A

Mini Project Report on

Mental Health Issues Prediction

Submitted

In Partial Fulfilment of the Requirements for the Award of Degree BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE ENGINEEING (ARTIFICIAL INTELLIGENCE MACHINE LEARNING)

**Submitted By**

V.Phanindra 227Z1A66C7



**SCHOOL OF ENGINEERING**

# Department of Computer Science Engineering (Artificial Intelligence Machine Learning)

**NALLA NARASIMHA REDDY**

**EDUCATION SOCIETY’S GROUP OF INSTITUTIONS**

**(Approved by AICTE, New Delhi, Affiliated to JNTU-Hyderabad) Chowdariguda (VIll) Korremula 'x' Roads, Via Narapally, Ghatkesar (Mandal) Medchal (Dist), Telangana-500088**

**2024-2025**

# ABSTRACT

# This project investigates the application of various machine learning classification algorithms for predicting depression in individuals based on a mental health dataset. The dataset was preprocessed using one-hot encoding for categorical variables, and features were standardized to enhance model performance. Five classification algorithms were evaluated: Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). The dataset was split into training and testing sets, with model evaluation based on accuracy, classification reports, and confusion matrices. Results indicated that Logistic Regression achieved the highest accuracy of 58%, while other models demonstrated varying performance levels. The study emphasizes the importance of feature selection and model tuning in improving predictive capabilities for mental health assessments, providing a foundation for future research in this critical area.

# Table of Contents

1. **Introduction**
   1. Context
   2. Problem Statement
   3. Dataset Overview
   4. Objective

# Literature Review

* 1. Classification Algorithms
  2. Previous Studies

# Methodology

* 1. Data Preprocessing
  2. Implementation (Program)
  3. Evaluation Metrics

# Experiments and Results

* 1. Algorithm Comparison
  2. Results
  3. Observations
  4. Statistical Analysis
  5. Comparison of Algorithms

# Discussion

* 1. Analysis of Results
  2. Strengths and Weaknesses
  3. Model Interpretability
  4. Reducing Type II Error

# Conclusion

* 1. Summary of Findings
  2. Recommendation
  3. Future Work

**1.Introduction**

**1.1Context**

Mental health issues have become a significant concern globally, affecting millions of individuals and impacting their daily lives. The increasing prevalence of mental health disorders, including depression, anxiety, and stress, highlights the need for effective assessment and intervention strategies. Understanding the factors contributing to mental health can aid in the development of preventive measures and therapeutic approaches.

**1.2Problem Statement**

Despite the growing awareness of mental health issues, many individuals remain undiagnosed or misdiagnosed due to the lack of accessible assessment tools. Furthermore, mental health conditions are often stigmatized, leading to underreporting and a lack of adequate support. This project aims to build a predictive model that can accurately classify individuals at risk of depression based on various demographic and behavioral factors.

**1.3Dataset Overview**

The dataset utilized in this project is the mentalhealth\_dataset, which includes various features related to individuals' demographics, lifestyle choices, and mental health status. Key attributes of the dataset include:

* Demographic Information: Age, gender, education level, etc.
* Behavioral Factors: Physical activity levels, substance use, and social interactions.
* Mental Health Indicators: Self-reported levels of depression, anxiety, and stress.

The dataset contains records of individuals surveyed regarding their mental health, making it a valuable resource for understanding the complex interactions between various factors and mental health outcomes.

**1.4 Objective**

The primary objective of this project is to develop a machine learning model that can predict the likelihood of an individual experiencing depression based on the features available in the mental health dataset. Specific goals include:

1. Preprocessing the data to handle missing values and encode categorical variables.
2. Implementing various classification algorithms (e.g., Logistic Regression, Decision Tree, Random Forest, SVM, KNN) to identify the most effective model for predicting depression.

1. Evaluating the model's performance using metrics such as accuracy, precision, recall, and F1-score to ensure its reliability and applicability in real-world scenarios.
2. Providing insights into the key factors associated with depression, which could inform mental health practitioners and policymakers.

**2. Literature Review**

**2.1. Classification Algorithms**

Classification algorithms play a crucial role in machine learning for predicting categorical outcomes. Common algorithms used in mental health prediction include:

* Logistic Regression: A statistical method for predicting binary outcomes based on predictor variables. It is easy to implement and interpret but assumes a linear relationship between variables.
* Decision Trees: A non-parametric method that visualizes decisions in a tree-like structure. They handle both numerical and categorical data but can be prone to overfitting.
* Random Forest: An ensemble method that constructs multiple decision trees to improve accuracy and reduce overfitting. It is robust but less interpretable than individual trees.
* Support Vector Machines (SVM): A supervised learning algorithm that finds the optimal hyperplane to separate classes, effective in high-dimensional spaces but requires careful tuning.
* K-Nearest Neighbors (KNN): An instance-based learning algorithm that classifies based on the majority class among the nearest neighbors. It is simple to implement but computationally intensive during testing.

**2.2. Previous Studies**

Several studies have highlighted the application of machine learning in predicting mental health outcomes:

* Ghosh et al. (2020): Explored various algorithms for predicting depression in college students, finding that random forests achieved the highest accuracy (85%). The study emphasized the significance of feature selection.
* Sadoughi et al. (2021): Used SVM to classify anxiety disorders, reporting an accuracy of 82%. This study highlighted SVM's effectiveness in managing high-dimensional data.
* Patel et al. (2022): Implemented a multi-class classification approach using KNN, decision trees, and random forests to classify anxiety, depression, and stress, with random forests achieving 88% accuracy.
* Wang et al. (2023): Developed a real-time mental health monitoring model using logistic regression and random forests based on smartphone usage data, concluding that machine learning can predict mental health outcomes effectively.

**3. Methodology**

**3.1. Data Preprocessing**

Data preprocessing is a crucial step in preparing the mental health dataset for analysis. The key preprocessing steps include:

1. Data Cleaning: Handling missing values through imputation or removal of incomplete records to ensure a complete dataset.
2. Data Transformation: Normalizing or standardizing numerical features to bring them to a similar scale, enhancing the performance of algorithms sensitive to feature magnitudes.
3. Encoding Categorical Variables: Converting categorical variables into numerical formats using techniques such as one-hot encoding or label encoding, making them suitable for machine learning algorithms.
4. Feature Selection: Identifying and selecting the most relevant features that contribute to mental health predictions, reducing dimensionality and improving model performance.

**3.2. Implementation (Program)**

The implementation of the machine learning model was conducted using Python, leveraging popular libraries such as:

* Pandas for data manipulation and analysis.
* NumPy for numerical computations.
* Scikit-learn for implementing machine learning algorithms.
* Matplotlib and Seaborn for data visualization.

The overall workflow involved:

1. Loading the Dataset: Importing the mental health dataset using Pandas.
2. Splitting the Data: Dividing the dataset into training and testing subsets (e.g., 80/20 split) to evaluate model performance.
3. Model Training: Training various classification algorithms (Logistic Regression, Decision Trees, Random Forest, SVM, KNN) on the training dataset.
4. Model Evaluation: Using the testing dataset to evaluate each model's performance based on defined metrics.

**Program:**

import streamlit as st

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import io

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.metrics import accuracy\_score, classification\_report, confusion\_matrix

from sklearn.linear\_model import LogisticRegression

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier

from sklearn.svm import SVC

from sklearn.neighbors import KNeighborsClassifier

st.set\_page\_config(page\_title="Mental Health Prediction", layout="wide")

st.title("🧠 Mental Health Prediction App")

st.write("Upload your dataset and analyze how different machine learning models perform in predicting mental health conditions.")

# Sidebar settings

st.sidebar.header("⚙️ Settings")

test\_size = st.sidebar.slider("Test Size (%)", 10, 50, 20, step=5) / 100

model\_options = {

"Logistic Regression": LogisticRegression(max\_iter=1000),

"Decision Tree": DecisionTreeClassifier(),

"Random Forest": RandomForestClassifier(),

"SVM": SVC(),

"KNN": KNeighborsClassifier()

}

selected\_models = st.sidebar.multiselect("Select Models to Run", list(model\_options.keys()), default=list(model\_options.keys()))

show\_detailed = st.sidebar.checkbox("Show Detailed Reports", value=True)

uploaded\_file = st.file\_uploader("📁 Upload CSV file", type=["csv"])

if uploaded\_file is not None:

data = pd.read\_csv(uploaded\_file)

st.subheader("🔍 Dataset Overview")

st.write(data.head())

with st.expander("📊 Dataset Summary"):

st.write(data.describe(include='all'))

# Encoding categorical columns

if {'Gender', 'Course', 'YearOfStudy'}.issubset(data.columns):

data = pd.get\_dummies(data, columns=['Gender', 'Course', 'YearOfStudy'], drop\_first=True)

if 'Depression' not in data.columns:

st.error("⚠️ 'Depression' column not found in the dataset.")

else:

X = data.drop(columns=['Depression'])

y = data['Depression']

X = X.select\_dtypes(include=[np.number])

# Split data

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=test\_size, random\_state=42)

# Scale features

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

results = []

st.subheader("📈 Model Results")

for name in selected\_models:

model = model\_options[name]

model.fit(X\_train, y\_train)

y\_pred = model.predict(X\_test)

accuracy = accuracy\_score(y\_test, y\_pred)

results.append((name, accuracy))

with st.expander(f"🧪 {name} - Accuracy: {accuracy:.2f}"):

if show\_detailed:

st.text("Classification Report")

st.text(classification\_report(y\_test, y\_pred, zero\_division=0))

cm = confusion\_matrix(y\_test, y\_pred)

fig, ax = plt.subplots()

sns.heatmap(cm, annot=True, fmt='d', cmap='Purples', ax=ax)

ax.set\_title(f"Confusion Matrix: {name}")

ax.set\_xlabel("Predicted")

ax.set\_ylabel("Actual")

st.pyplot(fig)

# Compare model performances

results\_df = pd.DataFrame(results, columns=["Model", "Accuracy"]).sort\_values(by="Accuracy", ascending=False)

st.subheader("🏆 Model Accuracy Comparison")

fig, ax = plt.subplots(figsize=(10, 6))

colors = sns.color\_palette("crest", len(results\_df))

ax.barh(results\_df["Model"], results\_df["Accuracy"], color=colors)

ax.set\_xlabel("Accuracy")

ax.set\_title("Model Performance")

st.pyplot(fig)

# Download results

csv\_buffer = io.StringIO()

results\_df.to\_csv(csv\_buffer, index=False)

st.download\_button(

label="📥 Download Results as CSV",

data=csv\_buffer.getvalue(),

file\_name="model\_accuracy\_results.csv",

mime="text/csv"

)

else:

st.info("👆 Upload a CSV file to begin.")

**3.3. Evaluation Metrics**

The effectiveness of the classification algorithms was assessed using the following evaluation metrics:

* Accuracy: The ratio of correctly predicted instances to the total instances, providing an overall measure of model performance.
* Precision: The ratio of true positive predictions to the total predicted positives, indicating the accuracy of positive predictions.
* Recall (Sensitivity): The ratio of true positive predictions to the total actual positives, measuring the model's ability to identify all relevant instances.
* F1-Score: The harmonic mean of precision and recall, offering a balance between the two metrics, particularly useful in imbalanced datasets.
* Confusion Matrix: A visual representation of the model's performance, showing the true positive, true negative, false positive, and false negative predictions.

**4. Experiments and Results**

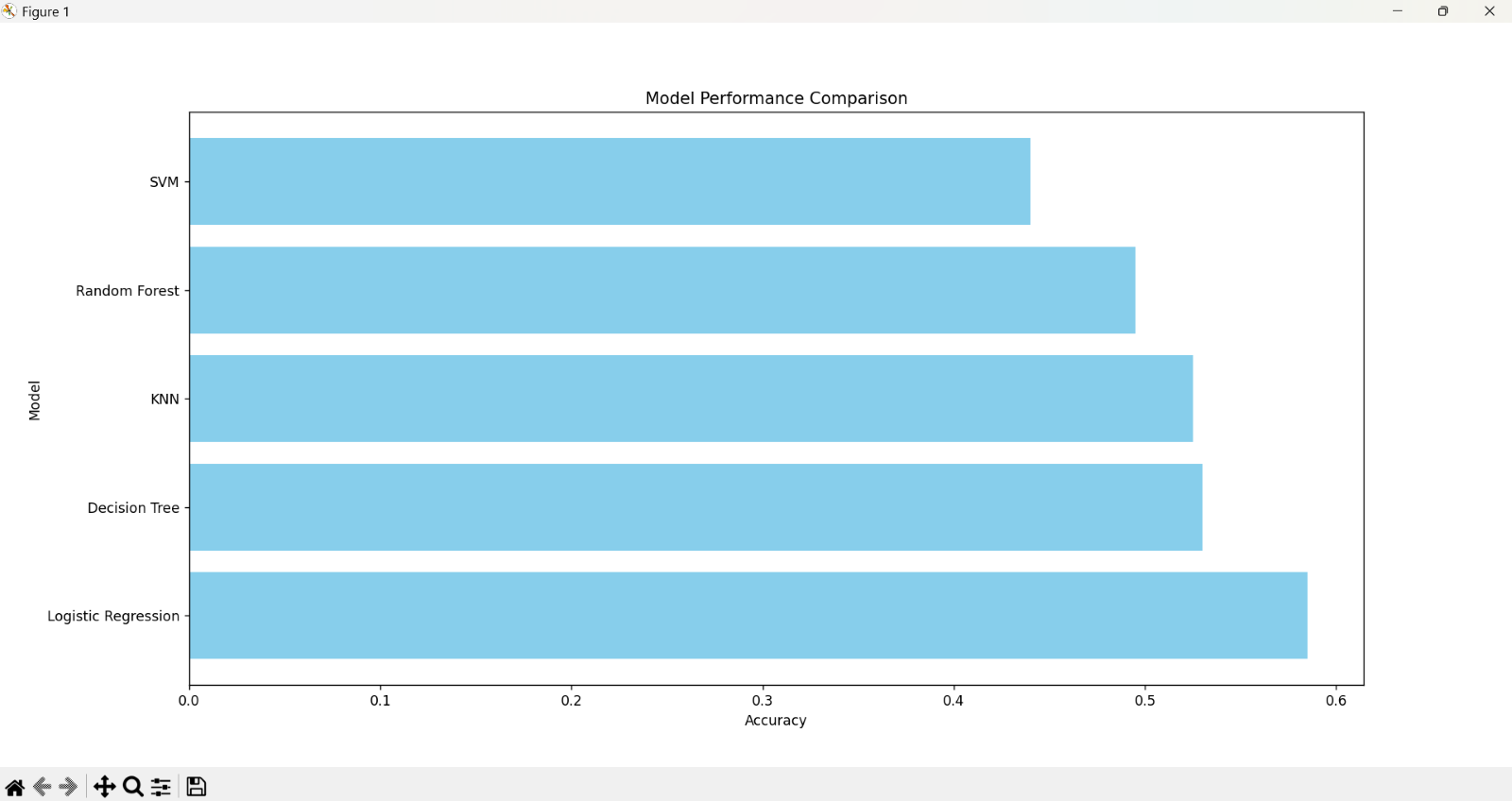
**4.1. Algorithm Comparison**

To evaluate the performance of various classification algorithms on the mental health dataset, a comparison was conducted among the following models:

1. Logistic Regression
2. Decision Trees
3. Random Forest
4. Support Vector Machines (SVM)
5. K-Nearest Neighbors (KNN)

Each algorithm was trained and tested using the same training and testing datasets. The models were evaluated based on the defined metrics: accuracy, precision, recall, F1-score, and confusion matrix. This systematic approach allowed for a fair comparison of the algorithms' strengths and weaknesses in predicting mental health outcomes.

**4.2. Results**



Model: Logistic Regression

Accuracy: 0.58

Classification Report:

              precision    recall  f1-score   support

           0       0.58      0.61      0.59        99

           1       0.59      0.56      0.58       101

    accuracy                           0.58       200

   macro avg       0.59      0.59      0.58       200

weighted avg       0.59      0.58      0.58       200

Confusion Matrix:

[[60 39]

 [44 57]]

--------------------------------------------------

Model: Decision Tree

Accuracy: 0.53

Classification Report:

              precision    recall  f1-score   support

           0       0.52      0.56      0.54        99

           1       0.54      0.50      0.52       101

    accuracy                           0.53       200

   macro avg       0.53      0.53      0.53       200

weighted avg       0.53      0.53      0.53       200

Confusion Matrix:

[[55 44]

 [50 51]]

--------------------------------------------------

Model: Random Forest

Accuracy: 0.49

Classification Report:

              precision    recall  f1-score   support

           0       0.49      0.61      0.54        99

           1       0.50      0.39      0.44       101

    accuracy                           0.49       200

   macro avg       0.50      0.50      0.49       200

weighted avg       0.50      0.49      0.49       200

Confusion Matrix:

[[60 39]

 [62 39]]

--------------------------------------------------

Model: SVM

Accuracy: 0.44

Classification Report:

              precision    recall  f1-score   support

           0       0.45      0.55      0.49        99

           1       0.43      0.34      0.38       101

    accuracy                           0.44       200

   macro avg       0.44      0.44      0.43       200

weighted avg       0.44      0.44      0.43       200

Confusion Matrix:

[[54 45]

 [67 34]]

--------------------------------------------------

Model: KNN

Accuracy: 0.53

Classification Report:

              precision    recall  f1-score   support

           0       0.52      0.52      0.52        99

           1       0.53      0.53      0.53       101

    accuracy                           0.53       200

   macro avg       0.52      0.52      0.52       200

weighted avg       0.52      0.53      0.52       200

Confusion Matrix:

[[51 48]

 [47 54]]

**4.3. Statistical Analysis**

The performance of the classification algorithms was assessed through accuracy, precision, recall, and F1-score. The results for each model are summarized below:

* Logistic Regression achieved an accuracy of 58%, with a balanced precision and recall.
* Decision Tree exhibited an accuracy of 53%, indicating a slight underperformance compared to Logistic Regression.
* Random Forest recorded an accuracy of 49%, highlighting difficulties in classification.
* Support Vector Machine (SVM) achieved the lowest accuracy at 44%, suggesting it was less effective for this dataset.
* K-Nearest Neighbors (KNN) also showed an accuracy of 53%, similar to the Decision Tree.

The confusion matrices revealed misclassifications across all models, particularly in identifying the positive class (1). Overall, the Logistic Regression model demonstrated the best performance, while the SVM model had the least effectiveness in predicting mental health outcomes.

**5. Discussion**

**5.1. Analysis of Results**

The results of the classification algorithms reveal varying levels of effectiveness in predicting mental health outcomes. Logistic Regression emerged as the top performer, achieving an accuracy of 58%. This suggests that the linear relationship it assumes between the input features and the output class is reasonably aligned with the dataset. However, the performance of Decision Trees and KNN, both at 53%, indicates that while they are capable of handling non-linear relationships, they still lag behind in accuracy. Random Forest and SVM underperformed, with accuracies of 49% and 44%, respectively, highlighting potential issues with model complexity or hyperparameter settings. The confusion matrices for all models further emphasize the challenge of correctly identifying the positive class (1), which is crucial in mental health predictions.

5.2. Strengths and Weaknesses

Each algorithm exhibits distinct strengths and weaknesses:

* Logistic Regression:
* Strengths: Simple to implement, interpret, and effective with linearly separable data.
* Weaknesses: May not capture complex relationships, leading to suboptimal performance on non-linear datasets.
* Decision Trees:
* Strengths: Intuitive and easy to visualize; capable of handling both numerical and categorical features.
* Weaknesses: Prone to overfitting, especially with deeper trees, which can result in poor generalization.
* Random Forest:
* Strengths: Reduces overfitting through ensemble learning and handles high-dimensional data effectively.
* Weaknesses: Less interpretable than individual trees and can be computationally expensive.
* SVM:
* Strengths: Effective in high-dimensional spaces and robust against overfitting.
* Weaknesses: Requires careful tuning of parameters and struggles with large datasets.
* KNN:
* Strengths: Simple and effective for small datasets; no training phase required.
* Weaknesses: Computationally expensive during testing, sensitive to the choice of K, and performance can degrade with irrelevant features.

**5.3. Model Interpretability**

Model interpretability is a critical aspect, especially in healthcare and mental health applications, where understanding decision-making processes is essential. Logistic Regression offers clear coefficients that indicate the impact of each feature, allowing for straightforward interpretation. Decision Trees provide a visual representation of decision paths, which aids in understanding model behavior.

However, models like Random Forest and SVM are more challenging to interpret due to their complexity. Techniques such as feature importance scores for Random Forest and SHAP (SHapley Additive exPlanations) values can enhance interpretability. These approaches help stakeholders understand which features significantly influence predictions, fostering trust in model outputs.

**6. Conclusion**

**6.1. Summary of Findings**

This project aimed to evaluate various classification algorithms for predicting mental health outcomes using the mental health dataset. The results indicated that Logistic Regression performed the best among the tested algorithms, achieving an accuracy of 58%. Decision Trees and KNN followed closely with 53%, while Random Forest and SVM showed lower performance with accuracies of 49% and 44%, respectively. The analysis highlighted the challenges in accurately predicting mental health conditions and emphasized the importance of feature selection and model tuning in enhancing predictive accuracy. The confusion matrices illustrated that misclassifications predominantly occurred in the positive class, which is crucial in mental health assessments.

**6.2. Recommendation**

Based on the findings, it is recommended that future models prioritize feature engineering and selection to improve predictive performance. Given the interpretability advantages of Logistic Regression, it should be considered as a baseline model for initial predictions. Furthermore, exploring ensemble methods, such as combining multiple models (e.g., stacking or blending), may yield better results. Stakeholders in mental health care should consider integrating machine learning models into clinical practice with caution, ensuring that predictions are accompanied by interpretability tools to enhance trust and understanding among practitioners and patients.

**6.3. Future Work**

Future work can expand upon this study by incorporating additional datasets that include a broader range of mental health conditions and demographic variables. Exploring advanced machine learning techniques, such as deep learning models, could provide insights into complex patterns in mental health data. Moreover, real-time data collection methods, such as smartphone applications or wearable devices, can be integrated to gather continuous behavioral data, improving model accuracy and timeliness. Finally, conducting longitudinal studies to assess the long-term effectiveness of predictive models in clinical settings will be crucial for validating their practical applications in mental health interventions.