EMOTION DETECTION

A Course Project report submitted

in partial fulfillment of requirement for the award of degree

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

by

D.SRI DEEKSHITHA	**(2103A51456)**
MAHVISH ISHAQ	**(2103A51473)**
NASHEER FATIMA	**(2103A51179)**

Under the guidance of

Mr. THARUN REDDY

Assistant Professor, Department of CSE.

SR UNIVERSITY

Department of Computer Science and Artificial Intelligence

SR UNIVERSITY

Department of Computer Science and Artificial Intelligence

CERTIFICATE

This is to certify that project entitled **“EMOTION DETECTION"** is the bonafied work carried out by **D.SRI DEEKSHITHA, MAHVISH ISHAQ, NASHEER FATIMA** as a Course Project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING** during the academic year 2022-2023 under our guidance and Supervision.

Mr. Tharun Reddy
Asst. Professor,
(CSE),
S R University,
Ananthasagar, Warangal.
Warangal.

Dr. M.Sheshikala
Assoc. Prof. & HOD

S R University,
Ananthasagar,

ACKNOWLEDGEMENT

We express our thanks to Course co-coordinator **Mr. Tharun Reddy, Asst. Prof.** for guiding us from the beginning through the end of the Course Project. We express our gratitude to Head of the department CS&AI, **Dr. M. Sheshikala, Associate Professor** for encouragement, support and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive and encouraging and ultimately drove us to the right direction.

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Dean, School of Computer Science and Artificial Intelligence, **Dr C. V. Guru Rao**, for his continuous support and guidance to complete this project in the institute.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

ABSTRACT

Human emotion detection is implemented in many areas requiring additional security or information about the person. It can be seen as a second step to face detection where we may be required to set up a second layer of security, where along with the face, the emotion is also detected. This can be useful to verify that the person standing in front of the camera is not just a 2-dimensional representation. Another important domain where we see the importance of emotion detection is for business promotions. Most of the businesses thrive on customer responses to all their products and offers. If an artificial intelligent system can capture and identify real time emotions based on user image or video, they can make a decision on whether the customer liked or disliked the product or offer. We have seen that security is the main reason for identifying any person. It can be based on finger-print matching, voice recognition, passwords, retina detection etc. Identifying the intent of the person can also be important to avert threats. This can be helpful in vulnerable areas like airports, concerts and major public gatherings which have seen many breaches in recent years. Human emotions can be classified as: fear, contempt, disgust, anger, surprise, sad, happy, and neutral. These emotions are very subtle. Facial muscle contortions are very minimal and detecting these differences can be very challenging as even a small difference results in different expressions. Also, expressions of different or even the same people might vary for the same emotion, as emotions are hugely context dependent . While we can focus on only those areas of the face which display a maximum of emotions like around the mouth and eyes, how we 2 extract these gestures and categorize them is still an important question. Neural networks and machine learning have been used for these tasks and have obtained good results.

Table of Contents

Chapter No. Title Page No.

1.INTRODUCTION

1.1 OVERVIEW

If someone showed you a picture of a person and asked you to guess what they're feeling, chances are you'd have a pretty good idea about it. What if your computer could do the same? Face detection has been around for ages. Taking a step forward, human emotion displayed by face and felt by brain, captured in either video, electric signal (EEG) or image form can be approximated. Human emotion detection is the need of the hour so that modern artificial intelligent systems can emulate and gauge reactions from face. This can be helpful to make informed decisions be it regarding identification of intent, promotion of offers or security related threats. Recognizing emotions from images or video is a trivial task for human eye, but proves to be very challenging for machines and requires many image processing techniques for feature extraction. Several machine learning algorithms are suitable for this job. Any detection or recognition by machine learning requires training algorithm and then testing them on a suitable dataset. This paper explores a couple of machine learning algorithms as well as feature extraction techniques which would help us in accurate identification of the human emotion.

1.2 PROBLEM STATEMENT

Emotion detection using machine learning is the task of automatically recognizing emotions in text, audio, image, and video data. Emotion detection has numerous applications, including customer service chatbots, sentiment analysis, and personalized marketing. However, accurately detecting emotions in unstructured data can be challenging due to the complexity of human emotions and the variety of ways in which they can be expressed.

The problem statement for emotion detection using machine learning is to build a model that can accurately recognize emotions in different types of data. This model should be able to classify emotions such as joy, sadness, anger, and fear with a high level of accuracy. The model should also be able to generalize to new data and handle the variability of human expression.

To solve this problem, we need to collect and pre process large amounts of labeled data to train the machine learning model. We need to choose an appropriate machine learning algorithm, such as support vector machines, random forests, or deep learning neural networks. We also need to carefully select the features and representations used to capture the relevant information in the data. Finally, we need to evaluate the performance of the model on a validation dataset and fine-tune the model parameters to improve its accuracy. The ultimate goal is to create a reliable and scalable model that can be deployed in real-world applications to improve the accuracy and efficiency of emotion detection.

1.3 EXISTING SYSTEM

There are a few software platforms such as Affectiva in this regard. Even though a particular system exists, the dataset we have worked with has produced more accurate results that we can rely on.

In the existing system, classification is done through simple image processing to classify images only.

- Existing work includes the application of feature extraction of facial expressions with the combination of neural networks for the recognition of different facial emotions (happy, sad, angry, fear, surprised, neutral, etc..).

- Humans are capable of producing thousands of facial actions during communication that varies in complexity, intensity, and meaning. The existing system is capable of analyzing the limitations of the existing system of Emotion recognition using brain activity.

Facial expression recognition systems: These systems use computer vision techniques to analyze facial expressions and detect emotions such as happiness, sadness, anger, and surprise.

Speech recognition systems: These systems analyze speech patterns and detect emotions based on tone, pitch, and other audio cues.

Text analysis systems: These systems analyze written text and detect emotions based on word choice, syntax, and other linguistic features.

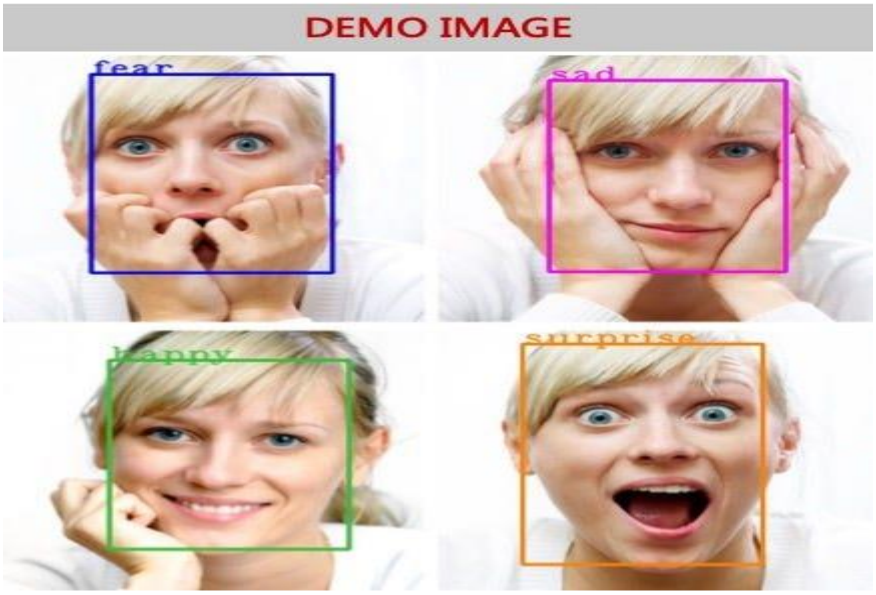
Physiological signals analysis systems: These systems analyze physiological signals such as heart rate, skin conductance, and brain waves to detect emotions.

Multimodal systems: These systems combine multiple modalities, such as facial expressions, speech, and physiological signals, to detect emotions more accurately.

Examples of commercial emotion detection systems include Affectiva, Emotient, Beyond Verbal, and Noldus FaceReader. These systems are used in a variety of applications, including market research, customer service, and mental health. However, there are also concerns about the accuracy and ethical implications of using these systems, particularly in sensitive contexts such as mental health diagnosis or employment screening.

1.4 PROPOSED SYSTEM

The proposed system for this project includes accurate prediction of facial emotions. This system uses classification algorithm models like logistic regression, Decision Tree, KNN Random forest, SVM to predict customer emotions with high accuracy.



1.5 OBJECTIVES

- To predict human facial emotions.
- Deliver high accurate results.
- Present different type of results using different models.
- Select a accurate model and use it for the prediction.
- Deliver a well presented outcome from the ML model.

1.6 ARCHITECTURE

This ML Model uses classification algorithms to solve a problem of accurate detection of human facial emotions. Logistic regression, Decision Tree, KNN Random forest, SVM algorithms are used on the dataset to finish the model in predicting accurate result from the given dataset.

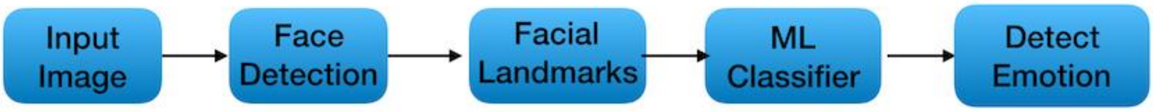
Logistic Regression is a statistical model used for binary classification tasks. It uses a sigmoid function to map input variables to the output probability of a binary class.

Decision Tree is a non-parametric algorithm that can be used for both classification and regression tasks. It works by partitioning the input space into a tree-like structure based on the input features. Each internal node of the tree represents a decision on an input feature, and each leaf node represents a classification or regression value.

KNN is a non-parametric algorithm used for classification tasks. It works by finding the k-nearest data points to the input point and assigning the class label that is most common among the k-nearest neighbours.

Random Forest is an ensemble algorithm that uses multiple decision trees to make a prediction. Each tree in the forest is trained on a random subset of the training data and a random subset of the input features. The final prediction is made by aggregating the predictions of all the trees in the forest.

SVM is a parametric algorithm that can be used for both classification and regression tasks. It works by finding a hyperplane that maximizes the margin between the two classes. SVM is often used when the input space is high-dimensional.



2. LITERATURE SURVEY

2.1.1. Survey done to collect dataset

This survey is done to comprehend the need and prerequisite of the general population, and to do as such, we went through different sites and applications and looked for the fundamental data. Based on these data, we made an audit that helped us get new thoughts and make different arrangements for our task. We reached the decision that there is a need of such application and felt that there is a decent extent of progress in this field.

Images of the people depicting facial emotions have been collected from the web and made together into a csv dataset. Data is collected in raw format and then is analysed and converted into a CSV format file where the data is stored column and row wise so that the ML model will efficiently use it for further predictive activities given by the user.

1) **Human Emotion Detection System Author**: Dilbagh Singh In this paper I am going the purpose a system(by using an existing simulator) which is capable for achieving up to 97 percentage result and easy than Emotion recognition using brain activity system. This purposed system depend upon human face as we know face also reflect the human brain activities or emotion. In this paper I have also tried to use neural network for better result by using a existing system simulator

2) **Emotion Detection From Facial Expression Author:** S. Srivastav, S. Agrawal, V. Saxena The project aims to develops automatic emotion detection system using machine learning algorithm for facial expression recognition.

3) **A Literature review on Emotion Detection Recognition For various Facial Emotion Extraction Author**: G. Kalaivani, S. Sathyapriya, Dr. D. Anita The main aim of this research work is to classify the emotional expression from the mouth region of the human face. As the initial task is to extracted the mouth region from the facial image, a survey on various existing research works to segment the face expression image is reviewed and discussed.

4) **Literature survey on face and face expression recognition Author**: J. Anil, L. Padma Suresh In this paper some of the tailor made face expression Recognition algorithm are presented. This paper also gives a brief insight into a feature extraction method of these face expression recognition

5) **Literature survey on facial expression recognition Techniques Author**: G.Rajeswari, P.IthayaRani In this paper, various technique and databases used in the discussed for facial expression recognition. Many researchers are interested to do research in facial expression. Because by the Quote "One person could be judge by their reaction not by actions" like that. Persons mood can be easily identified by their emotions rather than their words.

3. DATA PREPROCESSING

3.1. DATASET DESCRIPTION

Emotion detection image datasets typically consist of a collection of images, each of which is labelled with the corresponding emotion that is being expressed in the image. The emotions can range from basic emotions like happiness, sadness, anger, and surprise to more complex emotions like disgust, fear, and contempt.

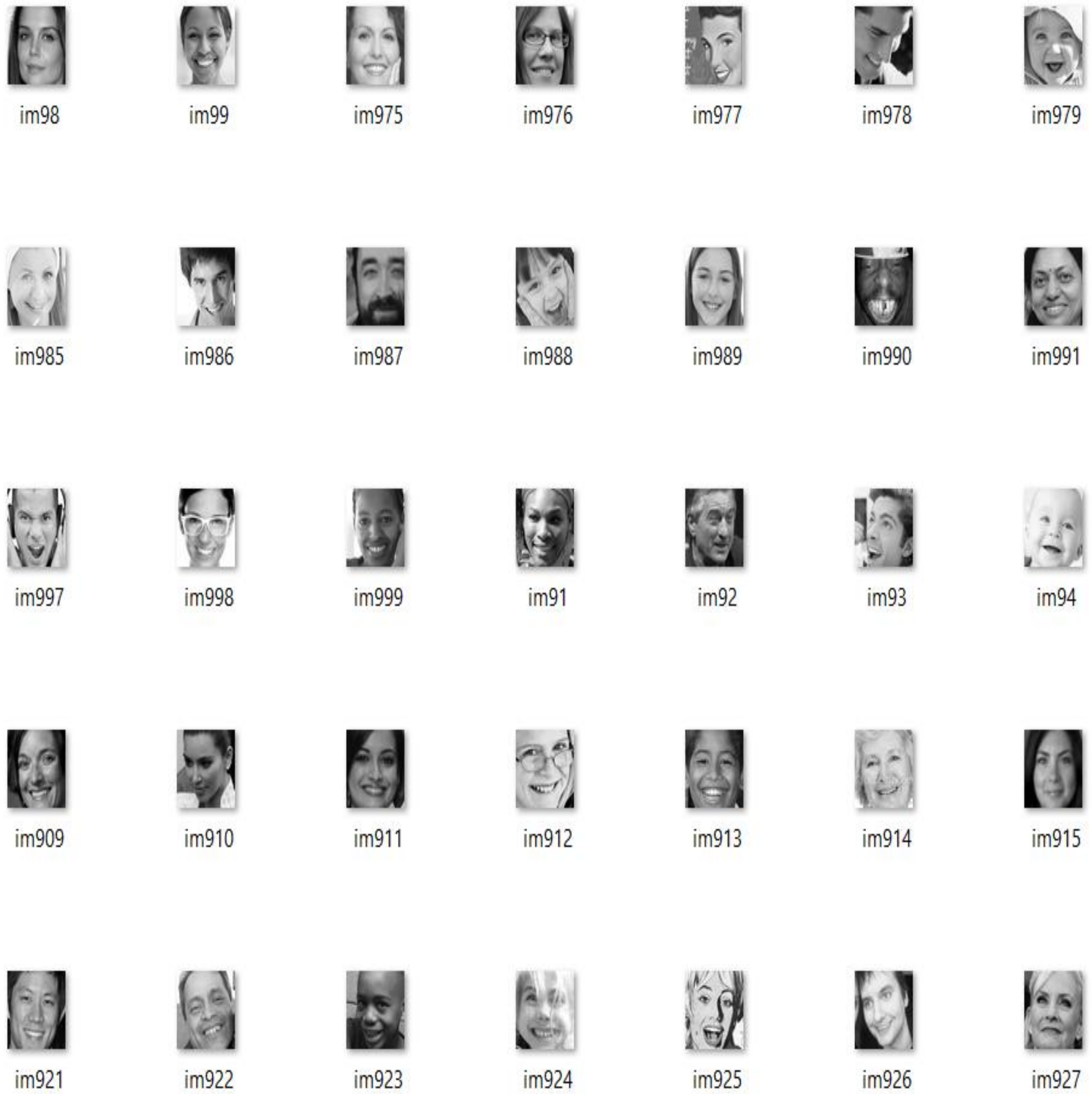
These datasets may also contain additional information such as age, gender, and ethnicity of the person in the image, as these factors can sometimes influence the expression of emotions.

In this project we have worked on two emotions of human being they are: Happy and Sad. In our literature survey we came to know that happy and sad are the most common and popular emotions and critical emotions which shows an ambiguity whether the person is happy or sad by looking at image. Here is the data set we used to train and test the model. We have collected this data set from some online sources like Kaggle.

In this dataset there are two types of images indicates two types of emotions they are happy and sad. The total images in the data set are 4368. In 4368 images there are 2184 images of each emotion.

DATA SET

Happy images:



Sad images:

im98
im99
im979
im980
im981
im982
im983
im984
im985
im986
im987
im988
im989
im990
im991
im992
im993
im994
im995
im996
im997
im998
im999
im92
im93
im94
im95
im96
im97
im910
im911
im912
im913
im914
im915
im916
im917
im918
im919
im920
im921
im922
im923
im924
im925
im926
im927
im928
im929
im930
im931
im932
im933
im934
im935
im936
im937
im938
im939
im940

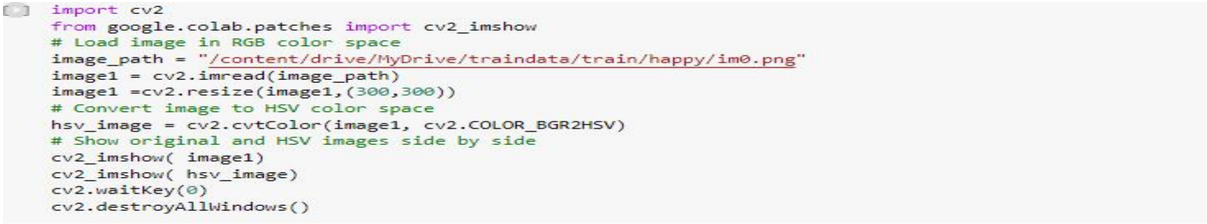
3.3. DATA AUGMENTATION AND TRANSFORMATION

Data augmentation and transformation are techniques used in machine learning and computer vision to artificially increase the size of a dataset by generating new training examples from the existing ones.

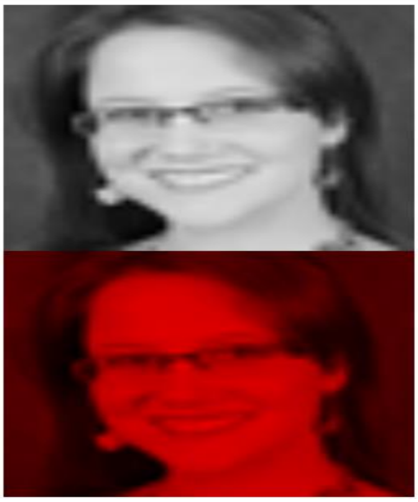
Data augmentation involves applying various transformations to the original data to create new samples, such as random rotations, translations, scaling, and flipping. This helps to increase the diversity of the data and reduce overfitting, thereby improving the model's ability to generalize to new data.

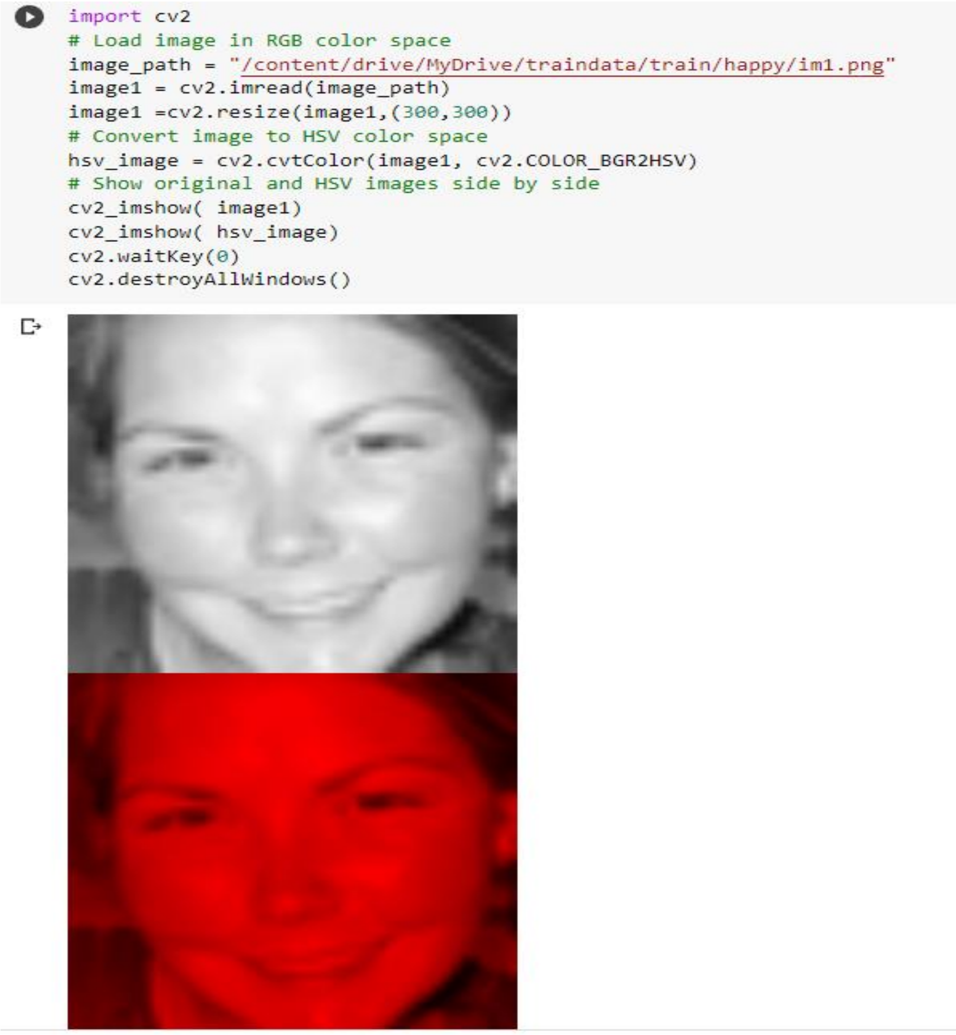
Data transformation, on the other hand, refers to the process of changing the representation or format of the data. This could involve converting images from colour to grayscale, resizing images, or applying filters or other types of image processing techniques. Data transformation can also be used to pre-process the data and make it more suitable for training the model.

Overall, data augmentation and transformation are essential tools in machine learning that enable us to generate more data and improve the performance of our models.



```
import cv2
from google.colab.patches import cv2_imshow
# Load image in RGB color space
image_path = "/content/drive/MyDrive/traindata/train/happy/im0.png"
image1 = cv2.imread(image_path)
image1 =cv2.resize(image1,(300,300))
# Convert image to HSV color space
hsv_image = cv2.cvtColor(image1, cv2.COLOR_BGR2HSV)
# Show original and HSV images side by side
cv2_imshow( image1)
cv2_imshow( hsv_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```





```
import cv2
# Load image in RGB color space
image_path = "/content/drive/MyDrive/traindata/train/happy/im1.png"
image1 = cv2.imread(image_path)
image1 =cv2.resize(image1,(300,300))
# Convert image to HSV color space
hsv_image = cv2.cvtColor(image1, cv2.COLOR_BGR2HSV)
# Show original and HSV images side by side
cv2_imshow( image1)
cv2_imshow( hsv_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Figure 6

4. METHODOLOGY

4.1. PROCEDURE TO SOLVE GIVEN PROBLEM

The procedure for solving a problem related to emotion detection depends on the specific details of the problem at hand. However, here is a general procedure that can be followed:

1.**Define the problem**: Clearly define the problem you are trying to solve. Determine what type of emotions you want to detect, what kind of data you have, and what kind of output you want to generate.

2.**Collect and pre process data**: Collect a suitable dataset of images, text, audio or video samples depending on the problem. Clean the data by removing any irrelevant information, and standardize it by converting to a common format.

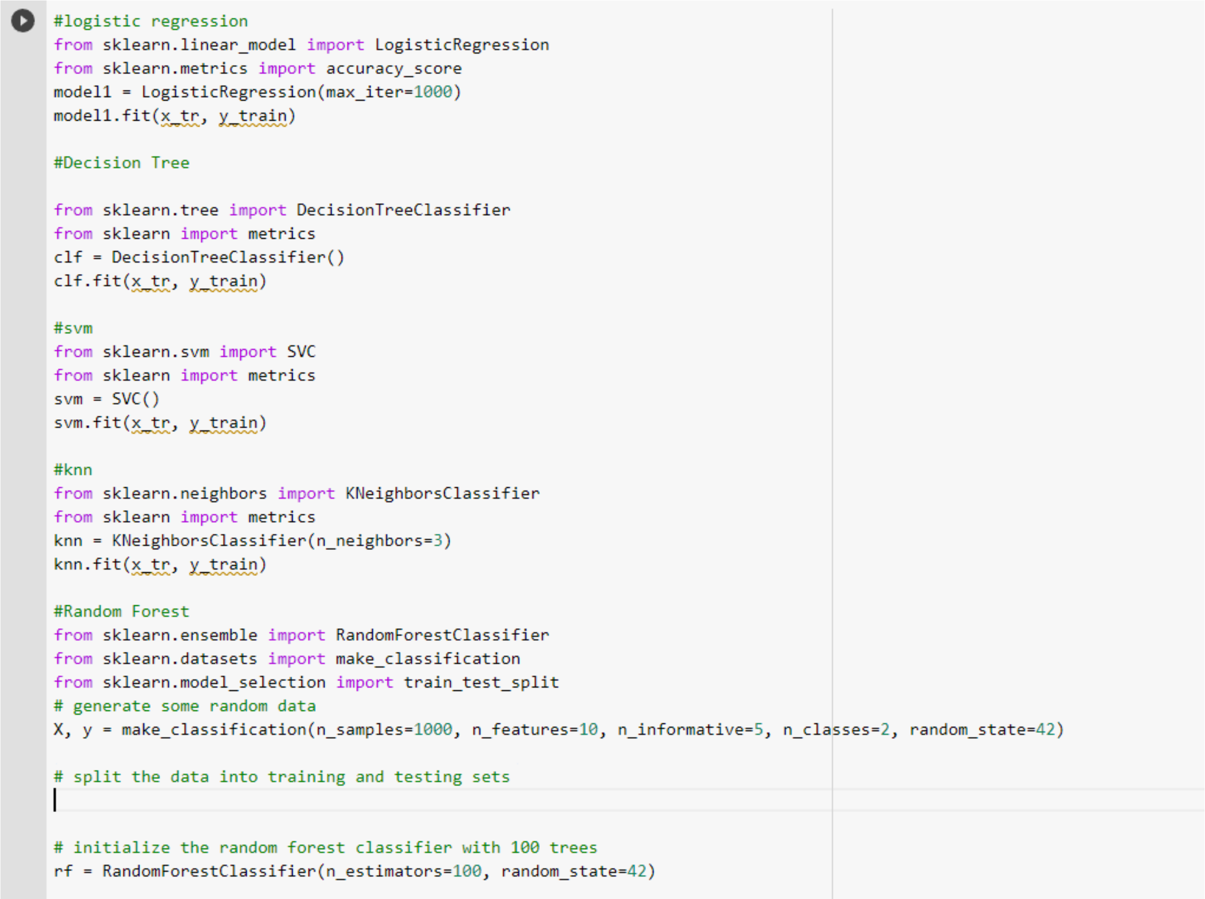
3.**Feature extraction**: Extract relevant features from the data. For instance, in text-based emotion detection, the features might include words, phrases, sentiment scores, and other linguistic features. For audio and video, features could include pitch, tone, facial expressions, body language and so on.

4.**Build a model**: Select a suitable model for the task and train it on the extracted features. Common models used for emotion detection include machine learning algorithms like Naive Bayes, SVM, and decision trees. Deep learning models like Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, and Transformers have also shown success in this area.

5.**Evaluate the model**: Evaluate the performance of the model on a test set of data. Calculate metrics like accuracy, precision, recall, and F1 score to determine the effectiveness of the model.

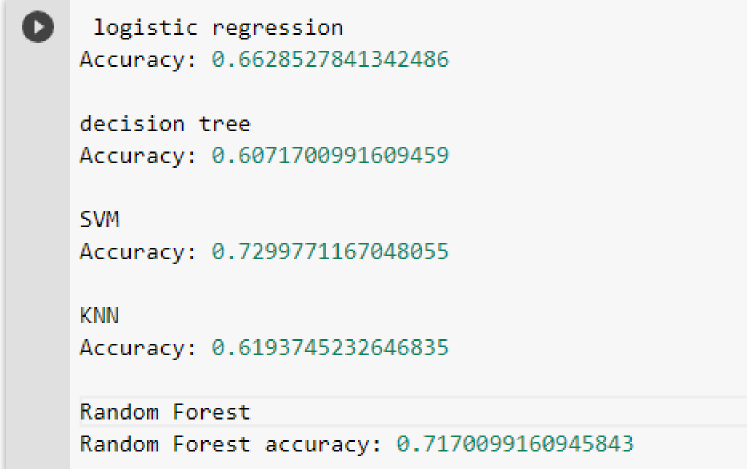
6.**Fine-tune the model**: Based on the results of the evaluation, fine-tune the model by tweaking hyperparameters or modifying the architecture.

Overall, emotion detection is a complex task that requires expertise in several fields such as linguistics, computer science, and psychology. Therefore, it is essential to have a clear understanding of the problem and to collaborate with experts from different fields for successful implementation.



```
#logistic regression
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
model1 = LogisticRegression(max_iter=1000)
model1.fit(x_tr, y_train)

#Decision Tree

from sklearn.tree import DecisionTreeClassifier
from sklearn import metrics
clf = DecisionTreeClassifier()
clf.fit(x_tr, y_train)

#svm
from sklearn.svm import SVC
from sklearn import metrics
svm = SVC()
svm.fit(x_tr, y_train)

#knn
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(x_tr, y_train)

#Random Forest
from sklearn.ensemble import RandomForestClassifier
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
# generate some random data
X, y = make_classification(n_samples=1000, n_features=10, n_informative=5, n_classes=2, random_state=42)

# split the data into training and testing sets


# initialize the random forest classifier with 100 trees
rf = RandomForestClassifier(n_estimators=100, random_state=42)
```

The above figure shows the accuracy of each algorithm used in the model as the dataset is already trained and tested as shown above. Accuracy of each algorithm is shown below to find the best fit model for the outcome that we are expecting.



```
 logistic regression
Accuracy: 0.6628527841342486

decision tree
Accuracy: 0.6071700991609459

SVM
Accuracy: 0.7299771167048055

KNN
Accuracy: 0.6193745232646835

Random Forest
Random Forest accuracy: 0.7170099160945843
```

4.2. MODEL ARCHITECTURE

There are several model architectures that can be used for emotion detection, depending on the type of data and the specific problem at hand. Here are some commonly used model architectures for different types of data:

For image datasets, the architectures of different machine learning models may differ depending on the specific task at hand. However, here are some general architectures for popular machine learning models used for image classification:

Logistic Regression:

Logistic regression can be used for binary classification of images. The input images are first pre-processed and then flattened into a vector. The flattened vector is then fed into the logistic regression model, which consists of an input layer, a fully connected layer, and a sigmoid activation function.

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.

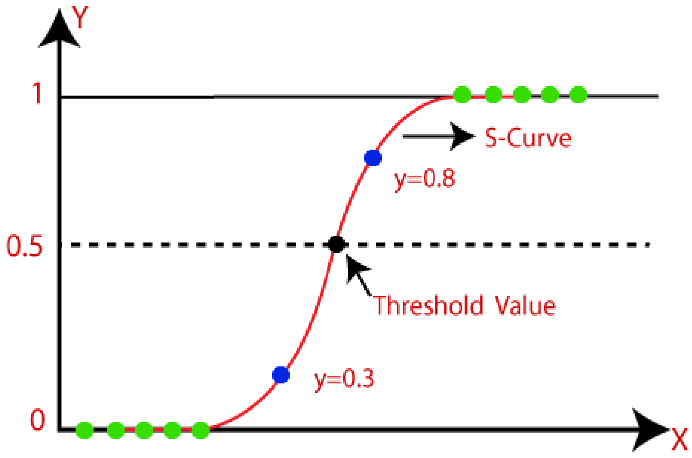
In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.

Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

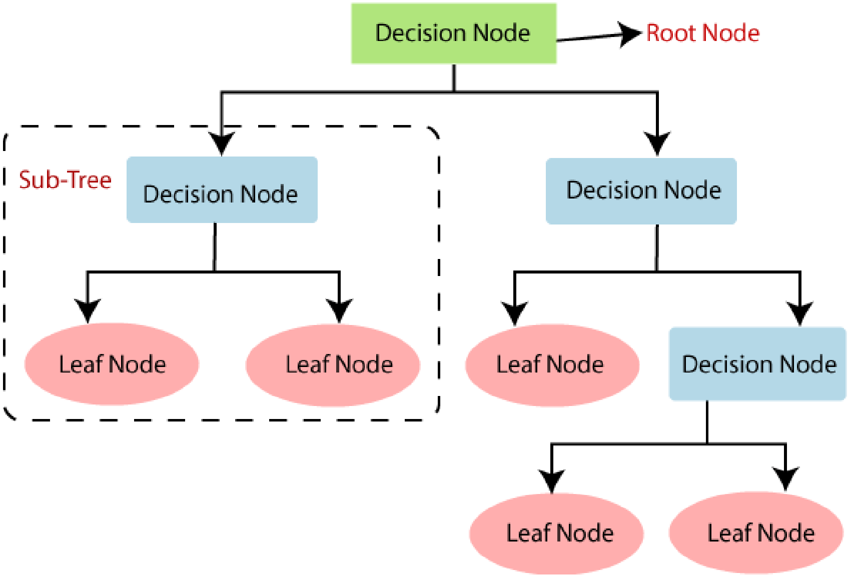
Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:

Logistic Regression in Machine Learning



Decision Tree: A decision tree is a tree-like model that uses a series of binary decisions to classify images. In image classification, a decision tree can be constructed by selecting image features and splitting the dataset based on the selected features. Each node in the decision tree represents a decision, and each leaf node represents a class label.

- Decision Tree is a **Supervised learning technique** that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.**
- In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node.** Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.
- The decisions or the test are performed on the basis of features of the given dataset.
- **It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.**
- It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.
- In order to build a tree, we use the **CART algorithm,** which stands for **Classification and Regression Tree algorithm.**
- A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.
- Below diagram explains the general structure of a decision tree:



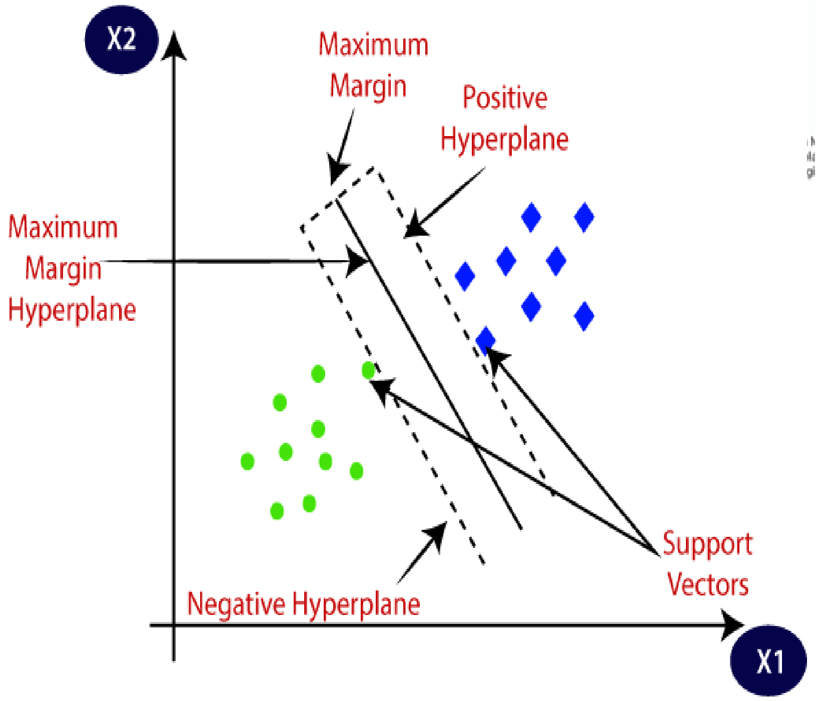
Support Vector Machine (SVM):

An SVM is a powerful model for image classification. The architecture of an SVM for image classification involves selecting appropriate features such as colour histograms or texture features, and then training the SVM model on the feature vectors. The SVM model consists of a kernel function, a decision boundary, and support vectors.

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

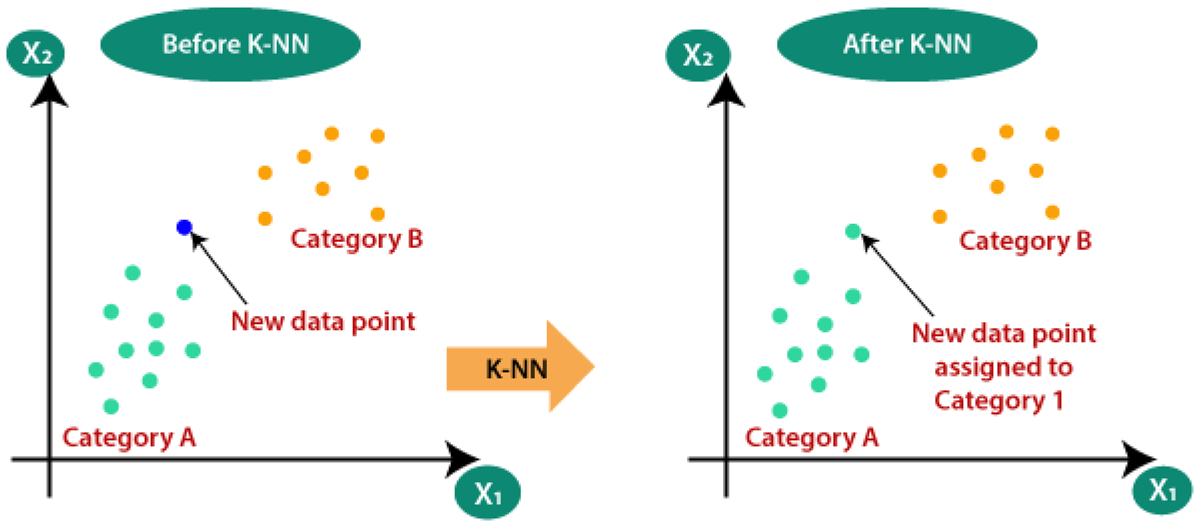
SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



K-Nearest Neighbours (KNN):

In KNN, the model makes predictions based on the k-nearest neighbours to the input image. In the case of image classification, KNN can be used by extracting feature vectors for each image and computing the distances between the input image and the other images in the dataset. The k-nearest neighbours to the input image are then used to make a prediction.

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.



Random Forest:

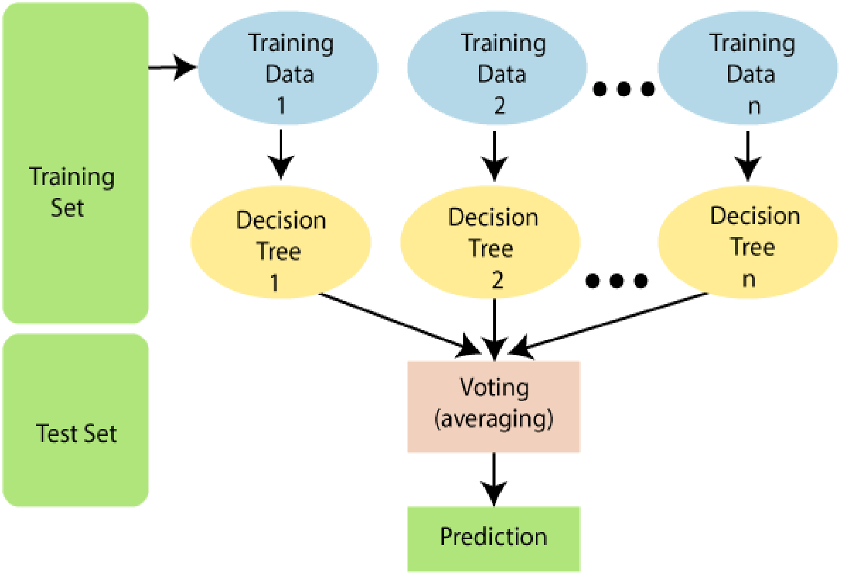
Random forest is an ensemble model that combines multiple decision trees to improve accuracy. In image classification, a random forest model can be trained by constructing multiple decision trees on different subsets of the input features and samples. The output of each decision tree is then combined to make a final prediction.

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

The below diagram explains the working of the Random Forest algorithm:



When it comes to emotion detection using images, a convolutional neural network (CNN) is a common choice for the model architecture. Here is a basic model architecture for emotion detection using images:

Input Layer: The input layer receives the image data and applies pre-processing steps such as normalization, resizing, and data augmentation.

Convolutional Layers: A series of convolutional layers are used to extract relevant features from the image data. Each convolutional layer applies a set of learnable filters to the input image to produce a set of feature maps.

Activation Layers: Activation layers such as ReLU are applied after each convolutional layer to introduce non-linearity and improve the model's ability to capture complex patterns in the data.

Pooling Layers: Pooling layers such as Max Pooling are used to down sample the feature maps and reduce the dimensionality of the data.

Flatten Layer: The output from the last convolutional layer is flattened into a one-dimensional vector.

Dense Layers: A series of fully connected dense layers are used to classify the emotion. The output from the last dense layer is passed through a soft max activation function to produce a probability distribution over the different emotion classes.

Output Layer: The output layer produces the predicted emotion class based on the highest probability in the soft max distribution.

4.3. SOFTWARE DESCRIPTION

- **Google Colab – Python Notebook**

Colab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks that you create can be simultaneously edited by your team members - just the way you edit documents in Google Docs. Colab supports many popular machine learning libraries which can be easily loaded in your notebook.

- **Python Programming Language and Its Libraries**

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library.

Python Libraries like **NumPy, Pandas**, **OS, sklearn, seaborn** are extensively used in the process of completing the ML Model.

Pandas: Pandas are an important library for data scientists. It is an open-source machine learning library that provides flexible high-level data structures and a variety of analysis tools. It eases data analysis, data manipulation, and cleaning of data

Numpy: The name “Numpy” stands for “Numerical Python”. It is the commonly used library. It is a popular machine learning library that supports large matrices and multi-dimensional data.

Scikit-learn or sklearn : It is a famous Python library to work with complex data. Scikit-learn is an open-source library that supports machine learning. It supports variously supervised and unsupervised algorithms like linear regression, classification, clustering, etc.

OS : This library is a built-in Python library that provides a way to interact with the operating system. It allows you to perform various tasks related to handling files and directories, managing data, and configuring the environment.

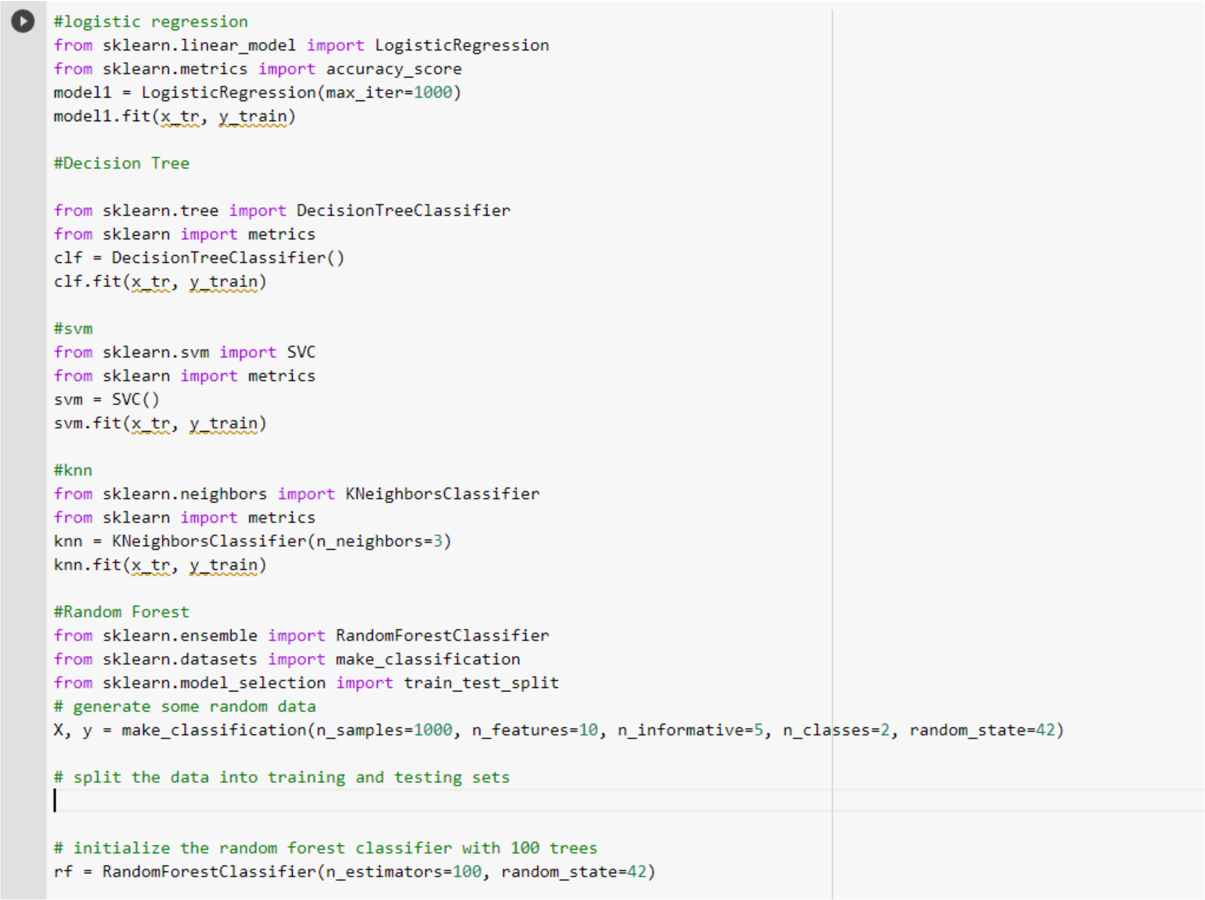
Code:

Importing the libraries and resizing the images and converting them into numpy array.

```
import cv2
import numpy as np
import os
#total train images= 4368

labels =['happy','sad']
img_size=200
data=[]

def get_training_data(data_dir):
  tz=0
  for label in labels:
    path=os.path.join(data_dir,label)
   # print(path)
    class_num=labels.index(label)
    #print(class_num)
    for img in os.listdir(path):
      tz=tz+1
      try:

img_arr=cv2.imread(os.path.join(path,img),cv2.COLOR_BAYER_GB2RGB)
        resized_arr=cv2.resize(img_arr,(img_size,img_size))
        data.append([resized_arr,class_num])
      except Exception as e:
        print(e)
  #returnnp.array(class_num)
  print(tz)
  return np.array(data)
train=get_training_data('/content/drive/MyDrive/traindata/train')
#data
```

5. RESULTS AND DISCUSSION

Dataset is trained with 4 kinds of Classification algorithms as shown in the figure and final accuracy of each of the algorithm

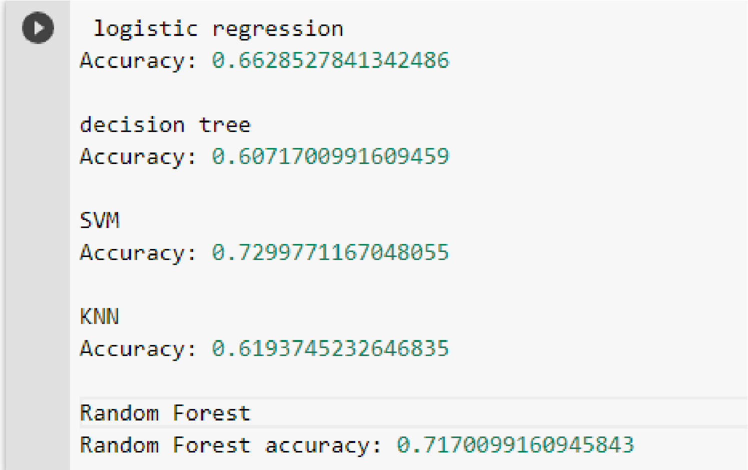


```
#logistic regression
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
model1 = LogisticRegression(max_iter=1000)
model1.fit(x_tr, y_train)

#Decision Tree

from sklearn.tree import DecisionTreeClassifier
from sklearn import metrics
clf = DecisionTreeClassifier()
clf.fit(x_tr, y_train)

#svm
from sklearn.svm import SVC
from sklearn import metrics
svm = SVC()
svm.fit(x_tr, y_train)

#knn
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(x_tr, y_train)

#Random Forest
from sklearn.ensemble import RandomForestClassifier
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
# generate some random data
X, y = make_classification(n_samples=1000, n_features=10, n_informative=5, n_classes=2, random_state=42)

# split the data into training and testing sets


# initialize the random forest classifier with 100 trees
rf = RandomForestClassifier(n_estimators=100, random_state=42)
```

From the above figure, we can conclude that Random Forest has high accuracy scores compared to all other algorithms in the given data set ml model. But we must not conclude the best usage of an algorithm just from accuracy of its prediction. We should also take into consideration of some points like usage of data set for a particular purpose so that we can conclude a machine learning model by use case of the dataset.

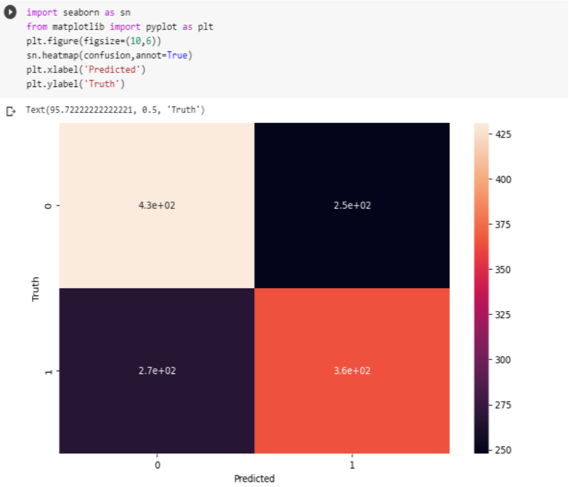
.

We may make some other considerations and finalise the result of the model that we are going to predict.



```
 logistic regression
Accuracy: 0.6628527841342486

decision tree
Accuracy: 0.6071700991609459

SVM
Accuracy: 0.7299771167048055

KNN
Accuracy: 0.6193745232646835

Random Forest
Random Forest accuracy: 0.7170099160945843
```

Confusion matrix:



```
import seaborn as sn
from matplotlib import pyplot as plt
plt.figure(figsize=(10,6))
sn.heatmap(confusion,annot=True)
plt.xlabel('Predicted')
plt.ylabel('Truth')
```

```
Text(95.7222222222221, 0.5, 'Truth')
```

Support Vector Machine is the best model as it accuracy is 72% when we trained and test the data. Out of each model we used Support Vector Machine has more accuracy so, we can say that SVM is the optimal model for the data.

6. CONCLUSION AND FUTURE SCOPE

Emotion detection is a rapidly growing field with numerous applications in areas such as mental health, customer service, and entertainment. With advances in machine learning and artificial intelligence, the accuracy of emotion detection models has improved significantly, making it a promising area for further research and development.

In conclusion, the ability to accurately detect emotions from text, audio, and video data has the potential to revolutionize many industries and improve the way we interact with technology. While there is still room for improvement in terms of accuracy and reliability, the future of emotion detection looks bright, and we can expect to see more innovative applications in the coming years.

Some potential areas for future research and development in emotion detection include:

Multi-modal emotion detection, Real-time emotion detection, Cross-cultural emotion detection Privacy and ethical concerns.

Overall, the future of emotion detection is promising, and we can expect to see more innovative applications that will enhance our understanding of human emotions and improve our interactions with technology.

7. REFERENCES

- https://www.kaggle.com/datasets/ananthu017/emotion-detection-fe
 - dataset import from Kaggle where legit datasets are found to train and test our models.
- https://www.javatpoint.com/classification-algorithm-in-machine-learning - Classification algorithmic related queries solved through this website.