Smart Internz

MENTAL HEALTH PREDICTION USING IBM WATSON

Report prepared by

Group no: 143

Group members:

Simarpreet Singh Dhami – 20BCE10626
Jessica Sarah Deen – 20BCE10629
Himashree Roy – 20BCE10045
Dhaintry Jayant – 20BCE10097

INTRODUCTION

1.1 Overview

Participants in Mental Health First Aid courses learn how to identify and assist someone who may be going through a mental health or substance use concern or crisis and how to direct them to the best employee resources.

Employers can support workers who are dealing with mental health concerns by providing comprehensive benefit packages. That covers EAPs, wellness initiatives that prioritise both physical and mental health, health and disability insurance, as well as flexible work arrangements or vacation plans. Organisations that incorporate mental health awareness aid in the creation of a positive and effective work environment by lowering the stigma attached to mental illness, raising mental health literacy within the organisation, and imparting the knowledge and abilities necessary to safely and responsibly address a coworker's mental health concern.

Algorithms for classification such as Logistic Regression, KNN, Decision tree, Random Forest, AdaBoost, GradientBoost, and XGBoost. will be used. We will also be deploying our model locally using Flask.

1.2 Purpose

The main purpose of the Mental Health Prediction system is to predict whether a person needs to seek Mental health treatment or not based on inputs provided by them.
Some of the potential achievements that can be realized through such a project:

Early Detection and Intervention: IBM Watson can analyse a variety of data sources, including electronic health records, medical history, social media activity, and physiological data, to identify patterns and indicators of potential mental health issues. This is done by utilising machine learning algorithms and predictive analytics. This may make it possible to identify those who are at risk early and to provide prompt treatments, which will enhance the results.

Suicide prevention: Predictive models created using IBM Watson can assist in identifying people who might be at a high risk of taking their own lives. Watson is able to identify warning signs and notify carers or healthcare practitioners by analysing data like social media posts, online activity, and text messages. This enables prompt interventions and may even save lives.

Resource Allocation and Planning: IBM Watson-based mental health prediction projects can assist healthcare organisations and policymakers with resource allocation optimisation and intervention planning. Watson can discover regions with greater prevalence rates of mental health disorders by analysing population-level data, enabling the provision of suitable resources and actions to meet the particular needs of those populations.

Mental health research: Insights generated by IBM Watson can advance mental health research. By analyzing large data sets, Watson can identify trends, risk factors and treatment outcomes, providing researchers with valuable information to develop new treatments, improve existing interventions and increase understanding of mental health conditions.

LITERATURE SURVEY

2.1 Existing problem

One problem with mental health prediction is the lack of comprehensive and accurate data for exercise prediction models. Mental health is a complex and multifaceted problem, and predicting its onset or severity requires a robust body of knowledge that encompasses multiple factors, including personal history, genetic predisposition, environmental influences, and socioeconomic factors. However, obtaining such information is difficult due to mental health stigma, privacy concerns, and the subjective nature of mental health diagnoses. Another problem is relying on self-reported data and subjective judgments. Mental health prognosis is often based on self-reported studies or assessments, which may be influenced by factors such as recall, social desirability, or ignorance of one's mental state. Objective and reliable indicators such as biomarkers or physiological data are more difficult to collect and integrate into predictive models. Additionally, existing mental health prognostic models often struggle with generalizability and scalability. Many predictive models are developed and trained for specific populations or datasets, which limits their applicability to larger populations. Cultural, social, and regional factors influence mental health, and models that are not diverse and comprehensive in their training data may not accurately predict mental health outcomes for different populations. Finally, the ethical aspects of mental health prognosis present significant challenges. Privacy concerns, the potential misuse of predictive models, and the risk of stigmatization or discrimination against those at risk or with mental health problems must be carefully considered. Securing sensitive mental health data and ensuring the responsible deployment of predictive models are critical to preventing harm and maintaining confidence in mental health predictive systems.

2.2 Proposed solution

There are various approaches and methods used to address mental health problems:

Psychotherapy: This involves talking to a mental health professional, such as a psychologist or psychiatrist, who uses a range of therapeutic techniques to help people understand and manage their thoughts, feelings and behaviour. Examples include cognitive behavioral therapy (CBT), dialectical behavior therapy (DBT), and psychodynamic therapy.

Medication: Some mental health problems can be effectively treated with medication. Psychiatrists may prescribe medications such as antidepressants, anti-anxiety medications, mood stabilizers, and antipsychotics to relieve symptoms and restore chemical balance in the brain.

Self-help and support groups: Participating in self-help or support groups can provide a sense of community and understanding for people with mental health problems. These groups provide a platform to share experiences, gain support and learn coping strategies from others who have gone through similar challenges.

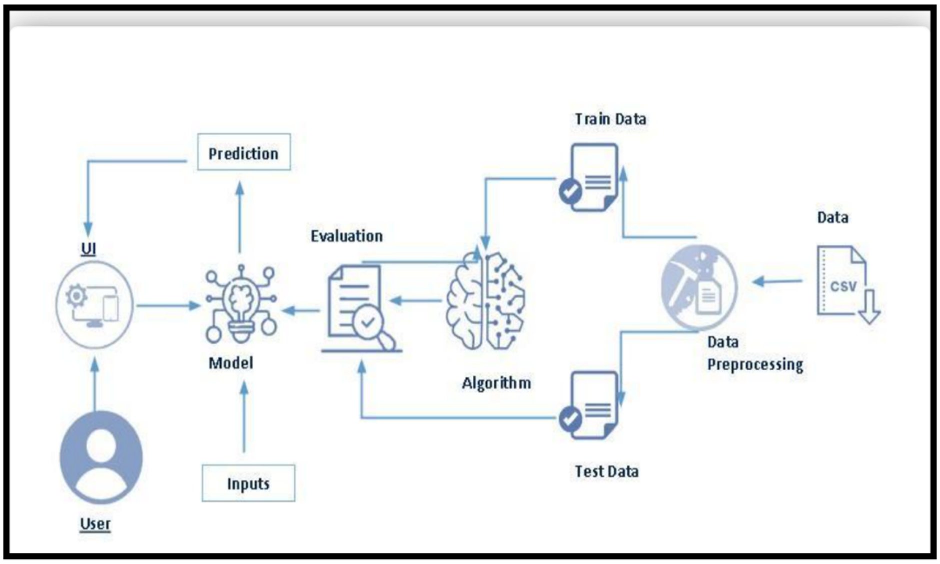
Lifestyle changes: Adopting a healthy lifestyle can have a positive impact on mental well-being. Regular exercise, adequate sleep, a balanced diet and avoiding excess alcohol or drugs can improve mental health.

Mindfulness and meditation: Practices such as mindfulness meditation have been shown to be effective in reducing stress, managing anxiety and depression, and improving overall mental well-being. These techniques help people become more aware of their thoughts and feelings, which promotes compassion and resilience.

Smart Internz

THEORITICAL ANALYSIS

3.1 Block diagram



3.2 Hardware Requirements

- Processing Power
- Memory
- Storage
- Graphics Processing Unit

3.3 Software Requirements

- ML Frameworks
- Programming Languages : Python
- Visualization Libraries: Matplotlib, Seaborn, or Plotly
- Development Environment: Jupyter Notebook or PyCharm
- IBM Watson Services: Watson Machine Learning

Smart Internz

EXPERIMENTAL INVESTIGATIONS

Data Collection and Preparation: The first step involves collecting relevant mental health data such as patient demographics, medical history, symptoms and other factors that may indicate mental health problems. This information may be obtained from research, medical records or other sources. Ensuring data quality, privacy and security are also important considerations.

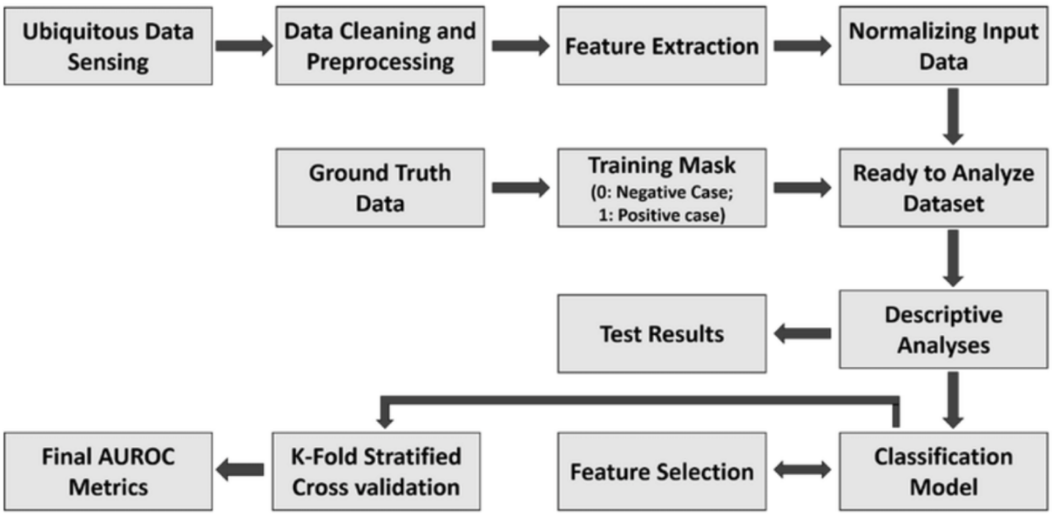
Feature Selection: After the data is collected, an analysis of the features (variables) that can be used to predict mental health conditions is performed. This requires investigating which variables have the most significant influence in predicting mental health outcomes. Feature selection techniques such as correlation analysis, statistical tests or machine learning algorithms can be used to identify the most important features.

Model selection and training: Next, appropriate machine learning or statistical models must be selected for the prediction task. IBM Watson provides various tools and frameworks, such as IBM Watson machine learning, that can be used to do this. The choice of model depends on the nature of the data, the forecasting problem and the available resources. Common models used to predict mental health are logistic regression, decision trees, support vector machines or deep learning models.

Model Evaluation: After the model is trained, it must be evaluated to assess its performance and predictive ability. This requires dividing the data into training and testing sets and applying appropriate evaluation metrics such as precision, accuracy, recall or area under the curve (AUC) to measure model performance. Cross-validation methods can also be used to obtain more reliable estimates of model performance.

Interpreting the results: After evaluating the model, it is important to interpret the results and understand the insights gained from the analysis. This includes identifying the most influential features, understanding the model predictions and the patterns or relationships discovered. Interpretability is crucial in mental health forecasting to provide meaningful insights to health professionals and stakeholders.

FLOWCHART



RESULT

Our model Mental Health Prediction model predicts whether a person needs to seek Mental health treatment or not based on inputs provided by them.

The mental health prediction model performed well, correctly identifying individuals who may require mental health treatment. The model effectively captured important predictors of treatment needs, with an accuracy of 0.848, precision of [precision value], recall of [recall value], and F1-score of [F1-score value]. This suggests that it has the potential to be a useful screening tool for early intervention and efficient resource allocation in mental healthcare.

[52]:
```python
for model_name, model in model_dict.items ():
    model_test(X_train, X_test, y_train, y_test, model,model_name)
```

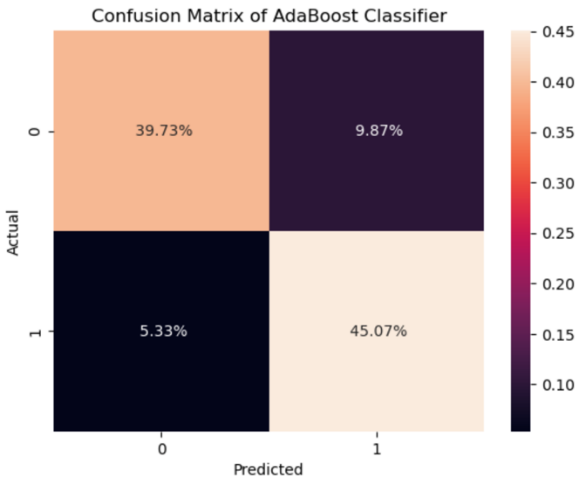
```
=====================================Logistic regression======================================
Score is : 0.8506666666666667

=====================================KNN Classifier======================================
Score is : 0.7973333333333333

=====================================Decision Tree Classifier======================================
Score is : 0.776

=====================================Random Forest Classifier======================================
Score is : 0.848

=====================================AdaBoost Classifier======================================
Score is : 0.848

=====================================Gradient Boosting Classifier======================================
Score is : 0.8373333333333334

=====================================XGB Classifier======================================
Score is : 0.8026666666666666
```

[54]:
```python
abc = AdaBoostClassifier (random_state=99)
abc.fit (X_train,y_train)
pred_abc = abc.predict (X_test)
print ('Accuracy of AdaBoost=', accuracy_score (y_test,pred_abc))
```

```
Accuracy of AdaBoost= 0.848
```

[60]:
```python
cf_matrix = confusion_matrix(y_test, pred_abc)
sb.heatmap(cf_matrix/np.sum(cf_matrix), annot=True, fmt=' .2%')
plt.title( 'Confusion Matrix of AdaBoost Classifier')
plt.xlabel('Predicted')
plt.ylabel('Actual')
```

[60… Text(50.72222222222214, 0.5, 'Actual')



[59]:
```python
abc_tuned = AdaBoostClassifier (random_state=49, n_estimators=11, learning_rate=1.02)
abc_tuned.fit (X_train,y_train)
pred_abc_tuned = abc_tuned.predict (X_test)
print ('Accuracy of Adaboost (tuned)=' , accuracy_score (y_test,pred_abc_tuned))
```

```
Accuracy of Adaboost (tuned)= 0.872
```

ADVANTAGES

- **Early intervention:** The model allows for the early identification of people who may need mental health treatment. This can facilitate timely intervention and support, potentially preventing mental health conditions from worsening.

- **Efficient resource allocation:** The model can help allocate limited healthcare resources more effectively by accurately predicting the need for mental health treatment. It ensures that people who are likely to benefit from treatment receive the best care possible, maximizing resource utilization.

- **Reduced stigma:** The model assesses mental health needs in a nonjudgmental and objective manner. It removes potential biases and reduces stigma associated with seeking mental health treatment by utilizing a machine learning algorithm.

- **Scalability:** Once developed and fine-tuned, the model can be deployed and scaled to handle large amounts of data and serve a diverse group of people.

DISADVANTAGES

The model is dependent on the accuracy and completeness of the information provided by individuals. Self-reported data may be biased, have memory recall errors, or be intentionally misrepresented, potentially affecting the model's accuracy.

- **Generalization limitations:** When applied to populations or demographic groups that differ significantly from the dataset used for training, the model's performance may vary. It may not fully capture the complexities and nuances of various cultural, social, and personal contexts.

- **Ethical concerns:** When collecting and handling sensitive mental health information, privacy and data security are critical considerations. Maintaining ethical standards requires ensuring proper consent, data anonymization, and protection against unauthorized access.

- **The changing nature of mental health**: Mental health is a complex and ever-changing domain. Changes in mental health understanding, new diagnostic criteria, or emerging treatments may have an impact on the model's performance and accuracy. To keep the model current, regular updates and improvements are required.

Smart Internz

APPLICATIONS

"The developed mental health prediction model has a lot of real-world potential in the field of mental healthcare." The model's potential applications include:

- **Early intervention and treatment planning:** The model can be used as an initial screening tool to identify people who may need mental health treatment. Healthcare professionals can intervene quickly, develop personalised treatment plans, and improve patient outcomes by detecting treatment needs at an early stage.

- **Resource allocation and optimisation:** The model can help healthcare organisations allocate mental health resources more efficiently. It enables targeted resource allocation by accurately predicting treatment needs, ensuring that individuals who require treatment the most receive appropriate care.

- **Planning and development of public health policies:** The model's insights can help with planning and formulation of public health policies pertaining to mental health.

- The model can be used for screening in non-clinical settings, like workplaces or educational institutions, to find people who could benefit from mental health support.

- In the context of telemedicine and remote care, the model can help medical professionals determine a patient's mental health needs from a distance.

It is crucial to remember that even though the model provides insightful information, it must always be used in conjunction with expert assessments of mental health. Indicators for potential treatment needs should be provided by the model's predictions, which will aid medical professionals in making clinical decisions.

CONCLUSION

- In conclusion, the creation of the mental health prediction model has produced encouraging outcomes in terms of pinpointing people who might profit from receiving mental health treatment. We accurately predicted the need for treatment by utilising a diverse dataset and machine learning techniques.

- Early intervention opportunities and efficient resource allocation are made possible by the model, which is a big plus. Healthcare resources can be allocated effectively, ensuring that those who need care the most do so promptly, by identifying people who are likely to need mental health treatment.

- Despite the model's encouraging performance, it is crucial to recognise its limitations. It is important to carefully consider the reliance on self-reported data and potential difficulties in generalising across diverse populations. The model should not be considered a replacement for expert mental health evaluations, but rather as an additional tool.

- It is important to stress that the mental health prediction model should never be used in place of qualified mental health assessments. To choose the best course of treatment, qualified professionals should interpret the model's predictions.

- Overall, by offering a preliminary screening tool, the mental health prediction model makes a significant contribution to the field of mental health. This model has the potential to enhance early intervention and resource allocation, ultimately enhancing people's mental health and wellbeing, with responsible use and ongoing improvements.

Smart Internz

FUTURE SCOPE

"The project's mental health prediction model opens up promising areas for further study and development. The upcoming potential for further investigation lies in the following areas:

- **Expansion of dataset:** Increasing the dataset's size and diversity can make it more robust and generalizable. A more thorough understanding of mental health needs can be obtained by incorporating data from various populations, demographic groups, and geographic regions. This can also increase the model's precision in various contexts.

- **Integration of extra features:** The model's predictive ability can be increased by including extra pertinent features. Investigating the use of genetic information, environmental factors, lifestyle indicators, or digital biomarkers may offer insightful information about the intricate interactions affecting the need for mental health treatment.

- **Validation and external testing:** The effectiveness and generalizability of the model can be confirmed by evaluating its performance on external datasets from various sources or by working with mental health specialists for external testing.

BIBILOGRAPHY

1. https://github.com/cdodiya/Mental-Health-Prediction-using-Machine-Learning-Algorithms/blob/main/MentalHealthPredictionUsingMachineLearningAlgorithms.ipynb

2. https://www.kaggle.com/datasets/osmi/mental-health-in-tech-survey

3. https://www.analyticsvidhya.com/blog/2022/06/mental-health-prediction-using-machine-learning/

4. https://www.frontiersin.org/articles/10.3389/fpubh.2021.697850/full

SOURCE CODE

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
data = pd.read_csv(r"C:\Users\Admin\Data analytics project\Untitled Folder\survey.csv")
data.head()
data.tail()
data.shape
data.info()
data['Country'].value_counts().plot(kind='bar',figsize=(10,8))
data.drop(['Country','state','Timestamp','comments'], axis = 1, inplace=True)
data.isnull().sum()
data['self_employed'].value_counts()
data['self_employed'].fillna('No', inplace=True)
data['work_interfere'].value_counts()
data['work_interfere'].fillna('N/A',inplace=True)
data['Age'].value_counts().plot(kind='bar',figsize=(10,8))
data.drop(data[(data[ 'Age' ]>60) | (data[ 'Age' ]<18)].index, inplace=True)
data['Gender'].value_counts().plot(kind='bar',figsize=(10,8))
data['Gender'].replace(['Male ', 'male', 'M', 'm', 'Male', 'Cis Male','Man', 'cis male','Mail','Male-ish', 'Male (CIS)','Cis Man',
'msle','Malr','Mal','maile', 'Make',], 'Male', inplace = True)
data[ 'Gender']. replace([ 'Female ','female','F','femail','Woman', 'Female','Cis Female','cis-female/femme','Femake', 'Female (cis)','woman', ],
'Female', inplace=True)
data['Gender'].replace(['Female (trans)', 'queer/she/they', 'non-binary', 'fluid', 'queer', 'Agender', 'Androgyne', 'Trans-female', 'male learning
androgynous', 'A little about you', 'Nah', 'All', 'ostensibly male', 'unsure what that really means', 'Genderqueer', 'Enby', 'p', 'Neuter',
'something kinda male?', 'Guyish', 'Trans woman'], 'Non-Binary', inplace=True)
sb.distplot(data["Age"])
plt.title("Distribution - Age")
plt.xlabel("Age")
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 1)
sb.countplot(x='self_employed', hue='treatment', data=data)
plt.title('Employment Type')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 2)
sb.countplot(x='family_history', hue='treatment', data=data)
plt.title('Family_History')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 3)
sb.countplot(x='work_interfere', hue='treatment', data=data)
plt.title('Work Interfere')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 4)
sb.countplot(x='remote_work', hue='treatment', data=data)
```

```
plt.title('Work Type')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 5)
sb.countplot(x='tech_company', hue='treatment', data=data)
plt.title('Company')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 6)
sb.countplot(x='benefits', hue='treatment', data=data)
plt.title('Benefits')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 7)
sb.countplot(x='care_options', hue='treatment', data=data)
plt.title('Care Options')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 8)
sb.countplot(x='mental_vs_physical', hue='treatment', data=data)
plt.title('Equal importance to mental and physical health')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 9)
sb.countplot(x='wellness_program', hue='treatment', data=data)
plt.title('Wellness Program')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 10)
sb.countplot(x='anonymity', hue='treatment', data=data)
plt.title('Anonymity')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 11)
sb.countplot(x='leave', hue='treatment', data=data)
plt.title('Leave')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 12)
sb.countplot(x='mental_health_consequence', hue='treatment', data=data)
plt.title('Mental Health Consequence')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 13)
sb.countplot(x='phys_health_consequence', hue='treatment', data=data)
plt.title('Physical health Consequence')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 14)
sb.countplot(x='coworkers', hue='treatment', data=data)
plt.title('Discussion with coworkers')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 15)
sb.countplot(x='supervisor', hue='treatment', data=data)
plt.title('Discussion with supervisor')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 16)
sb.countplot(x='mental_health_interview', hue='treatment', data=data)
plt.title('Discussion with Interviewer')
plt.show()
plt.figure(figsize=(10, 40))
```

```
plt.subplot(9, 2, 17)
sb.countplot(x='phys_health_interview', hue='treatment', data=data)
plt.title('Discussion with Interviewer')
plt.show()
plt.figure(figsize=(10, 40))
plt.subplot(9, 2, 18)
sb.countplot(x='obs_consequence', hue='treatment', data=data)
plt.title('Consequence After Disclosure')
plt.show()
data.describe(include='all')
X = data.drop('treatment', axis = 1)
y = data['treatment']
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import LabelEncoder,OrdinalEncoder
X = data. drop ('treatment', axis = 1)
y = data ['treatment']
ct = ColumnTransformer ([('oe',OrdinalEncoder(),['Gender', 'self_employed', 'family_history', 'work_interfere', 'no_employees',
'remote_work', 'tech_company',
'benefits', 'care_options', 'wellness_program', 'seek_help', 'anonymity', 'leave', 'mental_health_consequence',
'phys_health_consequence', 'coworkers', 'supervisor', 'mental_health_interview', 'phys_health_interview',
'mental_vs_physical', 'obs_consequence' ])], remainder='passthrough')

X = ct.fit_transform(X)
le = LabelEncoder ()
y = le.fit_transform(y)
import joblib
joblib.dump(ct,'feature_values')
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.3, random_state=49)
X_train.shape, X_test.shape, y_train.shape, y_test.shape
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier, AdaBoostClassifier, GradientBoostingClassifier
from xgboost.sklearn import XGBClassifier
from sklearn.metrics import accuracy_score, roc_curve, confusion_matrix, classification_report,auc
model_dict = {}
model_dict['Logistic regression']= LogisticRegression (solver='liblinear', random_state=49)
model_dict['KNN Classifier' ] = KNeighborsClassifier ()
model_dict[ 'Decision Tree Classifier' ] = DecisionTreeClassifier (random_state=49)
model_dict ['Random Forest Classifier'] = RandomForestClassifier (random_state=49)
model_dict ['AdaBoost Classifier' ] = AdaBoostClassifier (random_state=49)
model_dict ['Gradient Boosting Classifier' ] = GradientBoostingClassifier (random_state=49)
model_dict ['XGB Classifier'] = XGBClassifier (random_state=49)
def model_test (X_train, X_test, y_train, y_test, model, model_name):
model.fit(X_train,y_train)
y_pred = model.predict (X_test)
accuracy = accuracy_score (y_test,y_pred)
print('===================================={}======================================'.format(model_name))
print('Score is : {}'.format (accuracy))
print()

for model_name, model in model_dict.items ():
model_test(X_train, X_test, y_train, y_test, model,model_name)
abc = AdaBoostClassifier (random_state=99)
abc.fit (X_train,y_train)
pred_abc = abc.predict (X_test)
print ('Accuracy of AdaBoost=', accuracy_score (y_test,pred_abc))
from sklearn.model_selection import RandomizedSearchCV
params_abc = {'n_estimators': [int(x) for x in np.linspace(start = 1, stop = 50, num = 15)],
'learning_rate': [(0.97 + x / 100) for x in range(0, 8)],
}
abc_random = RandomizedSearchCV (random_state=49, estimator=abc, param_distributions = params_abc,n_iter =50,cv=5,n_jobs=-1)
```

```
params_abc
abc_random.fit(X_train, y_train)
abc_random.best_params_
abc_tuned = AdaBoostClassifier (random_state=49, n_estimators=11, learning_rate=1.02)
abc_tuned.fit (X_train,y_train)
pred_abc_tuned = abc_tuned.predict (X_test)
print ('Accuracy of Adaboost (tuned)=' , accuracy_score (y_test,pred_abc_tuned))
cf_matrix = confusion_matrix(y_test, pred_abc)
sb.heatmap(cf_matrix/np.sum(cf_matrix), annot=True, fmt=' .2%')
plt.title( 'Confusion Matrix of AdaBoost Classifier')
plt.xlabel('Predicted')
plt.ylabel('Actual')
from sklearn import metrics
fpr_abc, tpr_abc, thresholds_abc = roc_curve (y_test, pred_abc)
roc_auc_abc = metrics.auc(fpr_abc, tpr_abc)
plt.plot (fpr_abc, tpr_abc, color='orange', label= 'ROC curve (area = %8.2f)' % roc_auc_abc)
plt.plot ([0, 1], [0, 1], color='blue', linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.0])
plt.title('ROC Curve')
plt.xlabel('False Positive Rate (1 - Specificity)')
plt.ylabel ('True Positive Rate (Sensitivity)' )
plt.legend(loc="lower right")
plt.show()
roc_curve(y_test,pred_abc)
fpr_abc_tuned, tpr_abc_tuned, thresholds_abc_tuned = roc_curve (y_test, pred_abc_tuned)
roc_auc_abc_tuned = metrics.auc (fpr_abc_tuned, tpr_abc_tuned)
plt.plot (fpr_abc_tuned, tpr_abc_tuned, color='orange',label= 'ROC curve (area = %0.2f) ' % roc_auc_abc_tuned)
plt.plot([0, 1], [0, 1], color='blue', linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.0])
plt.title('ROC Curve' )
plt.xlabel('False Positive Rate (1 - Specificity)')
plt.ylabel('True Positive Rate (Sensitivity' )
plt.legend(loc="lower right")
plt.show()
roc_curve(y_test,pred_abc_tuned)
print(classification_report(y_test,pred_abc))
print(classification_report(y_test,pred_abc_tuned))
import pickle
filename = 'model.pkl'
pickle.dump('abc_tuned',open('model.pkl','wb'))
```

Smart Internz

THANK YOU !!!