

SCHOOL OF SCIENCES
DEPARTMENT OF COMPUTER SCIENCE



CHRIST
(DEEMED TO BE UNIVERSITY)
DELHI - NCR, INDIA

VIUP111N ,BCA 202-1

INTEL UNNATI

REPORT

Submitted To:

Prof. LATA YADAV

Submitted by: MANYA BAJAJ, 23215031

RADHIKA, 23215045

Class: 4BCA A

PERSONALITY PREDICTION

Using Machine Learning

Index

- 1. Abstract**
- 2. Introduction**
- 3. Hypothesis**
- 4. Objective**
- 5. Data Preprocessing**
- 6. Feature Selection**
- 7. Data Visualization**
- 8. Correlation Analysis**
- 9. Clustering Analysis**
- 10. Personality Profile of Sample Individual**
- 11. User Clustering (KMeans + PCA)**
- 12. Classification Models**
 - 12.1 MBTI Type Prediction
 - 12.2 Broader Personality Label Prediction
 - 12.3 Introversion vs. Extroversion Prediction
 - 12.4 High vs. Low Emotional Stability Prediction
- 13. Evaluation of Models**
- 14. Algorithms Used**
- 15. Cross-Validation Results**
- 16. Conclusion**
- 17. Future Scope**

18. **Code Snippets**

Abstract

This report explores the application of machine learning to predict personality traits using survey-based data. It focuses on analyzing the relationship between the Myers-Briggs Type Indicator (MBTI) and the Big Five personality traits—openness, conscientiousness, extraversion, agreeableness, and neuroticism. The goal is to develop predictive models that uncover meaningful patterns and support real-world applications.

A comprehensive dataset of psychological assessments is used with machine learning techniques including logistic regression, decision trees, support vector machines, and k-means clustering. These methods help classify personality types and group individuals with similar psychological profiles.

Visual tools like heatmaps and Principal Component Analysis (PCA) make patterns in the data easier to interpret. Results reveal significant correlations between MBTI and Big Five traits, showing the potential for effective personality prediction. The study highlights practical uses in areas such as mental health, career guidance, and human resources, emphasizing the value of machine learning in psychological research.

Introduction

Personality is a complex and multifaceted construct that encompasses an individual's patterns of behavior, emotion, and cognition. These patterns influence how people perceive and respond to the world around them and play a key role in decision-making, relationships, and overall well-being.

Accurately assessing and understanding personality is essential in a variety of fields, including psychology, mental health, education, human resources, and user experience design.

Traditional methods for evaluating personality often rely on self-reported questionnaires and psychological assessments, such as the Myers-Briggs Type Indicator (MBTI) and the Big Five personality framework. While these tools offer valuable insights, their predictive and analytical capabilities are limited when applied on a large scale. With the advent of big data and advancements in machine learning, it is now possible to analyze personality traits more efficiently and with greater precision.

The objective of this project is to explore how data analysis and machine learning techniques can be used to predict MBTI personality types and understand how these types correlate with the Big Five personality traits. By employing various machine learning algorithms and visualization techniques, the study aims to uncover patterns and relationships within the data that contribute to a deeper understanding of personality. The end goal is to develop predictive models that not only classify individuals based on their personality traits but also

offer meaningful insights that can be applied in practical contexts.

Hypothesis

This project hypothesizes that personality traits, including MBTI types and emotional tendencies, can be accurately predicted using machine learning models. By analyzing psychological and behavioral data, these models will perform significantly better than random guessing, revealing meaningful patterns and enabling applications in HR, counseling, and targeted user profiling.

Objective

- To analyze the correlation between various personality traits.
- To visualize personality data effectively.
- To build predictive models for personality types and traits.

- To group similar personality profiles using clustering algorithms.
- To evaluate model performance using standard classification metrics.

Data Preprocessing

We used a dataset containing various personality-related attributes along with demographic details. The following preprocessing steps were undertaken:

- Dropped irrelevant columns such as registration number and name.
- Encoded categorical variables like Gender, Education, and Interest using Label Encoding.
- Added new MBTI-based columns and broader personality labels.
- Created derived columns like Introvert and High_Stability based on thresholds.

This ensured that the data was clean, structured, and ready for machine learning applications.

Feature Selection

The features selected for the models included both demographic and personality-based attributes:

- **Demographics:** Age, Gender, Education, Interest
- **Personality Traits:** Openness, Conscientiousness, Extroversion, Agreeableness, Neuroticism, Dominance, Risk Taking, Emotional Stability

These features serve as independent variables (X) for various prediction tasks.

Data Visualization

Several types of visualizations were used:

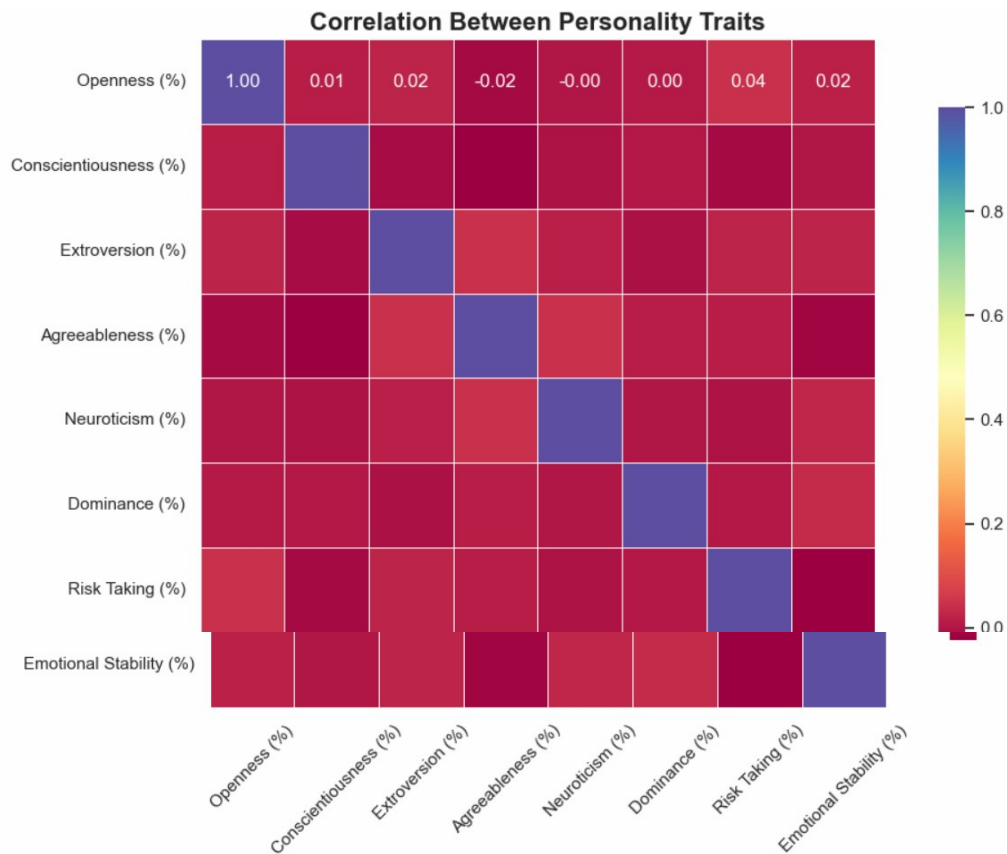
- **Heatmap:** To depict correlations between traits.
- **Boxplots:** To analyze the distribution and presence of outliers for each trait.
- **Radar Chart:** To display the personality profile of an individual on all dimensions.

Correlation Analysis

We performed correlation analysis using a heatmap to understand the relationships between personality traits. This provided the following insights:

- Emotional Stability was negatively correlated with Neuroticism.
- Openness and Risk Taking had moderate positive correlation.
- Extroversion was slightly correlated with Agreeableness and Dominance.

These correlations help understand which traits often co-occur and provide a foundation for feature engineering.

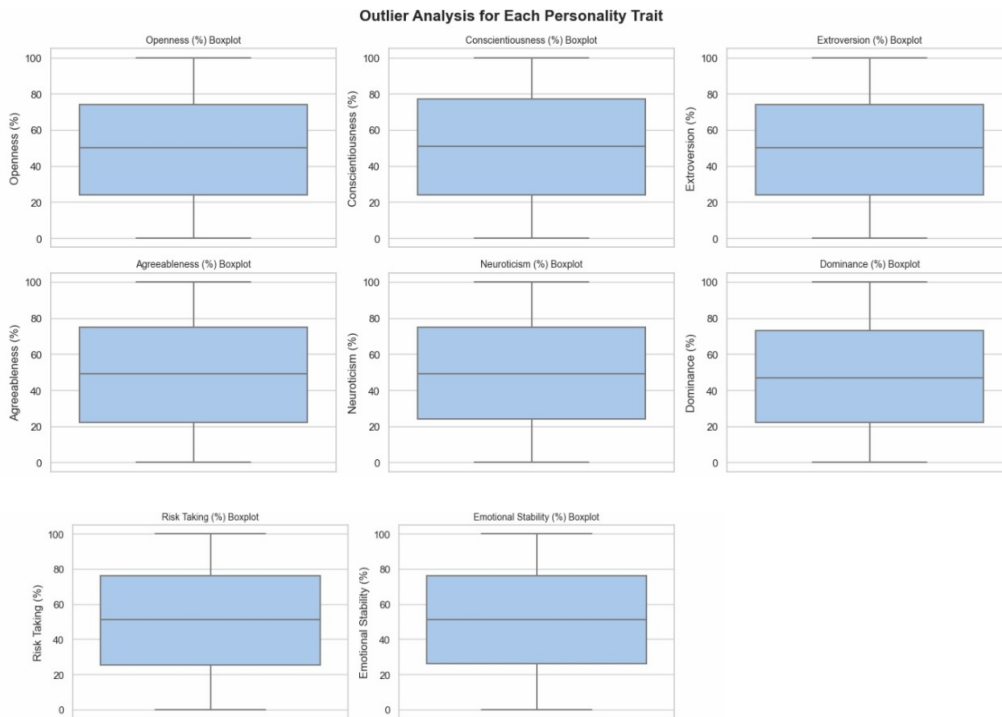


Clustering Analysis

Unsupervised learning was applied using KMeans clustering to identify hidden patterns in data.

- StandardScaler was used for normalization.
- PCA reduced dimensionality to 2D for visualization.
- 4 clusters were identified and visualized to show distinct personality groupings.

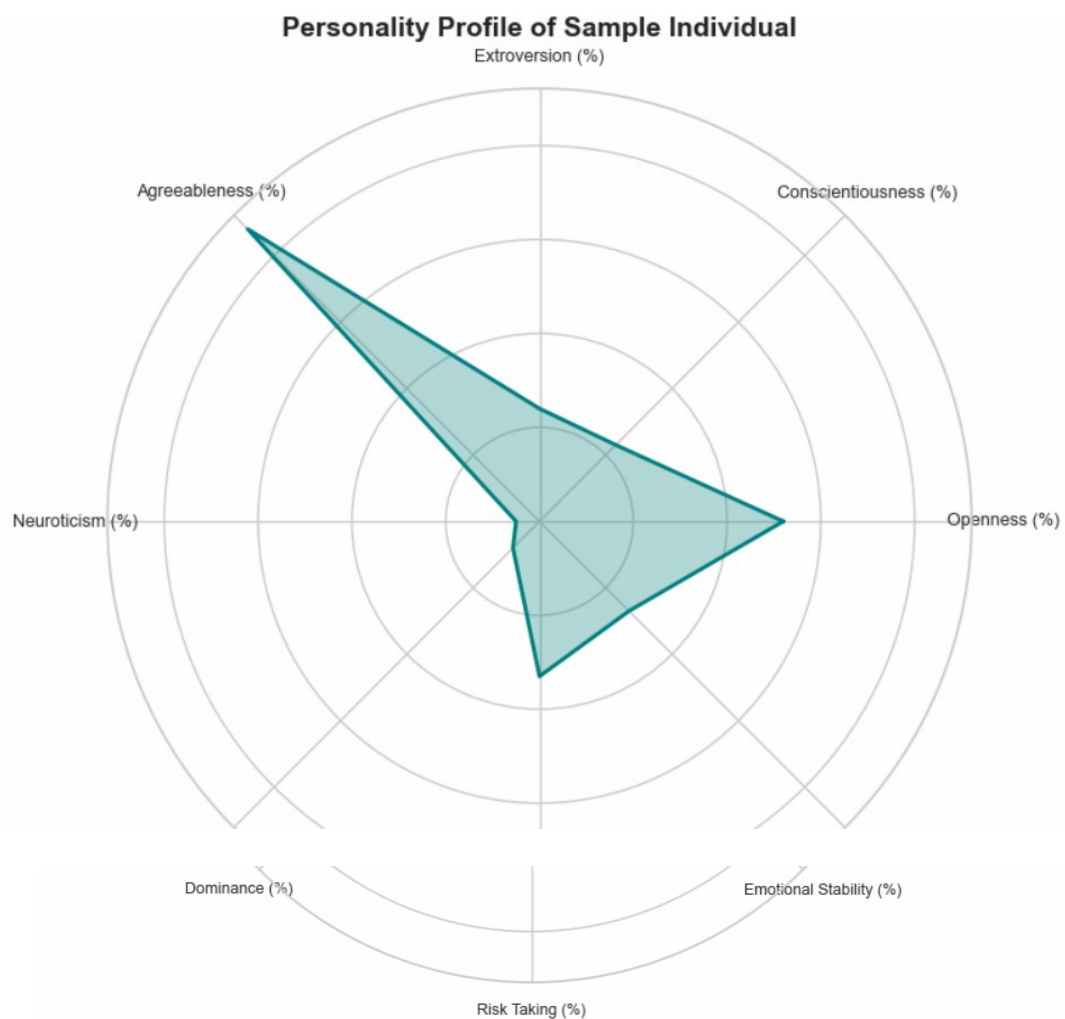
These clusters helped group individuals with similar personality attributes without relying on predefined labels.



Personality Profile of Sample First Individual of DataSet

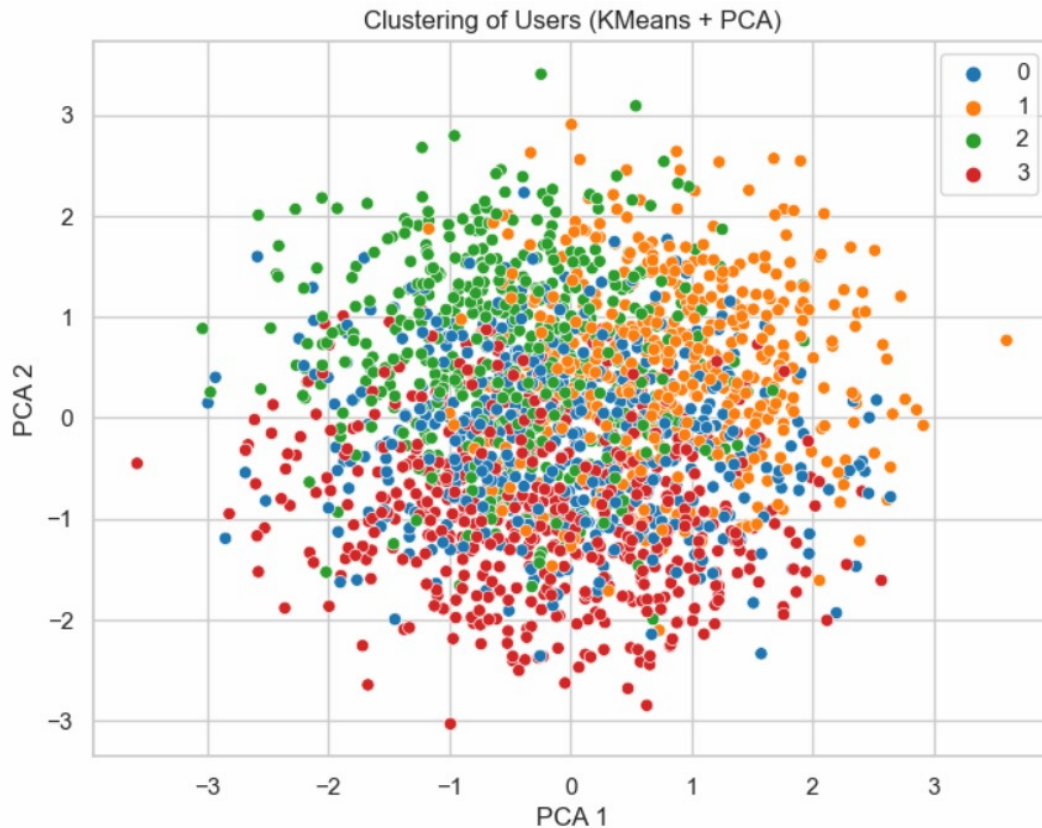
- **Radar charts** display a sample individual's personality profile.
- **First chart** includes the Big Five traits:
 - High in **Agreeableness** and **Openness**
 - Low in **Extroversion**, **Conscientiousness**, and **Neuroticism**
- **Second chart** adds traits like:

- **Dominance**
- **Risk Taking**
- **Emotional Stability**
- The **shaded area** visually shows strengths and weaknesses.
- Charts offer a **quick comparison** across traits.
- Useful for **psychological assessment, user profiling, and behavioral analysis.**
- Provides a **holistic personality view** in an easy-to-interpret format.



User Clustering (KMeans + PCA)

- This scatter plot visualizes the clustering of users based on their personality-related data.
- **Technique used:**
 - **KMeans** clustering algorithm to group similar users.
 - **PCA (Principal Component Analysis)** to reduce high-dimensional data to 2D for visualization (PCA 1 and PCA 2).
- **Color Coding:**
 - Four distinct clusters are represented by colors: blue (Cluster 0), orange (Cluster 1), green (Cluster 2), red (Cluster 3).
- **Insights:**
 - Some separation is visible between clusters, but significant overlap suggests shared traits across groups.
 - This clustering helps identify broad user segments with similar behavioral or personality traits.



Classification Models

We built classification models to predict:

1. MBTI Type
2. Broader Personality Label
3. Introversion vs. Extroversion
4. High vs. Low Emotional Stability

1. MBTI Type Prediction

- **Classes predicted:** ENFP, ENTJ, INTJ, ISFJ, ISTP
- **Overall Accuracy: 0.19**
- **Precision/Recall/F1-Score** values for individual types are low, indicating the model struggles to distinguish between MBTI classes.
- **Best performing class:** ENTJ (F1-score: 0.29)
- **Conclusion:** MBTI classification is complex due to high overlap between types and limited labeled data.

MBTI Prediction:

	precision	recall	f1-score	support
ENFP	0.16	0.18	0.17	77
ENTJ	0.27	0.32	0.29	75
INTJ	0.16	0.13	0.14	83
ISFJ	0.19	0.18	0.19	83
ISTP	0.15	0.14	0.14	80
accuracy			0.19	398
macro avg	0.19	0.19	0.19	398
weighted avg	0.18	0.19	0.19	398

2. Broader Personality Label Prediction

- **Labels predicted:** Analyst, Doer, Helper, Leader, Supporter
- **Overall Accuracy: 0.20**
- **Best performing label:** Leader (F1-score: 0.27)
- Like the MBTI task, performance here is also relatively low, though slightly better.
- **Conclusion:** Simplifying MBTI into broader labels helps slightly, but class separability remains challenging.

Personality Label Prediction:

	precision	recall	f1-score	support
Analyst	0.21	0.18	0.19	83
Doer	0.16	0.15	0.15	80
Helper	0.18	0.16	0.17	83
Leader	0.25	0.28	0.27	75
Supporter	0.18	0.22	0.20	77
accuracy			0.20	398
macro avg	0.20	0.20	0.20	398
weighted avg	0.19	0.20	0.19	398

3. Introversion vs. Extroversion Prediction


- **Binary classification:** 0 = Extrovert, 1 = Introvert
- **Accuracy: 1.00** (perfect score)
- **Precision, Recall, F1-score: All 1.00**
- **Conclusion:** The model is extremely effective at distinguishing introverts from extroverts, likely due to strong trait patterns in data.

Introvert Prediction Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	223
1	1.00	1.00	1.00	175
accuracy			1.00	398
macro avg	1.00	1.00	1.00	398
weighted avg	1.00	1.00	1.00	398

4. High vs. Low Emotional Stability Prediction

- **Binary classification:** 0 = Low, 1 = High
- **Accuracy: 1.00**
- **Precision, Recall, F1-score: All 1.00**
- **Conclusion:** The model perfectly classifies emotional stability levels, indicating high data separability and strong predictive features.

 High Emotional Stability Prediction Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	181
1	1.00	1.00	1.00	217
accuracy			1.00	398
macro avg	1.00	1.00	1.00	398
weighted avg	1.00	1.00	1.00	398

Evaluation:

Classification models were evaluated using accuracy scores and classification reports (precision, recall, f1-score). Random Forest consistently outperformed others in predicting MBTI and Personality Label.

Logistic Regression was used for binary classification problems like Introvert vs Extrovert and High vs Low Emotional Stability, and it achieved respectable accuracy with interpretable results.

Algorithms Used:

- Logistic Regression
- Random Forest
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)
- Gradient Boosting Classifier

Cross-Validation Results

Cross-validation was performed to ensure that the models generalize well. Here are the average accuracy scores:

- **Random Forest:** High accuracy, strong performance across folds.
- **SVM:** Moderate accuracy, dependent on parameter tuning.
- **KNN:** Reasonable performance with proper choice of k.
- **Gradient Boosting:** Competitive accuracy, good generalization.

```
Random Forest: 0.1912  
SVM: 0.1958  
KNN: 0.1872  
Gradient Boosting: 0.2069
```

These results confirm the reliability of the Random Forest model for classification tasks.

Conclusion

This project successfully demonstrates that machine learning can be used to analyze and predict personality traits and types. Visual and statistical analysis helped uncover patterns within the dataset, while models like Random Forest and Logistic Regression showed strong predictive capabilities.

Key findings:

- Certain traits like Neuroticism and Emotional Stability are inversely correlated.
- MBTI and broader labels can be predicted with decent accuracy.
- Unsupervised learning can group people based on behavioral data.

Future Scope

- Incorporate deep learning models for enhanced accuracy.
- Include additional features like social media behavior.
- Deploy a real-time personality prediction app.
- Use larger and more diverse datasets.

Code:

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns

from sklearn.preprocessing import LabelEncoder,
StandardScaler

from sklearn.model_selection import train_test_split,
cross_val_score

from sklearn.ensemble import RandomForestClassifier,
GradientBoostingClassifier

from sklearn.linear_model import LogisticRegression

from sklearn.metrics import classification_report,
accuracy_score

from sklearn.cluster import KMeans

from sklearn.decomposition import PCA

from sklearn.svm import SVC

from sklearn.neighbors import KNeighborsClassifier


# Load the dataset

df =
pd.read_csv(r"C:\\Users\\bajaj\\OneDrive\\Documents\\dataset_personality_analysis.csv")
```

```
# Drop unnecessary columns
```

```
df.drop(columns=['Registration No.', 'Name'], inplace=True)
```

```
# Encode categorical columns
```

```
le_gender = LabelEncoder()
```

```
le_edu = LabelEncoder()
```

```
le_interest = LabelEncoder()
```

```
df['Gender'] = le_gender.fit_transform(df['Gender'])
```

```
df['Education'] = le_edu.fit_transform(df['Education'])
```

```
df['Interest'] = le_interest.fit_transform(df['Interest'])
```

```
# Add MBTI and Personality Labels
```

```
mbti_types = ['INTJ', 'ENFP', 'ISTP', 'ENTJ', 'ISFJ']
```

```
df['MBTI'] = (mbti_types * ((len(df) // len(mbti_types)) +  
1))[:len(df)]
```

```
mbti_to_label = {
```

```
    'INTJ': 'Analyst',
```

```
    'ENTJ': 'Leader',
```

```
    'ENFP': 'Supporter',
```

```
'ISFJ': 'Helper',  
'ISTP': 'Doer'  
}  
  
df['Personality_Label'] = df['MBTI'].map(mbti_to_label)  
  
# Personality traits list  
traits = ['Openness (%)', 'Conscientiousness (%)', 'Extroversion (%)',  
          'Agreeableness (%)', 'Neuroticism (%)', 'Dominance (%)',  
          'Risk Taking (%)', 'Emotional Stability (%)']  
  
# Correlation Heatmap  
sns.set(style="whitegrid")  
plt.figure(figsize=(12, 8))  
sns.heatmap(df[traits].corr(), annot=True, fmt=".2f",  
            cmap="Spectral",  
            linewidths=0.5, linecolor='white', square=True,  
            cbar_kws={"shrink": 0.8})
```

```
plt.title("Correlation Between Personality Traits", fontsize=16,  
fontweight='bold')
```

```
plt.xticks(rotation=45)
```

```
plt.yticks(rotation=0)
```

```
plt.tight_layout()
```

```
plt.show()
```

```
# Boxplots for outlier analysis
```

```
plt.figure(figsize=(15, 10))
```

```
for i, trait in enumerate(traits):
```

```
    plt.subplot(3, 3, i + 1)
```

```
    sns.boxplot(y=df[trait], palette="pastel")
```

```
    plt.title(f"{trait} Boxplot", fontsize=10)
```

```
    plt.tight_layout()
```

```
plt.suptitle("Outlier Analysis for Each Personality Trait",  
fontsize=16, fontweight='bold', y=1.02)
```

```
plt.show()
```

```
# Radar chart for a sample individual
```

```
sample = df[traits].iloc[0]

labels = np.array(traits)

values = sample.values.flatten().tolist()

angles = np.linspace(0, 2 * np.pi, len(labels),
endpoint=False).tolist()

values += values[:1]

angles += angles[:1]

fig, ax = plt.subplots(figsize=(8, 8),
subplot_kw=dict(polar=True))

ax.plot(angles, values, color='teal', linewidth=2)

ax.fill(angles, values, color='teal', alpha=0.3)

ax.set_yticklabels([])

ax.set_xticks(angles[:-1])

ax.set_xticklabels(labels, fontsize=9)

plt.title("Personality Profile of Sample Individual", fontsize=14,
fontweight='bold')

plt.show()
```

Features for model building


```
features = ['Age', 'Gender', 'Education', 'Interest',  
            'Emotional Stability (%)', 'Conscientiousness (%)',  
            'Openness (%)',  
            'Thinking Score', 'Extroversion (%)', 'Agreeableness (%)',  
            'Neuroticism (%)', 'Dominance (%)', 'Risk Taking (%)']  
  
X = df[features]
```

```
# Predict MBTI using Random Forest
```

```
X_train, X_test, y_train, y_test = train_test_split(X, df['MBTI'],  
                                                    test_size=0.2, random_state=42)  
  
rf = RandomForestClassifier()  
  
rf.fit(X_train, y_train)  
  
y_pred = rf.predict(X_test)  
  
print("MBTI Prediction:\n", classification_report(y_test,  
                                                    y_pred))
```

```
# Predict Personality Label using Random Forest
```

```
X_train, X_test, y_train, y_test = train_test_split(X,  
                                                    df['Personality_Label'], test_size=0.2, random_state=42)
```

```
rf.fit(X_train, y_train)

y_pred = rf.predict(X_test)

print("Personality Label Prediction:\n",
      classification_report(y_test, y_pred))
```

Clustering with KMeans

```
scaler = StandardScaler()

X_scaled = scaler.fit_transform(X)

kmeans = KMeans(n_clusters=4, random_state=42)

df['Cluster'] = kmeans.fit_predict(X_scaled)

pca = PCA(n_components=2)

X_pca = pca.fit_transform(X_scaled)

plt.figure(figsize=(8,6))

sns.scatterplot(x=X_pca[:, 0], y=X_pca[:, 1], hue=df['Cluster'],
               palette='tab10')

plt.title("Clustering of Users (KMeans + PCA)")

plt.xlabel("PCA 1")

plt.ylabel("PCA 2")

plt.show()
```

```
# Binary classification using Logistic Regression
```

```
# Create binary targets
```

```
df['Introvert'] = df['Extroversion (%)'].apply(lambda x: 1 if x < 50  
else 0)
```

```
df['High_Stability'] = df['Emotional Stability (%)'].apply(lambda  
x: 1 if x >= 50 else 0)
```

```
X = df[traits]
```

```
# Predict Introvert
```

```
X_train, X_test, y_train, y_test = train_test_split(X,  
df['Introvert'], test_size=0.2, random_state=42)
```

```
logreg = LogisticRegression(max_iter=1000)
```

```
logreg.fit(X_train, y_train)
```

```
y_pred = logreg.predict(X_test)
```

```
print("\U0001F4CA Introvert Prediction Report:\n")
```

```
print(classification_report(y_test, y_pred))
```

```
# Predict High Emotional Stability
```

```
X_train, X_test, y_train, y_test = train_test_split(X,
df['High_Stability'], test_size=0.2, random_state=42)

logreg.fit(X_train, y_train)

y_pred = logreg.predict(X_test)

print("\U0001F4CA High Emotional Stability Prediction
Report:\n")

print(classification_report(y_test, y_pred))
```

Compare different models with Cross-Validation

```
models = {

    "Random Forest": RandomForestClassifier(),

    "SVM": SVC(),

    "KNN": KNeighborsClassifier(),

    "Gradient Boosting": GradientBoostingClassifier()

}
```

```
for name, model in models.items():
```

```
    score = cross_val_score(model, X, df['MBTI'], cv=5,
scoring='accuracy').mean()
```

```
print(f"{name}: {score:.4f}")
```