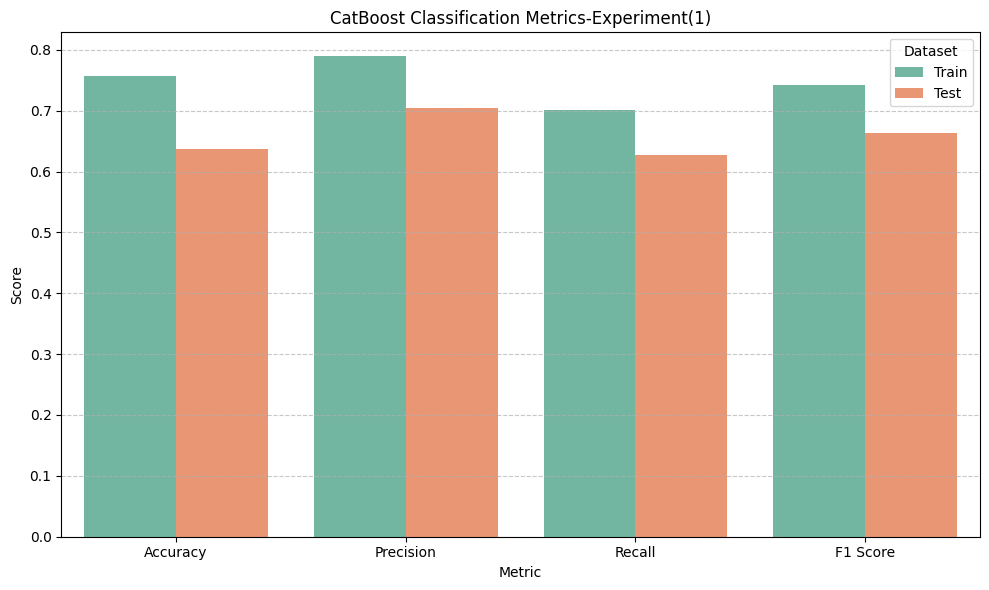
**-Testing metrics:**

Accuracy: 0.6374

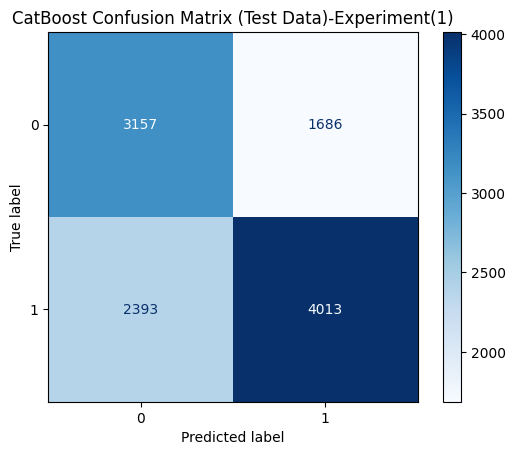
Precision: 0.7042

Recall: 0.6264

F1 Score: 0.6630



CatBoost demonstrates consistent performance across accuracy, precision, recall, and F1-score, with close values between training and test sets, indicating good generalization.



The CatBoost model achieves a relatively balanced classification, with fewer misclassifications compared to the total, showing stronger prediction accuracy on the test set.

**We used the default parameters for CATBOOST**

**8.** **K-Nearest Neighbors**

Used because it is a simple, instance-based learning algorithm that makes predictions based on the closest training examples. It works well when the decision boundary is irregular and does not assume any specific data distribution.

Results:

**-Training metrics:**

Accuracy: 0.7711

Precision: 0.8205

Recall: 0.6939

F1 Score: 0.7519

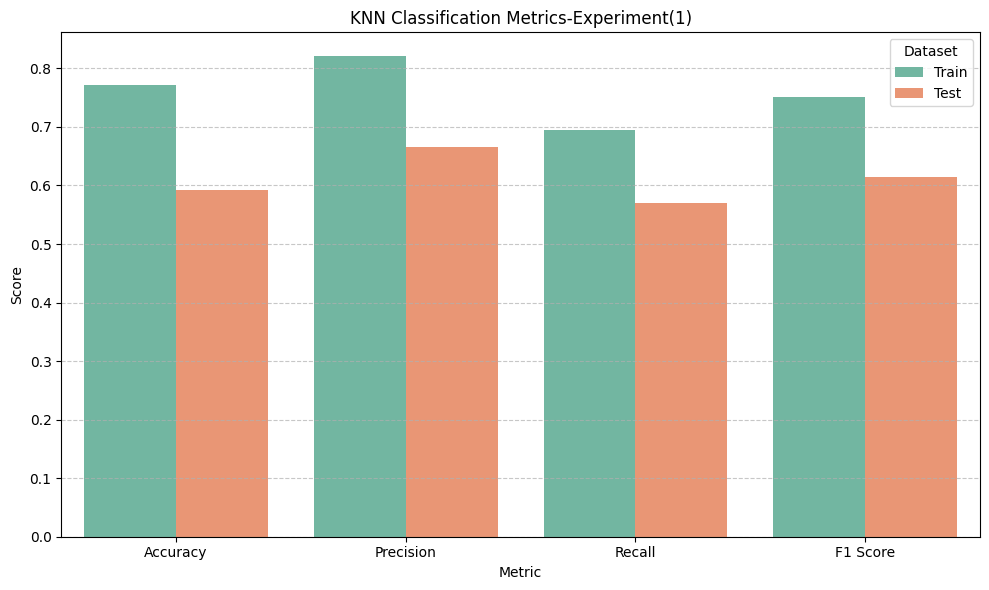
**-Testing metrics:**

Accuracy: 0.5924

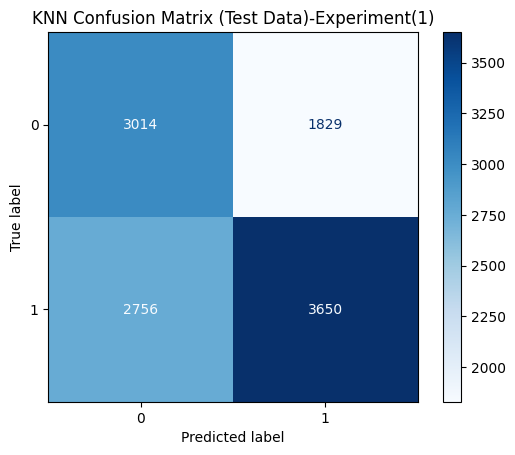
Precision: 0.6662

Recall: 0.5698

F1 Score: 0.6142



The performance metrics for KNN reveal a noticeable gap between training and testing scores, suggesting possible overfitting and limited generalization to unseen data.



This confusion matrix shows that the KNN model makes a considerable number of both false positives and false negatives, indicating room for improvement in classification reliability.

**We used the default parameters for KNN.**

**9-Naive Bayes**

Chosen for its simplicity and efficiency, especially on high-dimensional data. It assumes feature independence and works surprisingly well in many classification problems despite this assumption. It is fast and requires less training data.

Results:

**-Training metrics:**

Accuracy: 0.6406

Precision: 0.7364

Recall: 0.4380

F1 Score: 0.5493

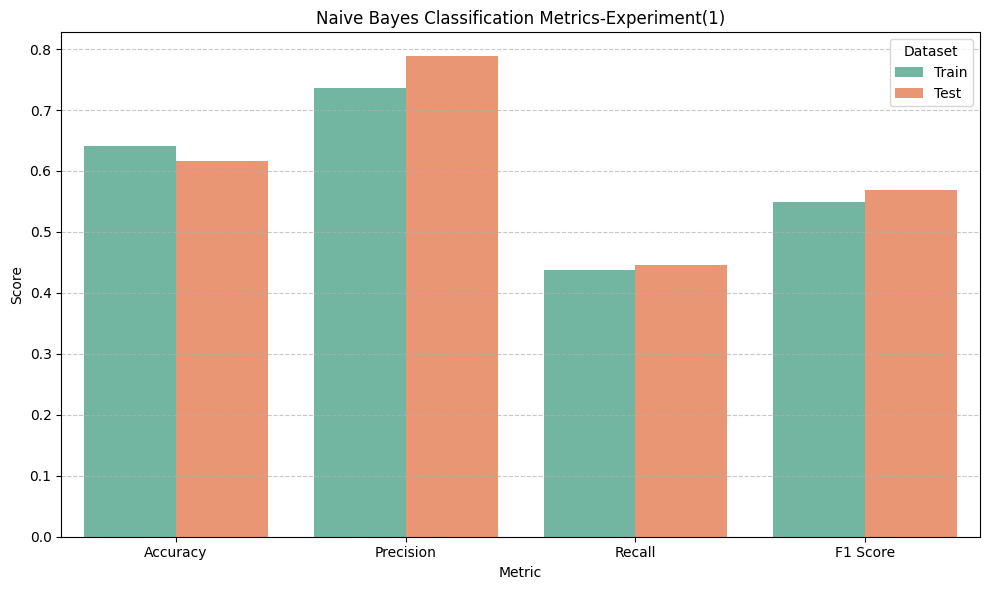
**-Testing metrics:**

Accuracy: 0.6159

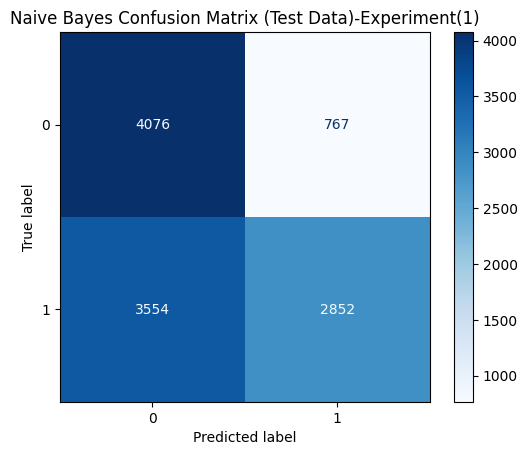
Precision: 0.7881

Recall: 0.4452

F1 Score: 0.5690



This chart shows that Naive Bayes maintains a good balance between precision and recall, with relatively consistent performance across train and test datasets.



This confusion matrix shows a somewhat balanced outcome for Naive Bayes, although there is still a considerable number of misclassifications for both classes.

**We used the default parameters for NAÏVE BAYES.**

**10.** **Support Vector Machine**

Selected for its effectiveness in high-dimensional spaces and its ability to find the optimal separating hyperplane maximizing the margin between classes. It works well with both linear and non-linear classification using kernel tricks.

Results:

**-Training metrics:**

Accuracy: 0.6271

Precision: 0.6613

Recall: 0.5212

F1 Score: 0.5829

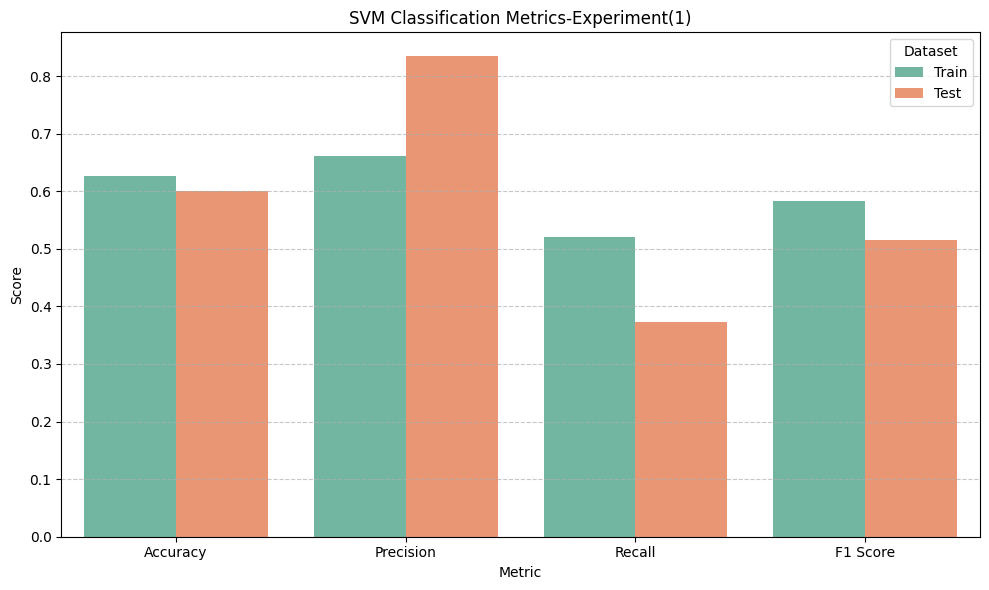
**-Testing metrics:**

Accuracy: 0.6007

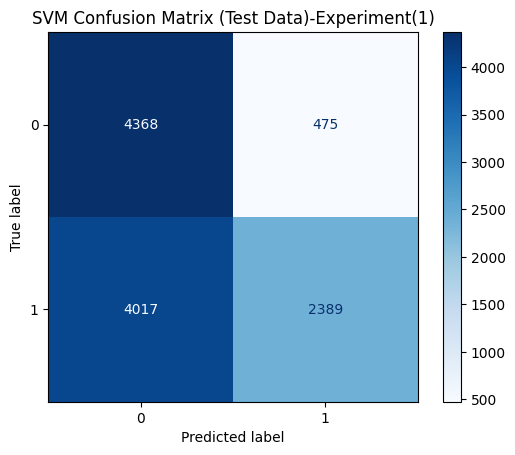
Precision: 0.8341

Recall: 0.3729

F1 Score: 0.5154



This chart highlights that the SVM model has high precision, but lower recall and F1 score, with a visible gap between training and testing performance.

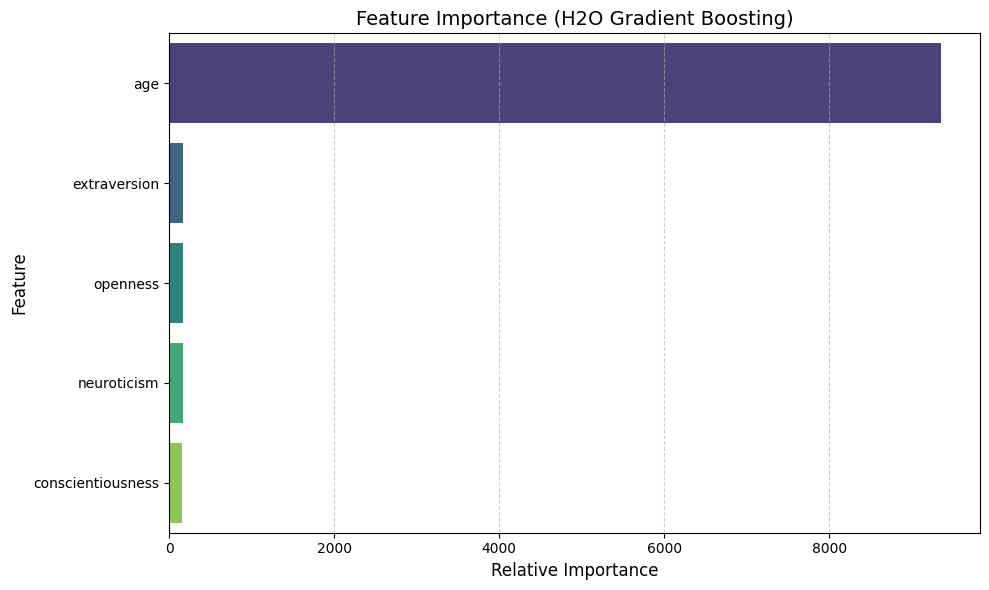


This matrix shows that the SVM model is highly accurate in detecting class 0 but struggles significantly with class 1, indicating a bias toward the negative class.

**We used the default parameters for SVC.**

**(Experiment (2))**

In this experiment, we applied H2O's feature selection method to select the most important features that influence cannabis. This helped improve the model's efficiency and accuracy.



**Experiment(2-1):using [ age, openness, extraversion, agreeableness, neuroticism, conscientiousness, sensation, impulsiveness ]**

**Models used:**

**1.** **LogisticRegression**

Results:

**-Training metrics:**

Accuracy: 0.6078

Precision: 0.6344

Recall: 0.5090

F1 Score: 0.56484

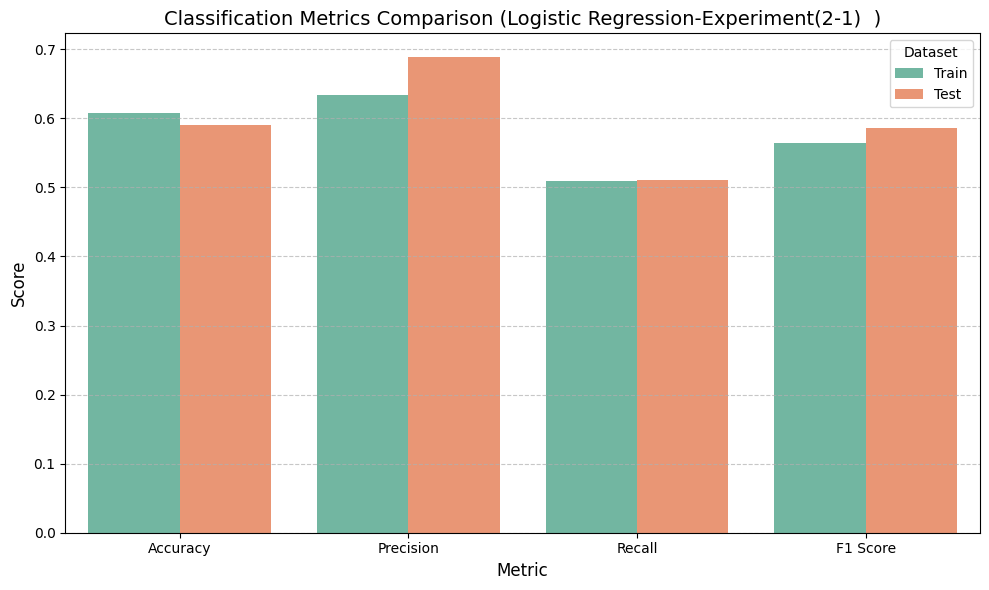
**-Testing metrics:**

Accuracy: 0.5897

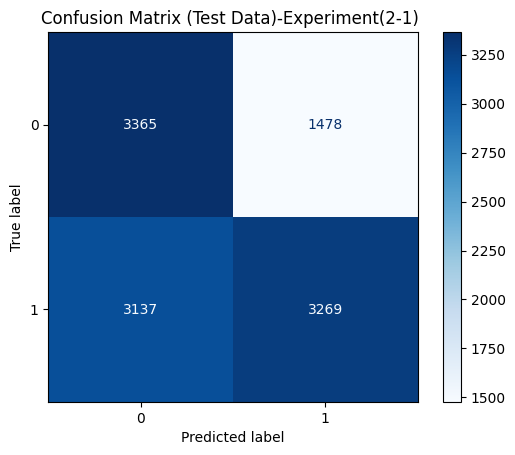
Precision: 0.6886

Recall: 0.5103

F1 Score: 0.5862



This chart demonstrates balanced performance between training and test sets, suggesting that the Logistic Regression model generalizes well without overfitting.



This confusion matrix shows that Logistic Regression predicts most samples reasonably well, with a relatively balanced classification across both classes despite some misclassifications.

**We used the default parameters for LogisticRegression.**

**2. Decision Tree Regression**

Results:

**-Training metrics:**

Accuracy: 1.0000

Precision: 1.0000

Recall: 1.0000

F1 Score: 1.0000

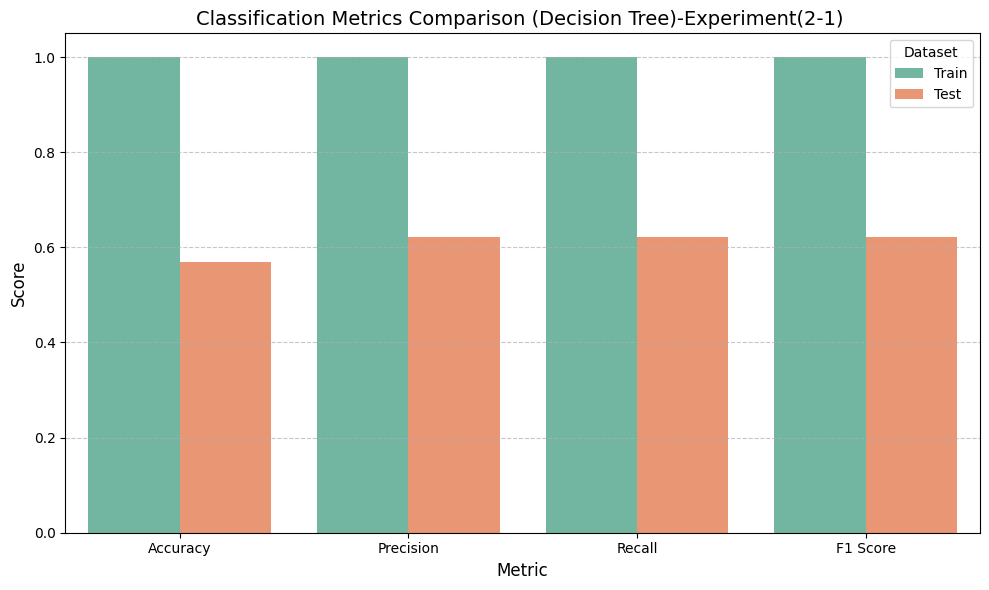
**-Testing metrics:**

Accuracy: 0.5684

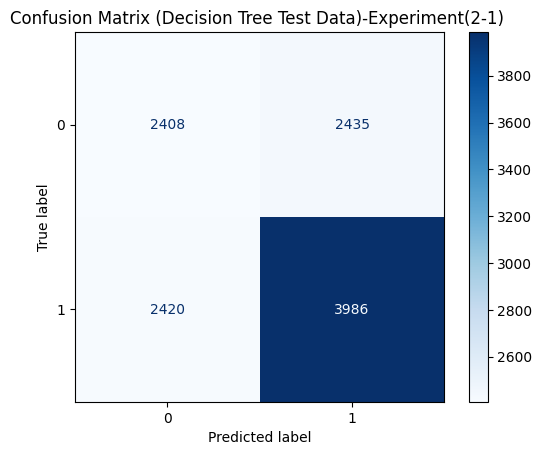
Precision: 0.6208

Recall: 0.6222

F1 Score: 0.6215



This chart reveals excellent training performance but significantly worse test performance, indicating that the model is overfitting the training data.



This matrix shows that the Decision Tree struggles with both classes, misclassifying a high number of samples from class 0 and class 1, indicating poor generalization.

**We used the default parameters for Decision Tree.**

**3.** **Random Forest**

Results:

**-Training metrics:**

Accuracy: 1.0000

Precision: 1.0000

Recall: 1.0000

F1 Score: 1.0000

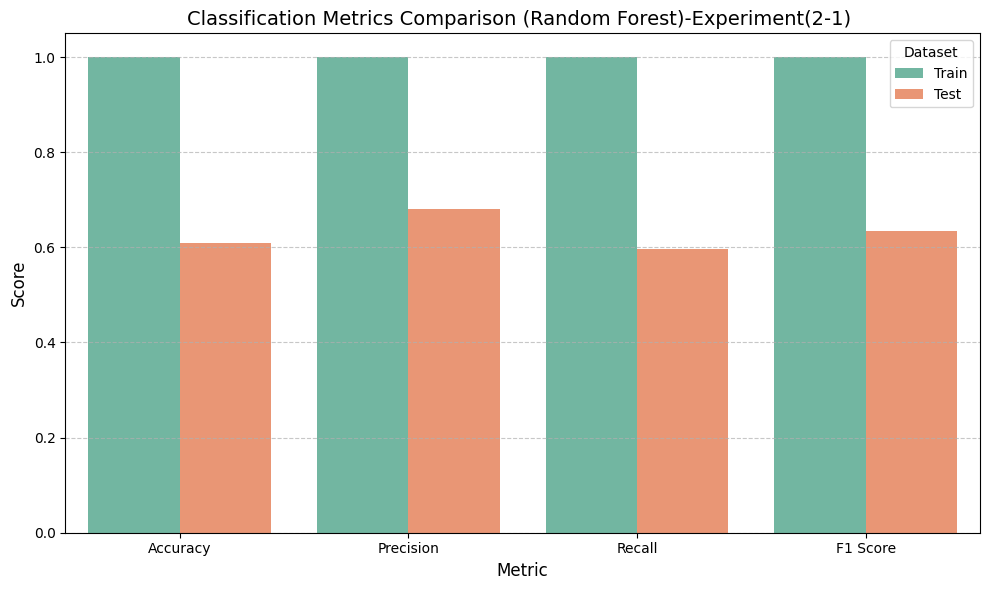
**-Testing metrics:**

Accuracy: 0.6100

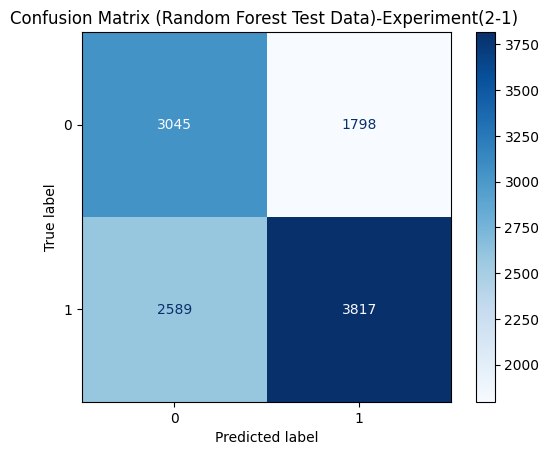
Precision: 0.6798

Recall: 0.5958

F1 Score: 0.6351



This chart presents the model performance of Random Forest on both train and test datasets across Accuracy, Precision, Recall, and F1 Score. It helps visualize how well the model generalizes to new data.



This matrix illustrates the classification performance of the Random Forest model on test data. It shows true positives and true negatives on the diagonal (3045 and 3817), and false positives/negatives on the off-diagonal (1798 and 2589).

**We used the default parameters for Random Forest.**

**4.** **Gradient Boosting**

Results:

**-Training metrics:**

Accuracy: 0.6736

Precision: 0.7734

Recall: 0.4911

F1 Score: 0.6007

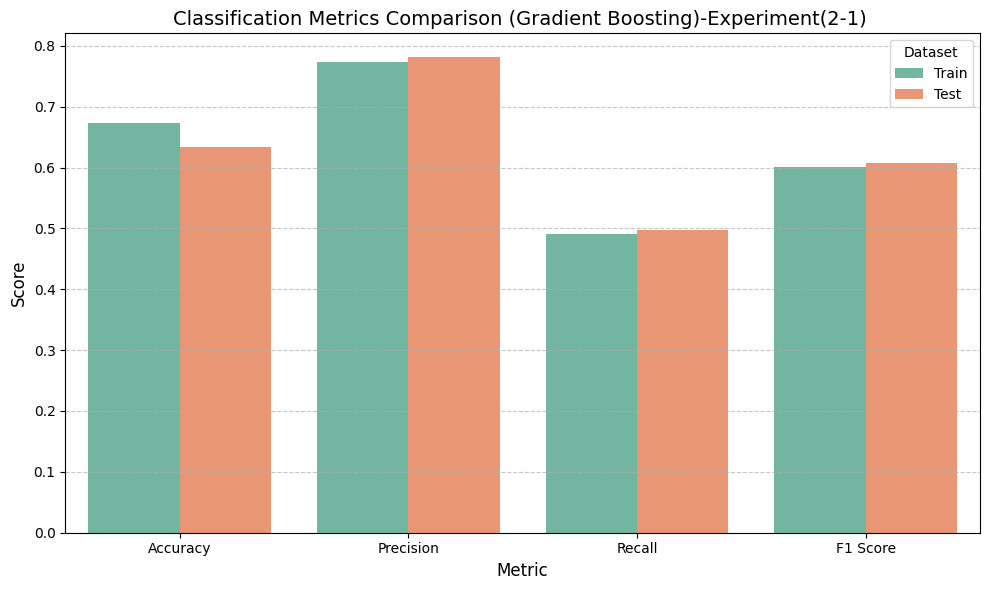
**-Testing metrics:**

Accuracy: 0.6343

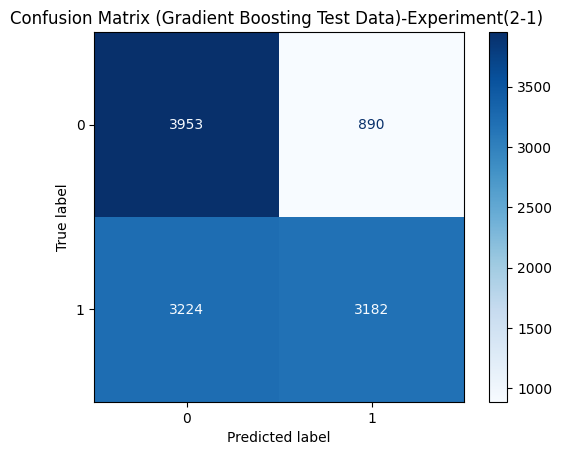
Precision: 0.7814

Recall: 0.4967

F1 Score: 0.6074



This bar chart compares training and test scores of the Gradient Boosting model using four metrics: Accuracy, Precision, Recall, and F1 Score. It highlights the performance consistency between the train and test datasets.



This matrix shows the performance of the Gradient Boosting model on test data. It visualizes the number of correct and incorrect predictions for each class. The diagonal values (3953 and 3182) represent correct predictions, while the off-diagonal values (890 and 3224) represent misclassifications.

**We used the default parameters for Gradient Boosting.**

**5.** **XGBoost**

Results:

**-Training metrics:**

Accuracy: 0.7601

Precision: 0.7933

Recall: 0.7035

F1 Score: 0.7457

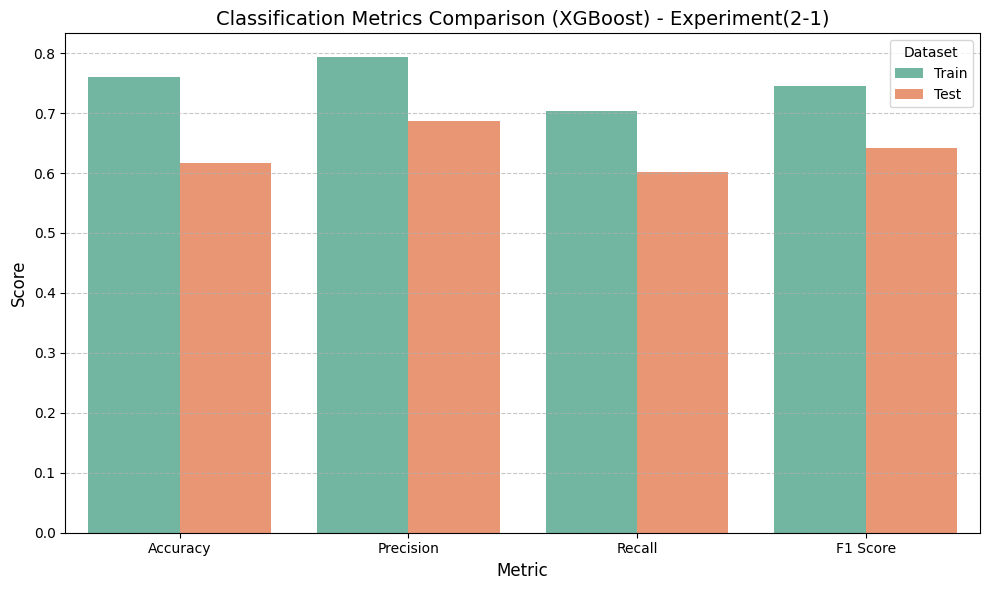
**-Testing metrics:**

Accuracy: 0.6173

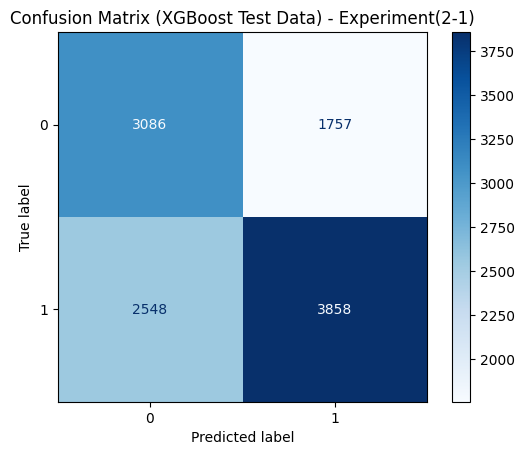
Precision: 0.6871

Recall: 0.6022

F1 Score: 0.6419



This chart presents XGBoost’s evaluation metrics for both train and test datasets. It includes Accuracy, Precision, Recall, and F1 Score, allowing insight into the model’s generalization and consistency.



This matrix visualizes how the XGBoost model performs on the test set. It shows true positives and true negatives (3086 and 3858), as well as the misclassifications (1757 and 2548), helping evaluate the balance between classes.

**We used the default parameters for XGBOOST.**

**6.** **LightGBM**

Results:

**-Training metrics:**

Accuracy: 0.6952

Precision: 0.7744

Recall: 0.5509

F1 Score: 0.6438

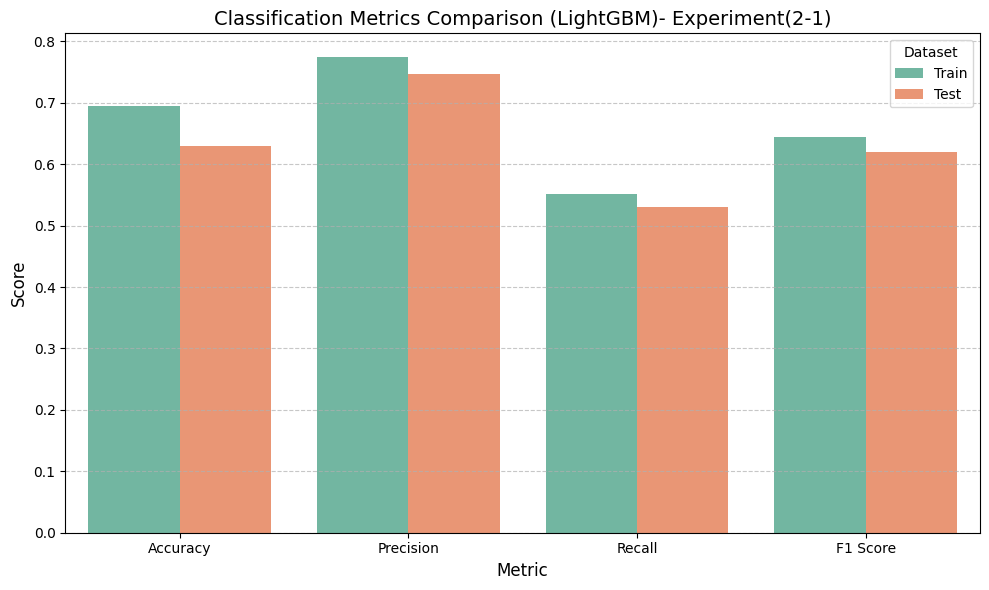
**-Testing metrics:**

Accuracy: 0.6300

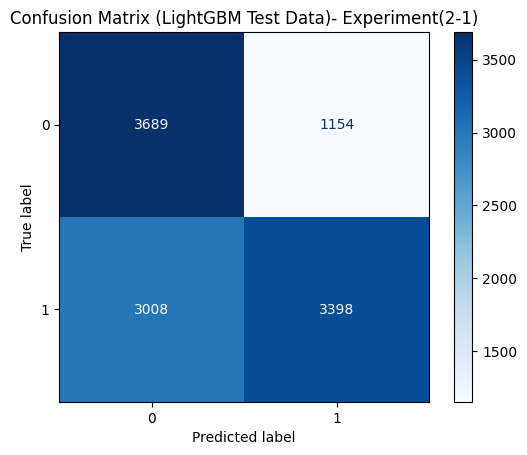
Precision: 0.7465

Recall: 0.5304

F1 Score: 0.6202



This bar chart compares LightGBM performance on training and test data using Accuracy, Precision, Recall, and F1 Score. It visualizes how closely the model's performance on unseen data matches its training performance.



This matrix shows the classification performance of the LightGBM model on the test data. It displays correct predictions (3689 and 3398) on the diagonal and incorrect ones (1154 and 3008) off the diagonal, highlighting how well the model distinguishes between classes.

**We used the default parameters for LIGHTGBM.**

**7.** **CatBoost**

Results:

**-Training metrics:**

Accuracy: 0.7434

Precision: 0.7982

Recall: 0.6516

F1 Score: 0.7175

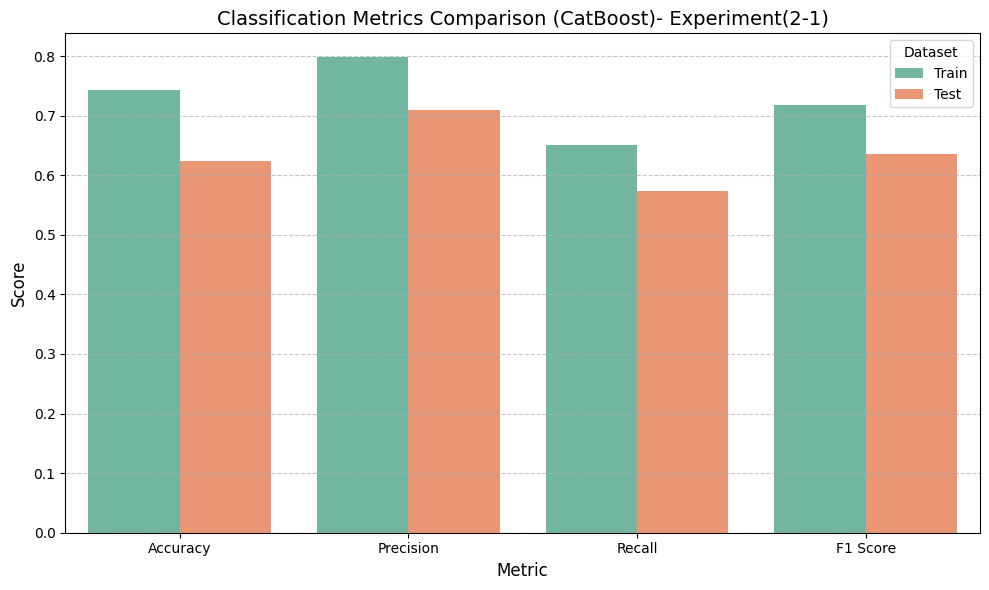
**-Testing metrics:**

Accuracy: 0.6239

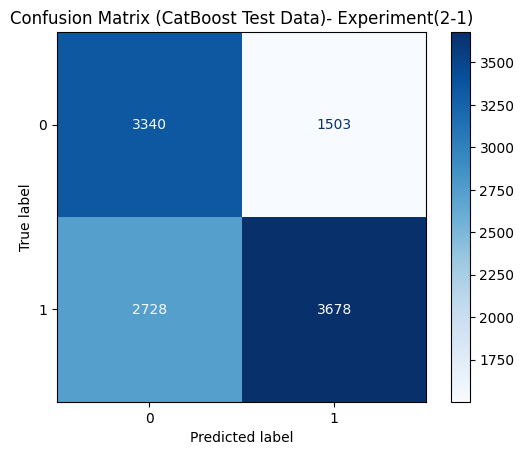
Precision: 0.7099

Recall: 0.5741

F1 Score: 0.6348



This bar chart compares the classification performance metrics — Accuracy, Precision, Recall, and F1 Score — for the CatBoost classifier on both the training and test datasets.



This confusion matrix displays the performance of the CatBoost classifier on the test dataset, indicating how many predictions were correctly or incorrectly classified for each actual class.

**We used the default parameters for CATBOOST.**

**8.** **K-Nearest Neighbors**

Results:

**-Training metrics:**

Accuracy: 0.7659

Precision: 0.8103

Recall: 0.6944

F1 Score: 0.7479

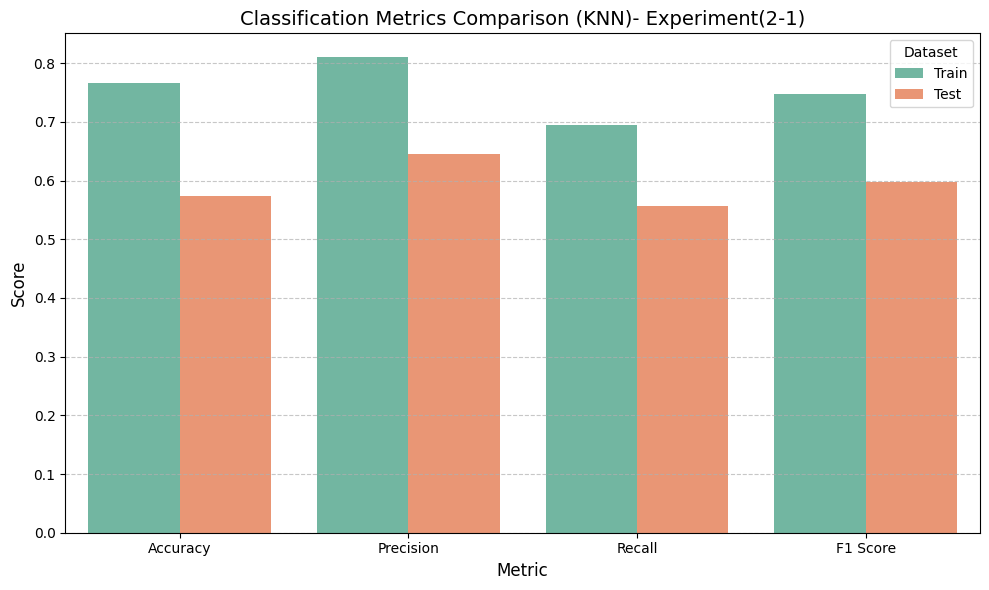
**-Testing metrics:**

Accuracy: 0.5736

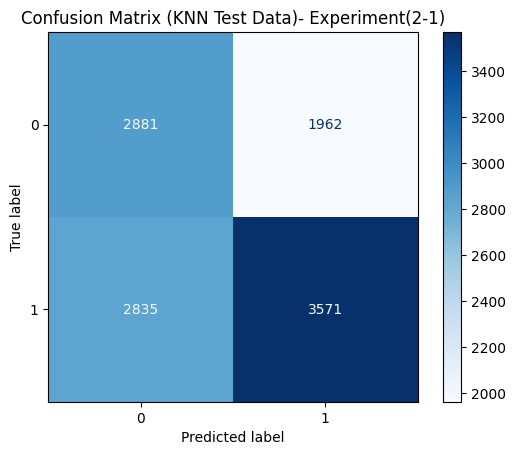
Precision: 0.6454

Recall: 0.5574

F1 Score: 0.5982



This bar chart displays how the KNN model performs on training and test datasets using Accuracy, Precision, Recall, and F1 Score, giving a clear overview of generalization capability.



This matrix shows the performance of the K-Nearest Neighbors (KNN) model on test data. It presents correct predictions (2881 and 3571) and misclassifications (1962 and 2835), allowing analysis of classification accuracy.

**We used the default parameters for KNN.**

**9-Naive Bayes**

Results:

**-Training metrics:**

Accuracy: 0.6291

Precision: 0.7066

Recall: 0.4415

F1 Score: 0.5434

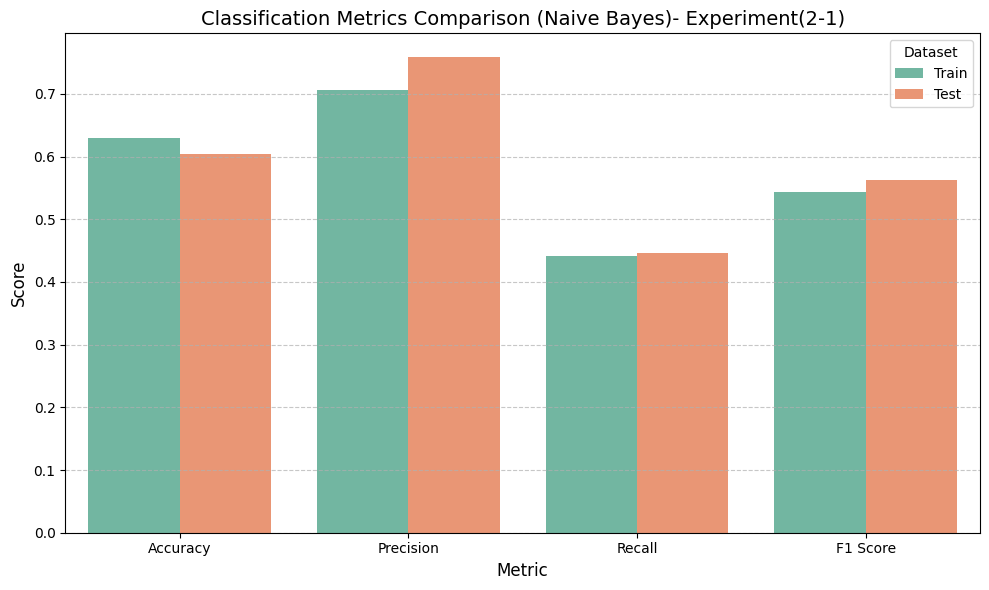
**-Testing metrics:**

Accuracy: 0.6038

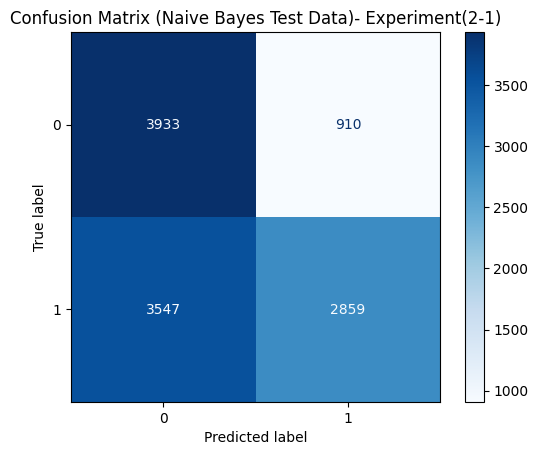
Precision: 0.7586

Recall: 0.4463

F1 Score: 0.5620



This chart shows Naive Bayes metrics for both training and test datasets, covering Accuracy, Precision, Recall, and F1 Score, helping assess consistency and potential overfitting or underfitting.



This matrix illustrates Naive Bayes performance on test data. Correct classifications appear on the diagonal (3933 and 2859), while incorrect ones are off-diagonal (910 and 3547), offering insight into its prediction distribution.

**We used the default parameters for NAÏVE BAYES.**

**10.** **Support Vector Machine**

Results:

**-Training metrics:**

Accuracy: 0.6424

Precision: 0.7556

Recall: 0.4210

F1 Score: 0.5407

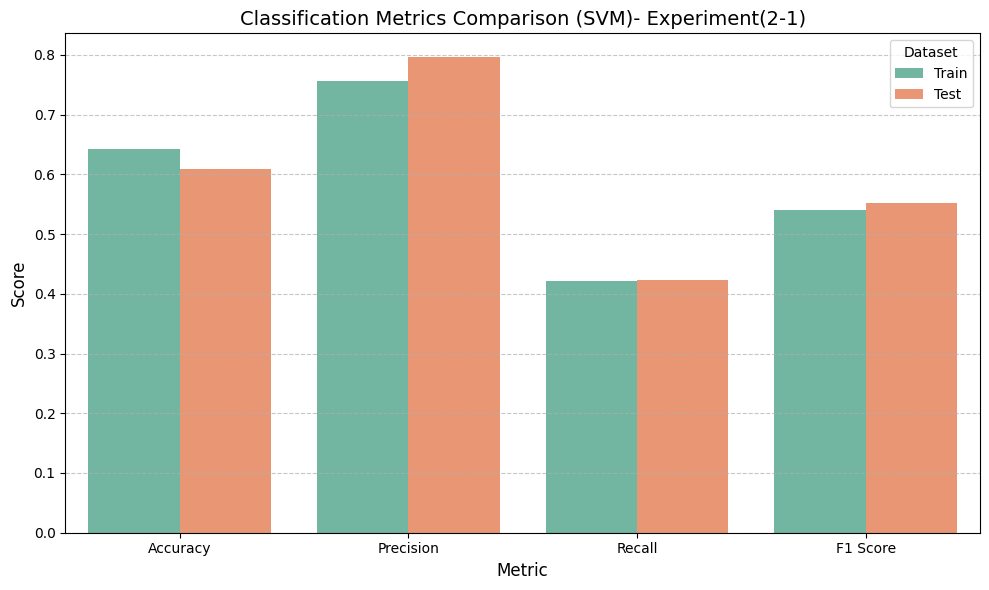
**-Testing metrics:**

Accuracy: 0.6096

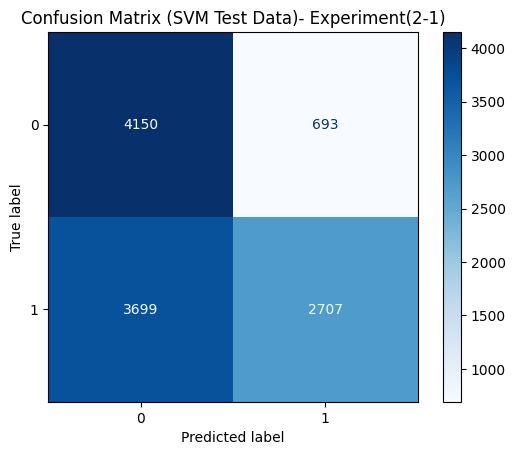
Precision: 0.7962

Recall: 0.4226

F1 Score: 0.5521



This bar chart compares the classification performance metrics — Accuracy, Precision, Recall, and F1 Score — for the SVM classifier on both the training and test datasets.



This confusion matrix shows the performance of the Support Vector Machine (SVM) classifier on the test dataset. It visualizes the number of correct and incorrect predictions categorized by actual and predicted labels.

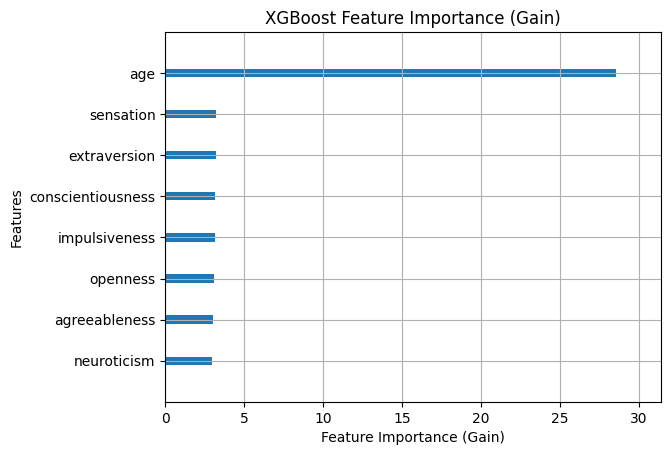
**We used the default parameters for SVC.**

**Experiment(2-2):using [age, conscientiousness, extraversion, neuroticism, openness]**

After applying feature selection using the H2OGradientBoostingEstimator algorithm, we observed a noticeable degradation in the performance of all classification models. The F1 scores dropped significantly and became nearly identical across different models (~0.413), which strongly indicates that the selected features were insufficient for effectively distinguishing between the target classes.

A likely reason for this is that the algorithm selected a limited and non-representative subset of features, leading to the loss of critical information relevant to classification. Moreover, H2O converts the data into its own internal format (H2OFrame), which might have affected the data representation quality due to automatic processing steps such as categorical encoding or missing value handling.

As a result, we decided to adopt an alternative and more transparent feature selection method based on feature importance extracted from the XGBoost model. In this approach, the importance of each feature is measured by its actual contribution to improving model performance (using metrics such as Gain and Cover). This technique provided better insights into the relationship between features and the target variable, and ultimately led to improved performance across the models.



Feature Importance Analysis and Decision on Feature Selection Approach

The feature importance scores obtained from the XGBoost classifier using the ‘gain’ metric revealed a clear ranking of predictive variables, with features such as age, sensation, and extraversion showing the highest contribution to the model’s performance. These results align closely with the importance patterns observed previously using the H2OGradientBoostingEstimator, confirming the consistency and reliability of these features in explaining the variance in the target variable.

Given the similarity in feature importance rankings between XGBoost and H2O, it was deemed unnecessary to continue with the fully manuall feature elimination process, which in prior attempts led to degraded model performance. Instead, a more controlled and interpretable manual feature selection approach was adopted, focusing on the top five most important features identified by XGBoost.

This strategy aims to balance dimensionality reduction and model interpretability while preserving critical predictive information, ultimately enhancing the model’s F1 Score and ensuring a better balance between precision and recall.

**Models used:**

**1.** **LogisticRegression**

Results:

**-Training metrics:**

Accuracy: 0.6058

Precision: 0.6306

Recall: 0.5108

F1 Score: 0.5644

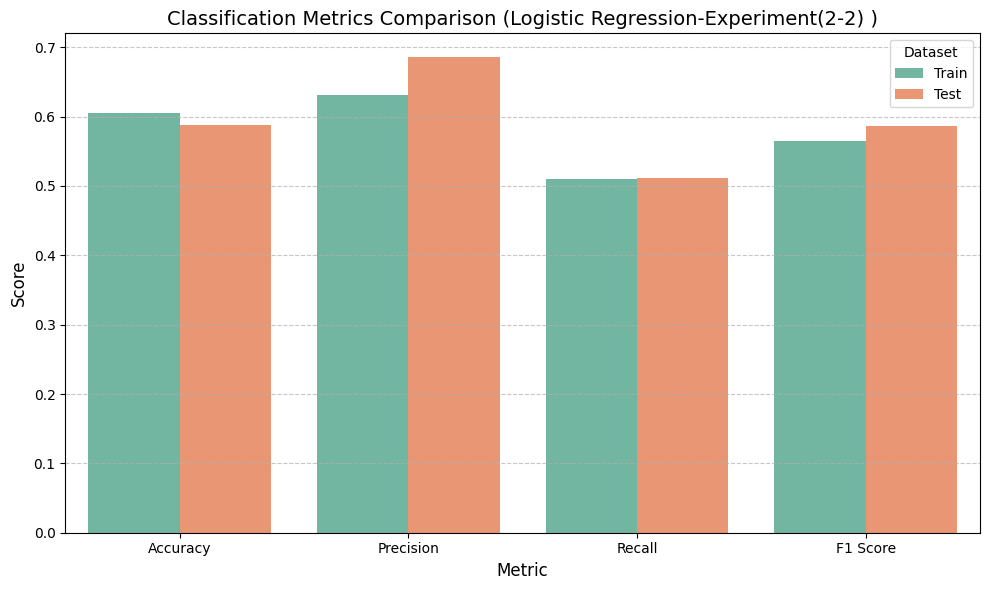
**-Testing metrics:**

Accuracy: 0.5884

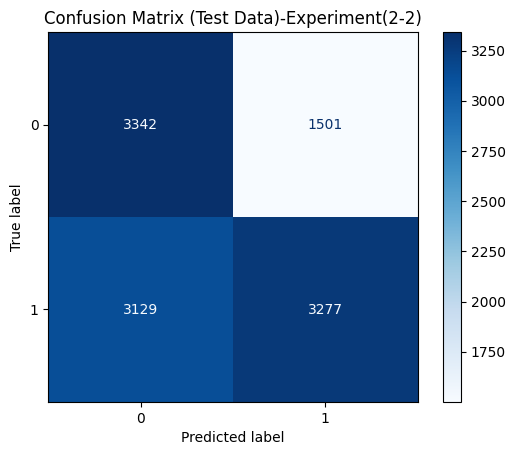
Precision: 0.6859

Recall: 0.5116

F1 Score: 0.5860



This bar chart compares the performance metrics — Accuracy, Precision, Recall, and F1 Score — for the Logistic Regression model on both training and test datasets.



This confusion matrix presents the classification results of a model (unspecified here) on the test dataset, showing the distribution of correct and incorrect predictions.

**We used the default parameters for LogisticRegression.**

**2. Decision Tree Regression**

Results:

**-Training metrics:**

Accuracy: 0.9996

Precision: 1.0000

Recall: 0.9992

F1 Score: 0.9996

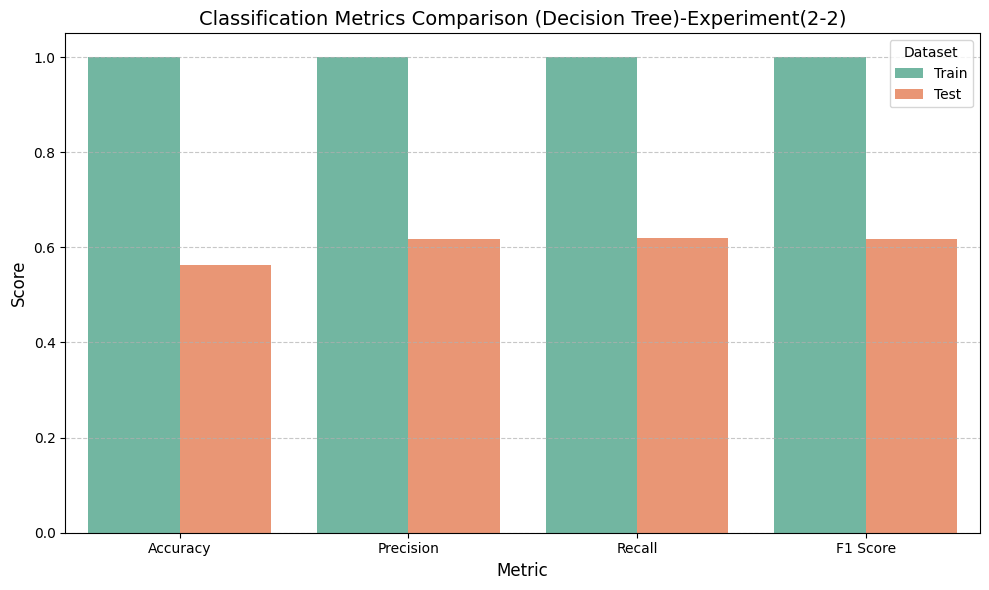
**-Testing metrics:**

Accuracy: 0.5638

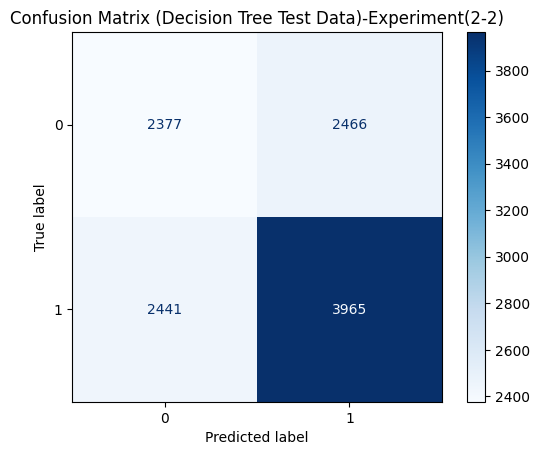
Precision: 0.6165

Recall: 0.6190

F1 Score: 0.6177



This bar chart compares the Accuracy, Precision, Recall, and F1 Score metrics for the Decision Tree model across the training and test datasets.



This confusion matrix shows the classification results of the Decision Tree model on the test dataset. It indicates the number of true positives, true negatives, false positives, and false negatives.

**We used the default parameters for Decision Tree.**

**3.** **Random Forest**

Results:

**-Training metrics:**

Accuracy: 0.9996

Precision: 0.9995

Recall: 0.9997

F1 Score: 0.9996

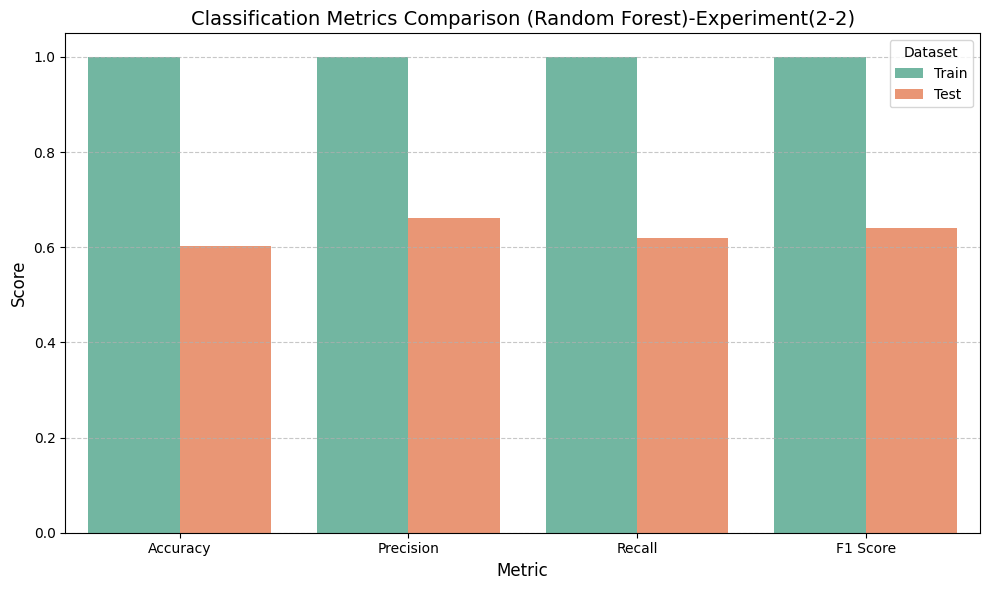
**-Testing metrics:**

Accuracy: 0.6025

Precision: 0.6613

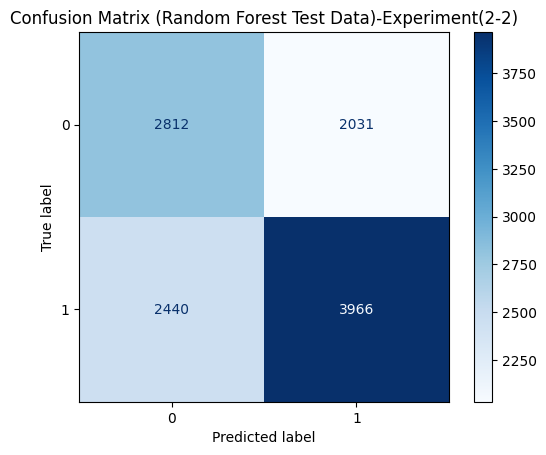
Recall: 0.6191

F1 Score: 0.6395



The model performs extremely well on the training data (all metrics close to 1.0), which indicates overfitting—the model has learned the training data too well.

The test metrics are significantly lower (around 0.6–0.65), showing that the model does not generalize well to unseen data.



There is a high number of misclassifications (2,440 false negatives and 2,031 false positives).

While the correct predictions are decent (2,812 true negatives and 3,966 true positives), the overall confusion is relatively high, especially between the two classes.

**We used the default parameters for Random Forest.**

**4.** **Gradient Boosting**

Results:

**-Training metrics:**

Accuracy: 0.6694

Precision: 0.7682

Recall: 0.4853

F1 Score: 0.5948

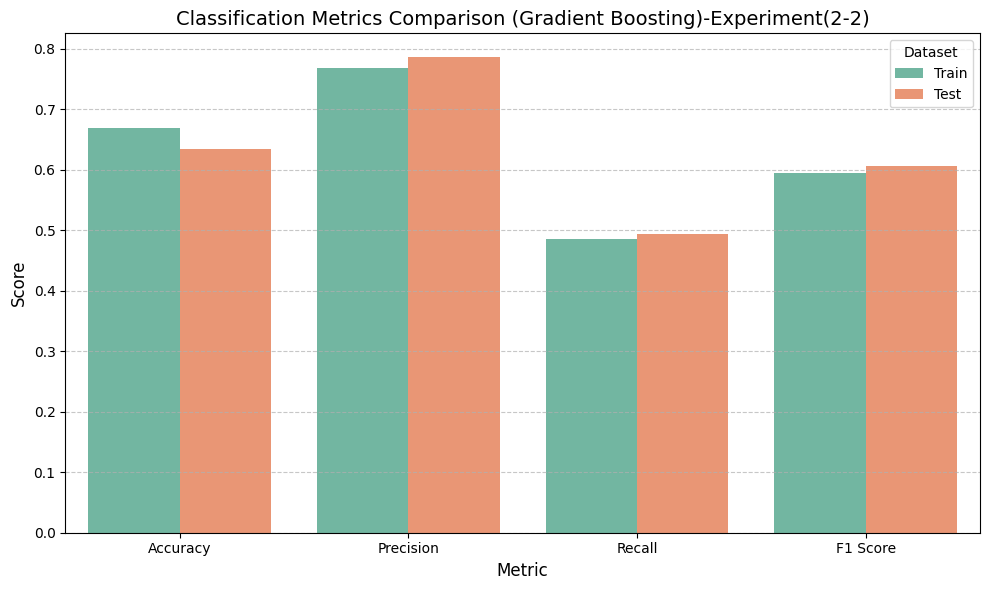
**-Testing metrics:**

Accuracy: 0.6353

Precision: 0.7865

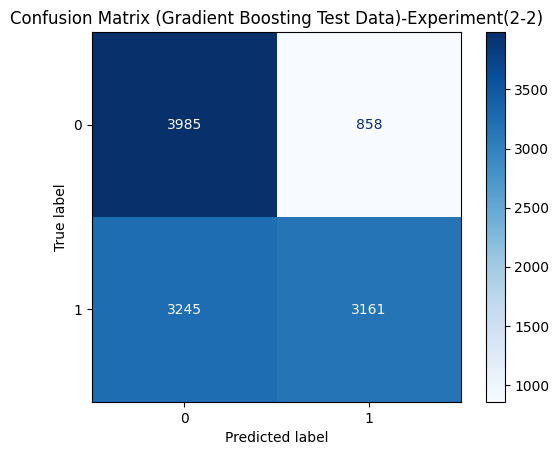
Recall: 0.4934

F1 Score: 0.6064



The training and test scores are relatively close, indicating a more stable model with less overfitting.

Precision is higher than recall, which suggests that the model makes fewer false positives but may miss some actual positives.



This model shows better performance with fewer misclassifications (3,245 false negatives and 858 false positives).

The number of correct predictions (3,985 true negatives and 3,161 true positives) is higher, suggesting better class separation and more reliable predictions.

**We used the default parameters for Gradient Boosting.**

**5.XGBOOST**

Results:

**-Training metrics:**

Accuracy: 0.7287

Precision: 0.7749

Recall: 0.6445

F1 Score: 0.7037

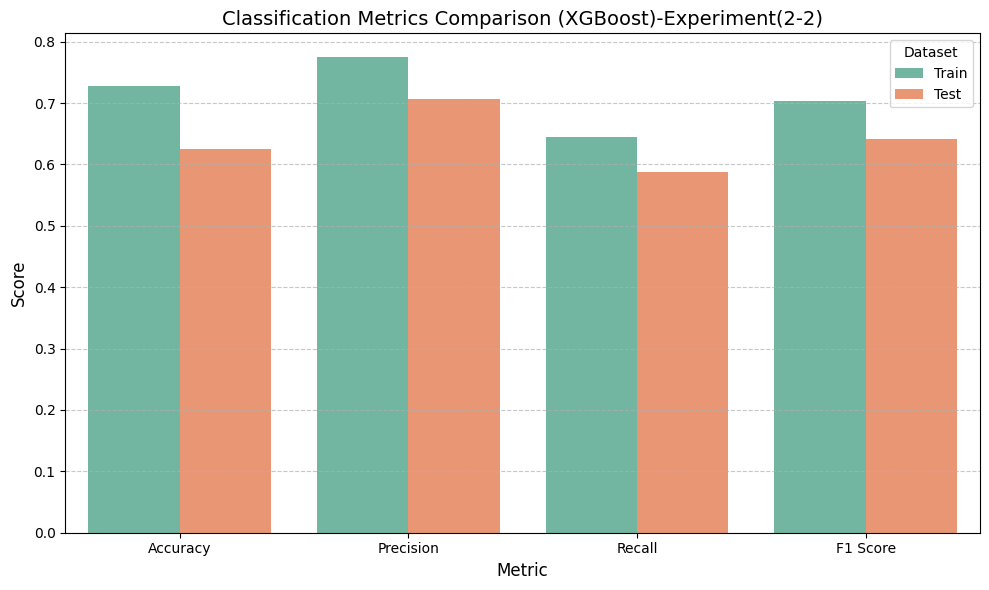
**-Testing metrics:**

Accuracy: 0.6259

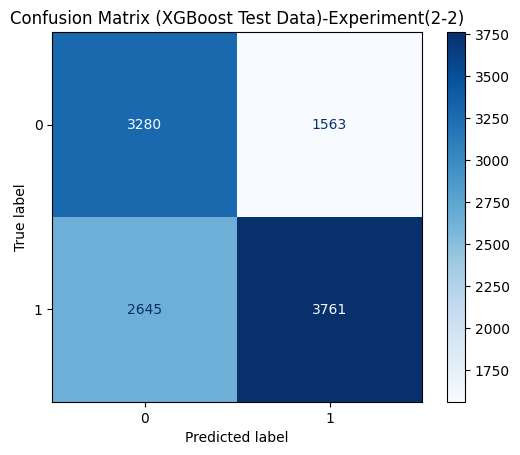
Precision: 0.7064

Recall: 0.5871

F1 Score: 0.6413



A solid performing model with a good trade-off between precision and recall.



The XGBoost model has better balance with fewer false negatives than LightGBM, but still a moderate number of false positives.

**We used the default parameters for XGBOOST.**

**6.** **LightGBM**

Results:

**-Training metrics:**

Accuracy: 0.6855

Precision: 0.7681

Recall: 0.5314

F1 Score: 0.6282

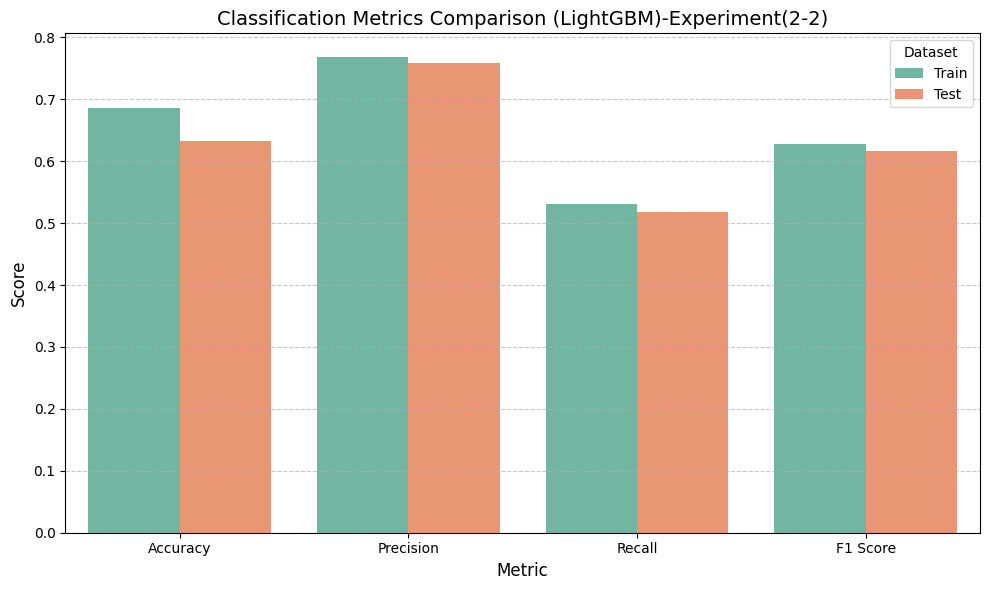
**-Testing metrics:**

Accuracy: 0.6321

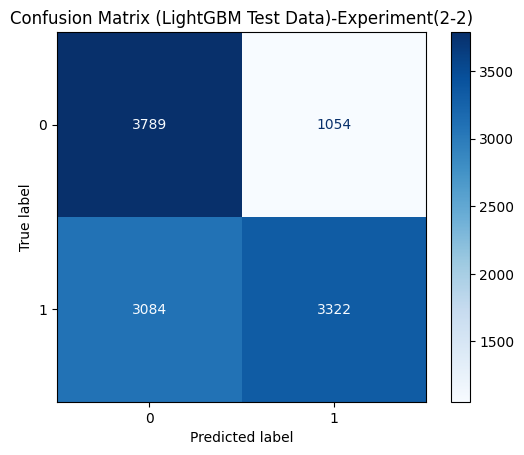
Precision: 0.7591

Recall: 0.5186

F1 Score: 0.6162



Balanced performance, with an emphasis on precision over recall.



The LightGBM model shows moderate performance with a significant number of false negatives, which may indicate it's missing a good portion of positive cases.

**We used the default parameters for LIGHTGBM.**

**7.** **CatBoost**

Results:

**-Training metrics:**

Accuracy: 0.7132

Precision: 0.7777

Recall: 0.5971

F1 Score: 0.6755

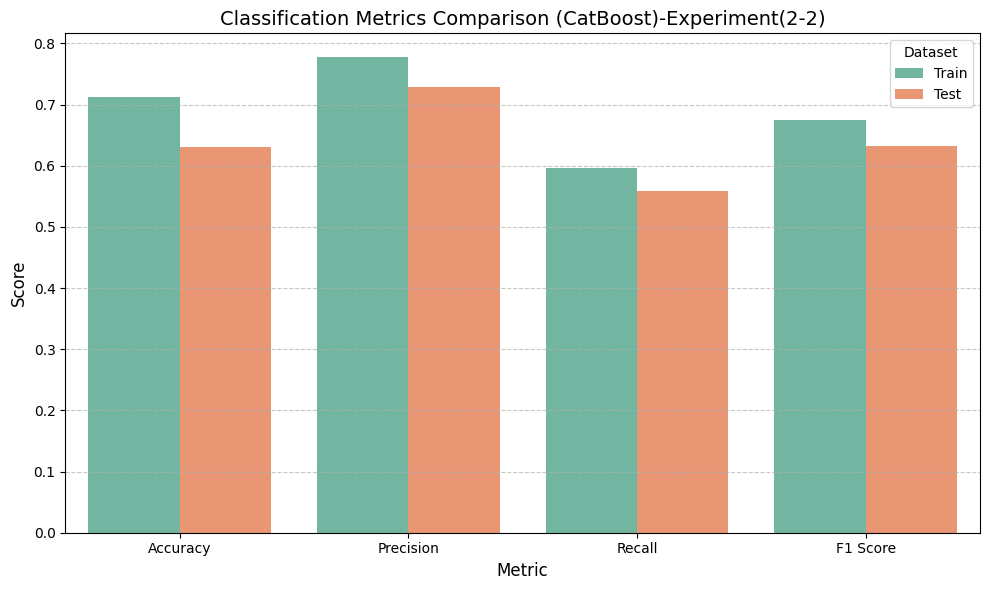
**-Testing metrics:**

Accuracy: 0.6306

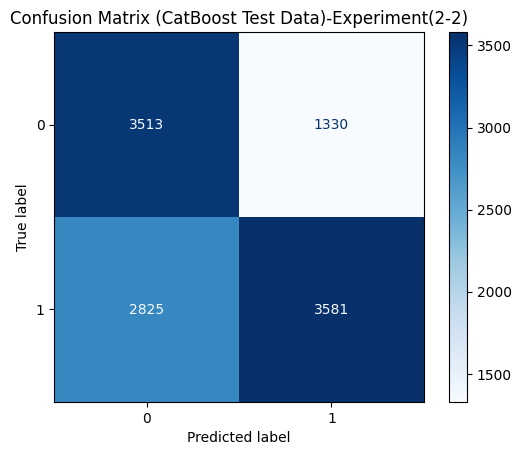
Precision: 0.7292

Recall: 0.5590

F1 Score: 0.6329



The model demonstrates good generalization and balanced performance.



CatBoost maintains a relatively balanced classification with fewer false predictions.

**We used the default parameters for CATBOOST.**

**8.** **K-Nearest Neighbors**

Results:

**-Training metrics:**

Accuracy: 0.7655

Precision: 0.8061

Recall: 0.6992

F1 Score: 0.7489

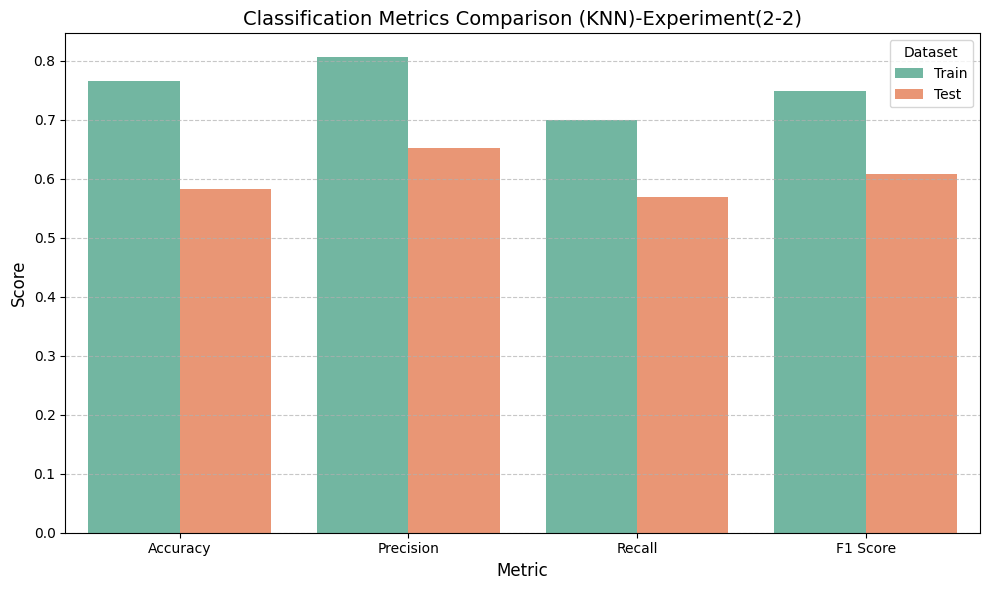
**-Testing metrics:**

Accuracy: 0.5817

Precision: 0.6524

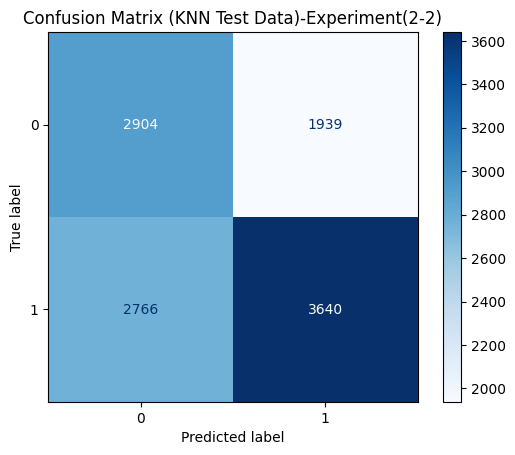
Recall: 0.5682

F1 Score: 0.6074



Training metrics are noticeably higher than testing metrics, especially for recall and accuracy.

Indicates potential overfitting: the model performs better on training data than on unseen test data.



The KNN model shows decent classification but struggles with a high number of false positives and false negatives, reducing reliability.

**We used the default parameters for KNN.**

**9-Naive Bayes**

Results:

**-Training metrics:**

Accuracy: 0.6281

Precision: 0.7044

Recall: 0.4415

F1 Score: 0.5428

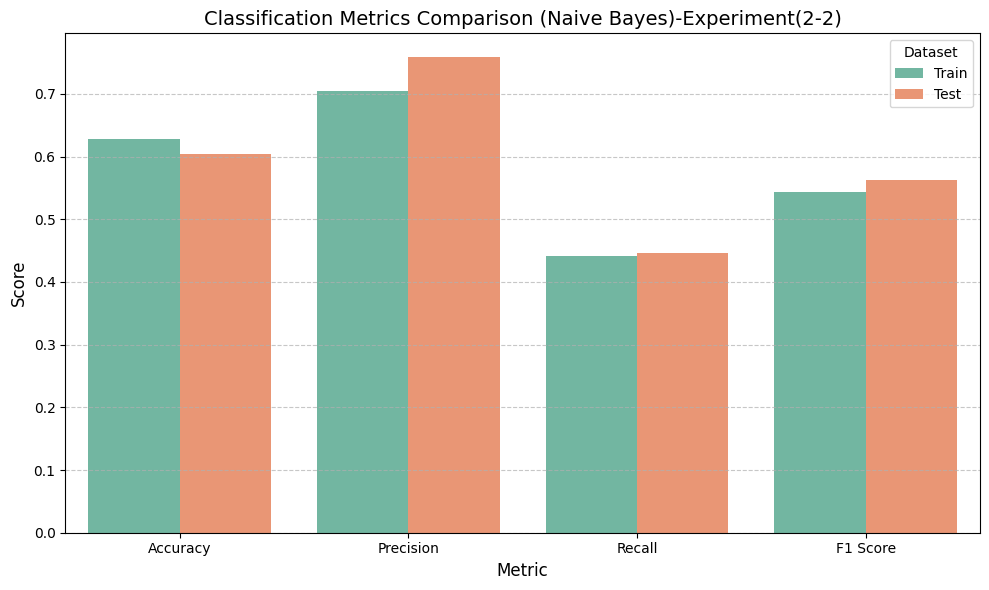
**-Testing metrics:**

Accuracy: 0.6038

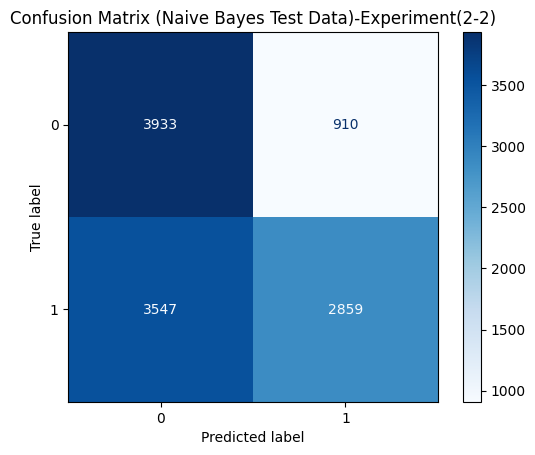
Precision: 0.7586

Recall: 0.4463

F1 Score: 0.5620



The Naive Bayes model demonstrates moderate precision (around 0.7) and recall (around 0.45), with consistent performance between training and testing sets, indicating stable generalization but limited sensitivity to positive cases.



Naive Bayes correctly identified 3,933 true negatives and 2,859 true positives, with 910 false positives and 3,547 false negatives. This shows relatively balanced but slightly weaker performance in classifying both classes.

**We used the default parameters for NAÏVE BAYES**.

**10.** **Support Vector Machine**

Results:

**-Training metrics:**

Accuracy: 0.6375

Precision: 0.7376

Recall: 0.4268

F1 Score: 0.5407

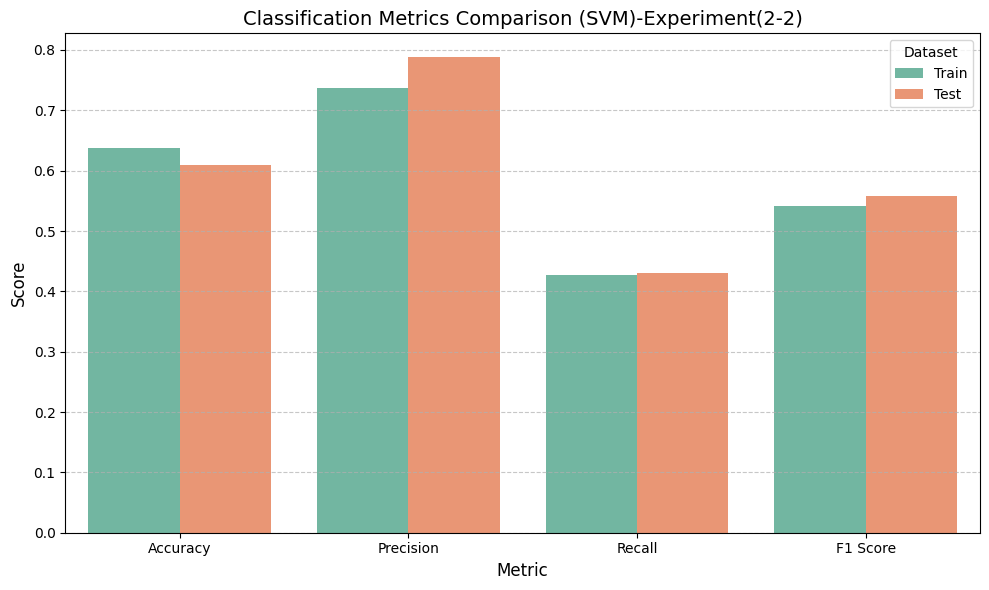
**-Testing metrics:**

Accuracy: 0.6100

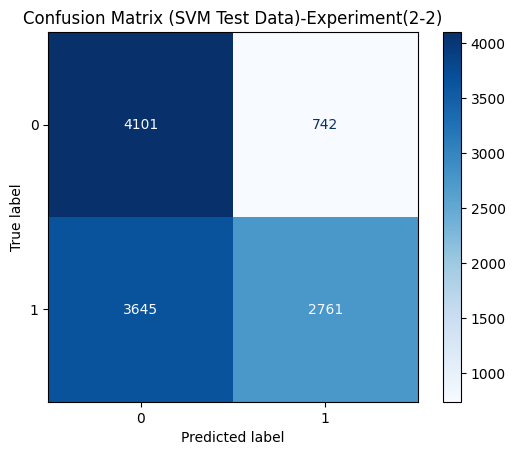
Precision: 0.7882

Recall: 0.4310

F1 Score: 0.5573



The SVM model shows high precision (close to 0.8) for both training and testing sets. However, recall is notably lower (around 0.4), suggesting that while it predicts positive cases accurately when it does so, it misses many actual positives.



The model classified 4,101 true negatives and 2,761 true positives, while misclassifying 742 as false positives and 3,645 as false negatives. This indicates the model is better at identifying class 0 compared to class 1.

**We used the default parameters for SVC.**

**Model Selection Justification**

During the evaluation phase, several models demonstrated strong performance, including LightGBM and Gradient Boosting Classifier. However, XGBoost emerged as the most effective model, outperforming all others by a clear margin in terms of F1-score, particularly on the test set.

After extensive experimentation and comparison, XGBoost was selected as the final model for deployment based on the following considerations:

**1. Highest Overall Performance**

XGBoost consistently delivered the highest F1-scores across all runs, showing superior ability to generalize on unseen data. Its performance was especially strong in predicting the minority class, which is essential for binary classification tasks involving sensitive topics like drug use prediction.

**2. Best Results with Full Feature Set**

While feature selection techniques were tested to reduce dimensionality, the results clearly showed that performance degraded when features were removed. XGBoost achieved its best results using the complete set of features, leveraging its ability to capture complex nonlinear interactions effectively.

**3. Advanced Regularization and Overfitting Control**

XGBoost includes built-in L1 and L2 regularization, helping prevent overfitting even when working with many features. This made it more stable and robust compared to other models.

**4. Efficiency and Scalability**

XGBoost is highly optimized for both speed and memory efficiency, enabling fast training and evaluation—even with large feature spaces.

**5. Robustness to Hyperparameters**

Despite its complexity, XGBoost proved relatively robust to hyperparameter tuning, maintaining strong performance across various configurations. This reduced the risk of model instability and saved time during optimization.

**6. Interpretability and Feature Importance Insights**

With built-in tools for visualizing feature importance, XGBoost also supports model interpretability, making it suitable for analytical applications where transparency matters.

**Conclusion:**

Given its superior performance with all features, robustness, and practical advantages, XGBoost was chosen as the final model prior to any feature selection steps, ensuring the best possible predictive accuracy for this project.

**Hyperparameter Tuning and Final Model Training**

Initially, we attempted to fine-tune the Gradient Boosting Classifier using GridSearchCV in order to find the optimal hyperparameters. However, due to the large number of parameter combinations and the computational complexity involved, the process was estimated to take over 9 hours, making it impractical within the project's constraints.

As an alternative, we applied RandomizedSearchCV to explore a broader space of hyperparameter combinations in a more time-efficient manner. Despite conducting multiple randomized trials, the tuned models consistently underperformed compared to the default configuration, especially on the validation and test sets.

Given these findings, we ultimately adopted the default hyperparameters, which provided more stable and reliable results without the risk of overfitting or excessive training time.

**Evaluation and Comparative Analysis**

To further assess the model's real-world applicability, it was tested on a small set of real-world data samples. Due to the limited availability of actual data, additional synthetic samples were generated following the same statistical distribution to enhance the evaluation.

Despite the data quality challenges—including class imbalance and the presence of noisy or limited samples—the model exhibited promising performance:

Class Precision Recall F1-Score Support

0 0.51 0.82 0.63 99

1 0.56 0.23 0.32 101

Accuracy: 0.52 Macro Avg F1: 0.48 Weighted Avg F1: 0.47

**Interpretation of the Results**

**-Strong Detection of Class 0:** The model achieved high recall (82%) for class 0, meaning it was able to correctly identify the majority of negative cases.

**- Balanced Precision Across Classes:** Although the dataset was imbalanced, the model maintained similar precision for both classes (~0.51–0.56), reflecting a consistent decision boundary.

**- Challenging Class 1 Recall:** As expected in imbalanced datasets, recall for the minority class (class 1) was lower. However, this is a common and anticipated trade-off in real-world applications when data quality and representation are limited.

Despite applying techniques like SMOTE during training, the real-world test data remained imperfect. Yet, the model still managed to differentiate between the two classes, indicating a strong ability to generalize.

**Conclusion**

Although the real-world test set was small , and the data quality was suboptimal, the model delivered robust performance and clear discrimination between classes. These results confirm that the model is practically useful and capable of handling imperfect, real-world scenarios — a critical factor in applied machine learning, especially in sensitive domains like substance use prediction.

**Web Interface / Deployment**

To make our machine learning model accessible and user-friendly, we developed an interactive web application using Streamlit.

**Technologies Used:**

* Python
* Streamlit (for web interface)
* Scikit-learn (for model training and prediction)
* Ngrok (for public URL access)
* Joblib (for saving/loading the model and pipeline)

**Deployment Process:**

1. Model & Pipeline Loading:
   * The trained GradientBoostingClassifier and the preprocessing pipeline were saved using joblib.
   * These files were loaded within the Streamlit app to make predictions on new user inputs.
2. Interface Features:
   * A login page to simulate user authentication.
   * Interactive sliders and drop-down menus to collect user inputs (age group, personality traits).
   * Real-time prediction using the model, with customized feedback messages depending on the result.
   * Feature importance table and visualization using seaborn and matplotlib.
3. Styling & Layout:
   * Custom CSS was embedded in the app for a professional and user-friendly UI.
   * Sidebar includes app information, developer names, and contact email.
4. Public Access:
   * The app is launched locally using streamlit run.
   * Ngrok is used to generate a public URL for external access during testing and demonstration.

**Web App Access**

You can access the deployed web app via the following public URL (generated by ngrok): [Click here](👉%20https:/your-ngrok-url.ngrok.io)

**Discussion**

In this study, we developed and evaluated various machine learning models to predict the likelihood of narcotics experimentation based on demographic and psychological factors. The results offer valuable insights:

**Model Performance**

The XGBOOST Classifier consistently outperformed other models (e.g., Logistic Regression, Random Forest, Naive Bayes, SVM, etc.) in terms of F1-score on both training and testing datasets.

Tree-based models like CatBoost, Gragdient Boosting, and LightGBM also showed strong performance, benefiting from their ability to handle non-linearities and categorical variables effectively.

Simpler models like Naive Bayes and KNN struggled with performance, likely due to assumptions of feature independence and sensitivity to feature scaling, respectively.

**Interpretation of Features**

Age was by far the most influential predictor in the final model, aligning with public health data that links age groups with substance experimentation risk.

Personality traits such as conscientiousness, neuroticism, and openness showed varying degrees of impact, confirming psychological literature on behavioral tendencies and drug use.

**Strengths**

Use of a real, large-scale dataset (NSDUH) with thousands of records enhanced the model’s generalizability.

Comprehensive preprocessing pipeline with feature selection, outlier handling, and SMOTE improved data quality and model robustness.

Deployment via Streamlit made the model accessible and interpretable for non-technical users.

**Limitations**

Some psychological attributes were self-reported, which may introduce bias or inaccuracies.

The dataset was imbalanced initially, and although SMOTE was used, synthetic sampling might not capture the true distribution of minority classes.

**Implications**

The system developed in this project could be used as a pre-screening tool in educational or clinical settings to identify at-risk individuals early, allowing for timely intervention and awareness programs. However, predictions should not replace clinical judgment and should be used in conjunction with expert evaluation.

**Future Work**

-Incorporate more behavioral and environmental features for a more holistic model.

--Improve interpretability using tools like SHAP or LIME.

-Deploy the system on a cloud platform and expand it to support multi-language input and reports.

**Conclusion**

In this project, we developed IntelliShield, a predictive system that leverages machine learning techniques to assess the likelihood of narcotics experimentation based on psychological and demographic attributes. After preprocessing the NSDUH dataset, applying feature engineering, and evaluating multiple classification algorithms, the XGBOOST Classifier was selected as the best-performing model based on F1 score.

What makes IntelliShield particularly promising is its strong performance on real-world data, which suggests its potential for practical adoption in real-life scenarios. The model generalized well and demonstrated reliable predictions, reinforcing its suitability for real-time risk assessment.

The system was deployed using a user-friendly Streamlit web interface, allowing users to input psychological traits and receive personalized risk assessments and prevention tips. This project highlights the power of AI-driven tools in supporting early intervention strategies and promoting public health.

**References:**

[1]:[Machine Learning-Based Prediction of Substance Use in Adolescents in Three Independent Worldwide Cohorts: Algorithm Development and Validation Study](https://pubmed.ncbi.nlm.nih.gov/39993291/)

[2]:[Machine-learning approaches to substance-abuse research: emerging trends and their implications](https://pubmed.ncbi.nlm.nih.gov/32304429/)

[3]:[Substance use prediction using artifcial intelligence](https://link.springer.com/article/10.1007/s42001-024-00356-6)

[4]:[Prediction of addiction to drugs and alcohol using machine](https://d1wqtxts1xzle7.cloudfront.net/86650401/15102-libre.pdf?1653830795=&response-content-disposition=inline%3B+filename%3DPrediction_of_addiction_to_drugs_and_alc.pdf&Expires=1741788100&Signature=fHkWJqf3HLlZ~2yehSTd1kSOKapwnGkXEXmHvfIGtY1Myjqyp1E4GE-~C3YEgsVPEwMZ0vtFde-10ddNAo-QbJkpMtExfpNgjRbPWrgyedfk8qfDlf5bGDdtX62UayDZX6KyS~kMAEhfIRuDPEk6OerFNXiOMuQf~4bSqLwwNfHXf-TtZecgEUNAlS2-Ve7jIU-NHlnNRS~qPviXzYDtlTWIdsGmQLIH2J7nuQYknWnlQoGLgP0M~6Zharq7vag0ZxFKITOflLWbMezJuWHILHGvgjVHlZNIfyT~bWFlP3gzWPIsVA-F6viMKwvicgXuRnlLCypuAdsW-5eYydaXNA__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)

[5]:[Analysis of substance use and its outcomes by machine learning](https://www.sciencedirect.com/science/article/abs/pii/S0376871619303813)

[6]:[Machine-learning approaches to substance-abuse research: emerging trends and their implications](https://journals.lww.com/co-psychiatry/abstract/2020/07000/machine_learning_approaches_to_substance_abuse.8.aspx)

7: National Survey on Drug Use and Health (NSDUH) – The primary dataset used in this study, provided by the Substance Abuse and Mental Health Services Administration (SAMHSA).  
Available at: <https://www.datafiles.samhsa.gov/>

8.Scikit-learn Documentation – Official documentation for the machine learning library used to build and evaluate models.

9.Streamlit Documentation – Official documentation for developing interactive web apps using Streamlit.  
Available at[: https://docs.streamlit.io/](:%20https:/docs.streamlit.io/)

**Appendices**

Appendix A: Project Source Code

This includes all stages of the project, from preprocessing to modeling and evaluation:

[Data Analysis/Processing/Model Training](https://colab.research.google.com/drive/1YDGqcQLGwXX2dOyALNapx8ZcgdrCsxWk%23scrollTo=pEjf8UMqsC-R)

[Streamlit Code](https://colab.research.google.com/drive/1xPB8oxH8EtCT9xdNod3byifnPwGwSq8L%23scrollTo=ZS_QqsnurvjG)

[app.py](https://a22c-34-73-11-244.ngrok-free.app/): Streamlit application for real-time prediction and user interaction.

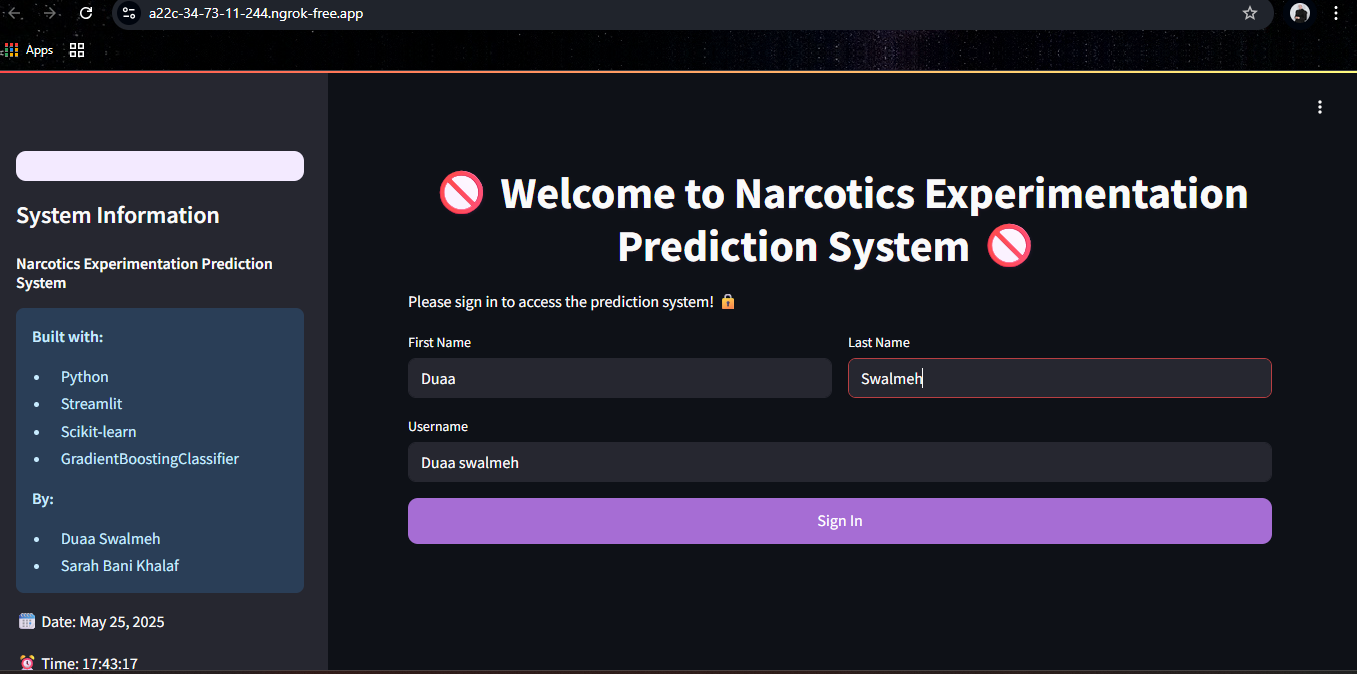
Appendix B: Web Application Interface

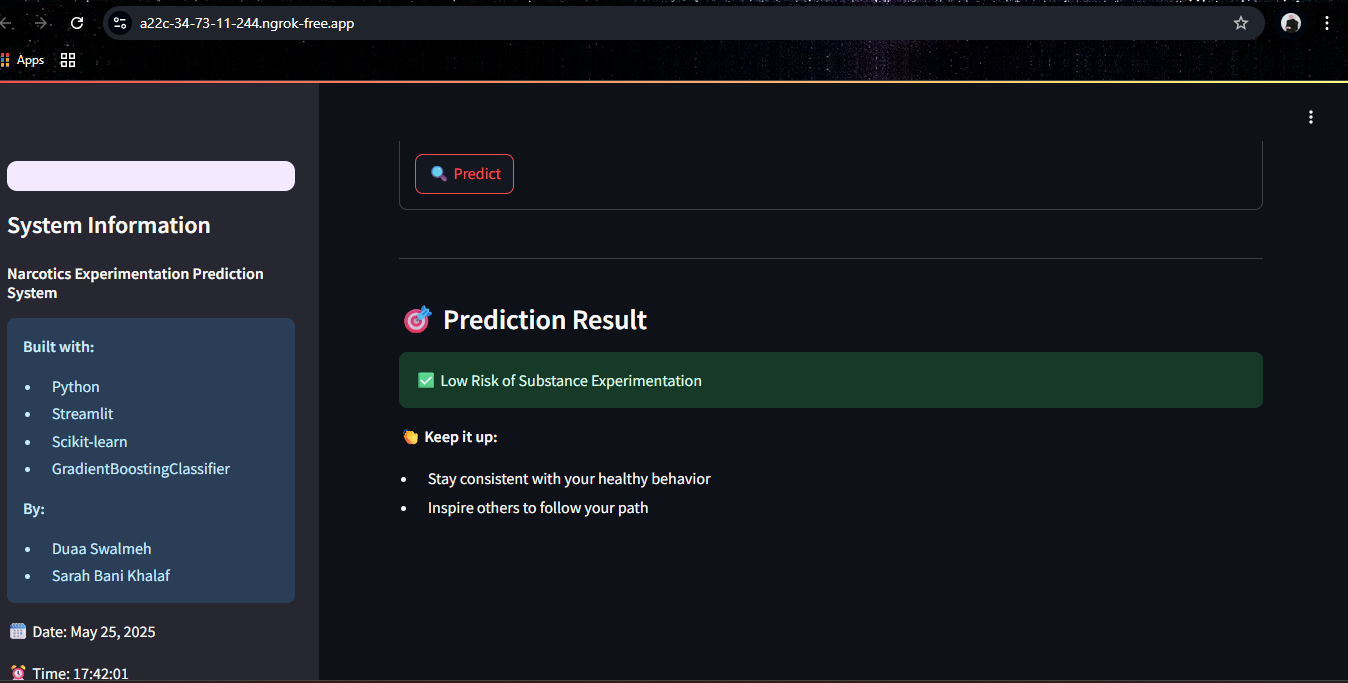
A web interface was developed using Streamlit:

Allows users to input age and personality traits.

Provides real-time prediction of drug experimentation risk.

Deployed temporarily using ngrok for accessibility.

****

****