# MACHINE LEARNING

(Gender Detection and Age Prediction)

*Summer Internship Report Submitted in partial fulfillment of the requirement for undergraduate degree of*

## Bachelor of Technology

In

## Computer Science Engineering

By

## Joshi Aniruddha

## 221710308021

*Under the Guidance of* **Mr.**

Assistant Professor



Department Of Computer Science and Engineering

GITAM School of Technology

GITAM (Deemed to be University) Hyderabad-502329

June 2020

i

# DECLARATION

I submit this industrial training work entitled **“PREDICTING THE AGE AND GENDER USING FACE IMAGES**” to GITAM (Deemed To Be University), Hyderabad in partial fulfillment of the requirements for the award of the degree of “**Bachelor of Technology**” in “**Computer Science Engineering**”. I declare that it was carried out independently by me under the guidance of **Mr.** Asst. Professor, GITAM (Deemed To Be University), Hyderabad, India.

The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma.

Place: HYDERABAD Joshi Aniruddha

Date: 221710308021

ii

GITAM (DEEMED TO BE UNIVERSITY)

Hyderabad-502329, India Dated:

**C ERTIFICATE**

This is to certify that the Industrial Training Report entitled **“PREDICTING THE AGE AND GENDER USING FACE IMAGES”** is being submitted by Joshi Aniruddha (221710308021) in partial fulfillment of the requirement for the award of **Bachelor of Technology in & Computer Science Engineering** at GITAM (Deemed To Be University), Hyderabad during the academic year 2019-20

It is faithful record work carried out by her at the **Computer Science and Engineering Department**, GITAM University Hyderabad Campus under my guidance and supervision.

### Mr. Dr.Phani Kumar

Assistant Professor Professor and HOD

Department of CSE Department of CSE

**ACKNOWLEDGEMENT**

Apart from my effort, the success of this internship largely depends on the encouragement and guidance of many others. I take this opportunity to express my gratitude to the people who have helped me in the successful competition of this internship.

I would like to thank respected **Dr. N. Siva Prasad,** Pro Vice Chancellor, GITAM Hyderabad and **Dr. CH. Sanjay,** Principal, GITAM Hyderabad

I would like to thank respected **Dr. Phani Kumar,** Head of the Department of ComputerScience and Engineering for giving me such a wonderful opportunity to expand my knowledge for my own branch and giving me guidelines to present a internship report. It helped me a lot to realize of what we study for.

I would like to thank the respected faculties **Mr.** who helped me to make this internship a successful accomplishment.

I would also like to thank my friends who helped me to make my work more organized and well-stacked till the end.

Joshi Aniruddha

221710308021

## ABSTRACT

Machine learning algorithms are used to predict the values from the data set by splitting the data set in to train and test and building Machine learning algorithms models of higher accuracy to predict the age and gender is the primary task to be performed on face images set My perception of understanding the given data set has been in the view of undertaking a client’s requirement of overcoming the prediction analysis of age and gender.

In the modern world the age and gender prediction is most needed for computers in various aspects.Such as in validation of the clients working in a company.It is very difficult for an individual to check the gender and age for a mass group of people.In those situatuion a system which consist of prediction of age and gender is neccessary to save the time and work.Google,Instagram and other AI detectors of face and gender feature uses the machine learning algortihms and build the model.Then they deploy the machine learning algorithms to process the images.

1. **MACHINE LEARNING ……………………………………………………………………….4**
   1. INTRODUCTION TO MACHINE LEARNING**……………………………………………4**
   2. IMPORTANCE OF MACHINE LEARNING**………………………………………………4**
   3. USES OF MACHINE LEARNING**…………………………………………………………6**
   4. TYPES OF LEARNING ALGORITHMS**…………………………………………………..7**
      1. SUPERVISED LEARNING**………………………………………………………..7**
      2. UNSUPERVISED LEARNING**…………………………………………………….9**
      3. SEMI SUPERVISED LEARNING**………………………………………………...10**
   5. RELATION BETWEEN DATA MINING, ML AND DEEP LEARNING**………………10**
2. **DEEP LEARNING……………………………………………………………………………11**

2.1 KEY TAKEAWAYS**……………………………………………………………………….11**

2.2 HOW DEEP LEARNING WORKS**………………………………………………………..11**

2.3 DEEP LEARNING VS MACHINE LEARNING**………………………………………….11**

2.4 USES**………………………………………………………………………………………..12**

2.5 RELATION BETWEEN DATA MINING, ML AND DEEP LEARNING**……………….12**

2.6 ALGORITHMS IN DEEP LEARNING**…………………………………………………...13**

**3.** **PYTHON…………………………………………………………………………..………..15**

3.1 INTRODUCTION TO PYTHON**……………………………………………………………15**

3.2 HISTORY OF PYTHON**…………………………………………………………………….15**

3.3 FEATURES OF PYTHON**…………………………………………………………………..15**

3.4 HOW TO SETUP PYTHON**………………………………….……………………………..16**

3.4.1 INSTALLATION(USING PYTHON IDLE)**…………………………………………16**

3.4.2 INSTALLATION(USING ANACONDA)**……………………………………………17**

3.5 PYTHON VARIABLE TYPES**……………………………………………………………...19**

3.5.1 PYTHON NUMBERS**…………………………………………………………………20**

3.5.2 PYTHON STRINGS**…………………………………………………………………..20**

3.5.3 PYTHON LISTS**………………………………………………………………………20**

3.5.4 PYTHON TUPLES**……………………………………………………………………21**

3.6 PYTHON FUNCTION**……………………………………………………………………...22**

3.6.1 DEFINING A FUNCTION**…………………………………………………………..22**

3.6.2 CALLING A FUNCTION**……………………………………………………………23**

3.7 PYTHON USING OOP’S CONCEPTS**…………………………………………………….23**

3.7.1 INIT METHOD IN CLASS**……………………………………………………………23**

**4. GENDER DETECTION AND AGE PREDICTION……………………………………….....24**

4.1 PROBLEM STATEMENT**…………………………………………………………………..25**

4.2 OBJECTIVE OF THE CASE STUDY**………………………………………………………25**

4.3 PRE-PREOCESSING THE DATA**………………………………………………………….25**

4.3.1 GETTING THE FACE IMAGES**……………………………………………………..25**

4.3.2 IMPORTING THE REQUIRED PACKAGES**……………………………………….25**

4.3.3 DEFINING A FUNCTION TO DETECT THE FACES**……………………………..25**

4.3.4 SETTING THE INPUT VALUE FOR NETWORK**…………………………………26**

4.3.5 DETECTION OF FACES AND THE CONFIDENCE**………………………………26**

4.3.6 DRAWING THE RECTANGLE AROUND THE DETECTED FACE**……………..27**

4.3.7 TAKING THE INPUT FROM USER BY ARGPARSE**……………………………..27**

4.4 MODEL BUILDING AND NETWORK ARCHITECTURE**………………………………27**

4.4.1 MODEL MEAN VALUES,AGE LIST AND GENDER LIST**………………………28**

4.4.2 LOADING AND READING THE NETWORK**……………………………………..28**

4.5 READING THE INPUT**……………………………………………………………………29**

4.6 EXTRACTING THE REGION OF INTEREST(ROI) AND CONSTRUCTING A BLOB

FOR ROI**……………………………………………………………………………………29**

4.6.1 PREDICTING THE GENDER**………………………………………………………30**

4.6.2 PREDICTING THE AGE**……………………………………………………………30**

4.7 PRINTING THE IMAGE,GENDER,AGE AND TIME**…………………………………..31**

4.7.1 OUTPUT FOR A SAMPLE FACE IMAGE**………………………………………..31**

**5. CONCLUSION………………………………………………………………………………....32**

**6.** **REFERENCES………………………………………………………………………………….32**

**MACHINE LEARNING AND DEEP LEARNING**

## INTRODUCTION TO MACHINE LEARNING:

Machine Learning(ML) is the scientific study of algorithms and statistical models that computer systems use in order to perform a specific task effectively without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of Artificial Intelligence(AI).

## IMPORTANCE OF MACHINE LEARNING:

Consider some of the instances where machine learning is applied: the self-driving Google car, cyber fraud detection, online recommendation engines—like friend suggestions on Facebook, Netflix showcasing the movies and shows you might like, and “more items to consider” and “get yourself a little something” on Amazon—are all examples of applied machine learning. All these examples echo the vital role machine learning has begun to take in today’s data-rich world.

Machines can aid in filtering useful pieces of information that help in major advancements, and we are already seeing how this technology is being implemented in a wide variety of industries.

With the constant evolution of the field, there has been a subsequent rise in the uses, demands, and importance of machine learning. Big data has become quite a buzzword in the last few years; that’s in part due to increased sophistication of machine learning, which helps analyze those big chunks of big data. Machine learning has also changed the way data extraction, and interpretation is done by involving automatic sets of generic methods that have replaced traditional statistical techniques.

The process flow depicted here represents how machine learning works

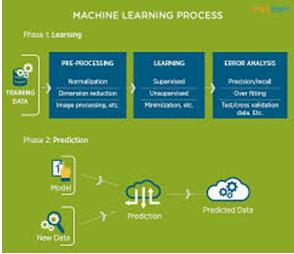


Figure 1 : The Process Flow

## USES OF MACHINE LEARNING:

Earlier in this article, we mentioned some applications of machine learning. To understand the concept of machine learning better, let’s consider some more examples: web search results, real-time ads on web pages and mobile devices, email spam filtering, network intrusion detection, and pattern and image recognition. All these are by-products of applying machine learning to analyze huge volumes of data

Traditionally, data analysis was always being characterized by trial and error, an approach that becomes impossible when data sets are large and heterogeneous. Machine learning comes as the solution to all this chaos by proposing clever alternatives to analyzing huge volumes of data.

By developing fast and efficient algorithms and data-driven models for real-time processing of data, machine learning can produce accurate results and analysis.

## TYPES OF LEARNING ALGORITHMS:

The types of machine learning algorithms differ in their approach, the type of data they input and output, and the type of task or problem that they are intended to solve.

## Supervised Learning :

When an algorithm learns from example data and associated target responses that can consist of numeric values or string labels, such as classes or tags, in order to later predict the correct response when posed with new examples comes under the category of supervised learning.

Supervised machine learning algorithms uncover insights, patterns, and relationships from a labelled training dataset – that is, a dataset that already contains a known value for the target variable for each record. Because you provide the machine learning algorithm with the correct answers for a problem during training, it is able to “learn” how the rest of the features relate to the target, enabling you to uncover insights and make predictions about future outcomes based on historical data.

Examples of Supervised Machine Learning Techniques are Regression, in which the algorithm returns a numerical target for each example, such as how much revenue will be generated from a new marketing campaign.

Classification, in which the algorithm attempts to label each example by choosing between two or more different classes. Choosing between two classes is called binary classification, such as determining whether or not someone will default on a loan. Choosing between more than two classes is referred to as multiclass classification.

## Unsupervised Learning:

When an algorithm learns from plain examples without any associated response, leaving to the algorithm to determine the data patterns on its own. This type of algorithm tends to restructure the data into something else, such as new features that may represent a class or a new series of uncorrelated values. They are quite useful in providing humans with insights into the meaning of data and new useful inputs to supervised machine learning algorithms.

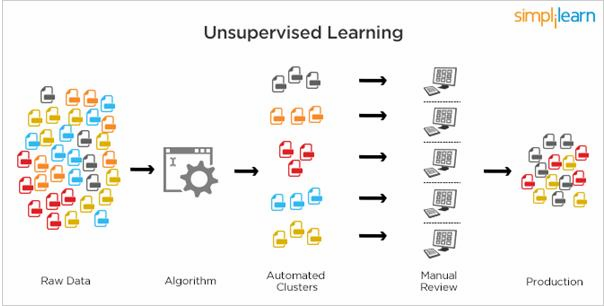


Figure 2 : Unsupervised Learning

Popular techniques where unsupervised learning is used also include self-organizing maps, nearest neighbor mapping, singular value decomposition, and k-means clustering. Basically, online recommendations, identification of data outliers, and segment text topics are all examples of unsupervised learning.

## Semi Supervised Learning:

As the name suggests, semi-supervised learning is a bit of both supervised and unsupervised learning and uses both labeled and unlabeled data for training. In a typical scenario, the algorithm would use a small amount of labeled data with a large amount of unlabeled data.

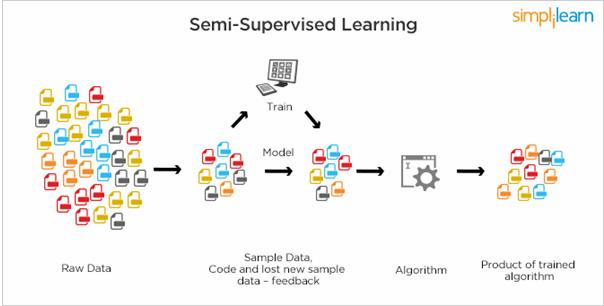


Figure 3 : Semi Supervised Learning

## RELATION BETWEEN DATA MINING,MACHINE LEARNING AND DEEP LEARNING:

Machine learning and data mining use the same algorithms and techniques as data mining, except the kinds of predictions vary. While data mining discovers previously unknown patterns and knowledge, machine learning reproduces known patterns and knowledge—and further automatically applies that information to data, decision-making, and actions.

Deep learning, on the other hand, uses advanced computing power and special

types of neural networks and applies them to large amounts of data to learn, understand, and identify complicated patterns. Automatic language translation and medical diagnoses are examples of deep learning.

**2**. **DEEP LEARNING**

Deep learning is an artificial intelligence function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence (AI) that has networks capable of learning unsupervised from data that is unstructured or unlabeled. Also known as deep neural learning or deep neural network.

**2.1 KEY TAKEAWAYS**

Deep learning is an AI function that mimics the workings of the human brain in processing data for use in detecting objects, recognizing speech, translating languages, and making decisions.

Deep learning AI is able to learn without human supervision, drawing from data that is both unstructured and unlabeled.

Deep learning, a form of machine learning, can be used to help detect fraud or money laundering, among other functions.

**2.2 How Deep Learning Works**

Deep learning has evolved hand-in-hand with the digital era, which has brought about an explosion of data in all forms and from every region of the world. This data, known simply as big data, is drawn from sources like social media, internet search engines, e-commerce platforms, and online cinemas, among others. This enormous amount of data is readily accessible and can be shared through fintech applications like cloud computing.

However, the data, which normally is unstructured, is so vast that it could take decades for humans to comprehend it and extract relevant information. Companies realize the incredible potential that can result from unraveling this wealth of information and are increasingly adapting to AI systems for automated support.

**2.3 Deep Learning vs. Machine Learning**

One of the most common AI techniques used for processing big data is machine learning, a self-adaptive algorithm that gets increasingly better analysis and patterns with experience or with newly added data.

If a digital payments company wanted to detect the occurrence or potential for fraud in its system, it could employ machine learning tools for this purpose. The computational algorithm built into a computer model will process all transactions happening on the digital platform, find patterns in the data set, and point out any anomaly detected by the pattern.

Deep learning, a subset of machine learning, utilizes a hierarchical level of artificial neural networks to carry out the process of machine learning. The artificial neural networks are built like the human brain, with neuron nodes connected together like a web. While traditional programs build analysis with data in a linear way, the hierarchical function of deep learning systems enables machines to process data with a nonlinear approach.

**2.4 Uses:**

Deep learning models are widely used in extracting high-level abstract features, providing improved performance over the traditional models, increasing interpretability and also for understanding and processing biological data. To predict splicing action of exons, a fully connected feedforward neural network was designed by Xiong et al. [60]. In recent years, CNNs were applied on the DNA dataset directly without the requirement of defining features a priori [2], [44]. Compared to a fully connected network, CNNs use less parameters by applying a convolution operation on the input data space and also parameters are shared between the regions. Hence, large DNA sequence data can be trained using these models and also improved pattern detection accuracy can be obtained. Deepbind, a deep architecture based on CNNs, was proposed by Alipanathi et al. [57], which predicts specificities of DNA and RNA binding proteins. CNNs were also used for predicting chromatin marks from a DNA sequence [44]. Angermueller et al. [35] have incorporated CNNs for predicting DNA methylation states. Like CNNs, other deep architectures were also applied for extracting features from raw DNA sequence data and for processing the data.

**2.5 RELATION BETWEEN DATA MINING,MACHINE LEARNING AND DEEP LEARNING:**

Machine learning and data mining use the same algorithms and techniques as data

mining, except the kinds of predictions vary. While data mining discovers previously

unknown patterns and knowledge, machine learning reproduces known patterns and

knowledge—and further automatically applies that information to data, decision-making, and

actions.

Deep learning, on the other hand, uses advanced computing power and special

5

types of neural networks and applies them to large amounts of data to learn, understand, and

identify complicated patterns. Automatic language translation and medical diagnoses are

examples of deep learning.

**2.6 Algorithms in Deep Learning**

Deep Learning is a new area of Machine Learning research, which has been introduced with the objective of moving Machine Learning closer to one of its original goals: Artificial Intelligence. See these course notes for a brief intro of ml of ai and an introduction to deep learning algorithms

Deep Learning is about learning multiple levels of representation and abstraction that help to make sense of data such as images, sound, and text. For more about deep learning algorithms, see for example:

THINGS presented here will introduce you to some of the most important deep learning algorithms and will also show you how to run them using theano. Theano is a python library that makes writing deep learning models easy, and gives the option of training them on a GPU.

The algorithm tutorials have some prerequisites. You should know some python, and be familiar with numpy. Since this tutorial is about using Theano.

The purely supervised learning algorithms are meant to be read in order:

1. Logistic Regression - using Theano for something simple
2. Multilayer percepton - introduction to layers
3. Deep Convolution Network- a simplified version of LeNet5

The unsupervised and semi-supervised learning algorithms can be read in any order (the auto-encoders can be read independently of the RBM/DBN thread):

* [Auto Encoders, Denoising Autoencoders](http://deeplearning.net/tutorial/dA.html#daa) - description of autoencoders
* [Stacked Denoising Auto-Encoders](http://deeplearning.net/tutorial/SdA.html#sda) - easy steps into unsupervised pre-training for deep nets
* [Restricted Boltzmann Machines](http://deeplearning.net/tutorial/rbm.html#rbm) - single layer generative RBM model
* [Deep Belief Networks](http://deeplearning.net/tutorial/DBN.html#dbn) - unsupervised generative pre-training of stacked RBMs followed by supervised fine-tuning

Building towards including the mcRBM model, we have a new tutorial on sampling from energy models:

* [HMC Sampling](http://deeplearning.net/tutorial/hmc.html#hmc) - hybrid (aka Hamiltonian) Monte-Carlo sampling with scan()

Building towards including the Contractive auto-encoders tutorial, we have the code for now:

* [Contractive auto-encoders](https://github.com/lisa-lab/DeepLearningTutorials/blob/master/code/cA.py) code - There is some basic doc in the code.

Recurrent neural networks with word embeddings and context window:

* [Semantic Parsing of Speech using Recurrent Net](http://deeplearning.net/tutorial/rnnslu.html#rnnslu)

LSTM network for sentiment analysis:

* [LSTM network](http://deeplearning.net/tutorial/lstm.html#lstm)

Energy-based recurrent neural network (RNN-RBM):

* [Modeling and generating sequences of polyphonic music](http://deeplearning.net/tutorial/rnnrbm.html#rnnrbm)

Segmentation for medical imagery (meant to be read in order):

* [Fully Convolutional Networks (FCN) for 2D segmentation](http://deeplearning.net/tutorial/fcn_2D_segm.html#fcn-2d-segm)
* [U-Net](http://deeplearning.net/tutorial/unet.html#unet)
* 1D segmentation

# PYTHON

Basic programming language used for machine learning is : PYTHON

## 3.1 INTRODUCTION TO PYHTON:

* + - Python is a high-level, interpreted, interactive and object-oriented scripting language.
    - Python is a general purpose programming language that is often applied in scripting roles
    - Python is Interpreted: Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is like PERL and PHP.
    - Python is Interactive: You can sit at a Python prompt and interact with the interpreter directly to write your programs.
    - Python is Object-Oriented: Python supports the Object-Oriented style or technique of programming that encapsulates code within objects.

## 3.2 HISTORY OF PYTHON:

* + - Python was developed by GUIDO VAN ROSSUM in early 1990’s
    - Its latest version is 3.7 , it is generally called as python3

## 3.3 FEATURES OF PYTHON:

* + - Easy-to-learn: Python has few keywords, simple structure, and a clearly defined syntax,

This allows the student to pick up the language quickly.

* + - Easy-to-read: Python code is more clearly defined and visible to the eyes.
    - Easy-to-maintain: Python's source code is fairly easy-to-maintaining.
    - A broad standard library: Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
    - Portable: Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
    - Extendable: You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
    - Databases: Python provides interfaces to all major commercial databases.
* GUI Programming: Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

## 3.4 HOW TO SETUP PYTHON:

* + - Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.
    - The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python.

## 3.4.1 Installation(using python IDLE):

* + - * Installing python is generally easy, and nowadays many Linux and Mac OS distributions include a recent python.
      * [Download python from www.python.org](http://www.python.org/)
      * When the download is completed, double click the file and follow the instructions to install it.
      * When python is installed, a program called IDLE is also installed along with it. It provides a graphical user interface to work with python.

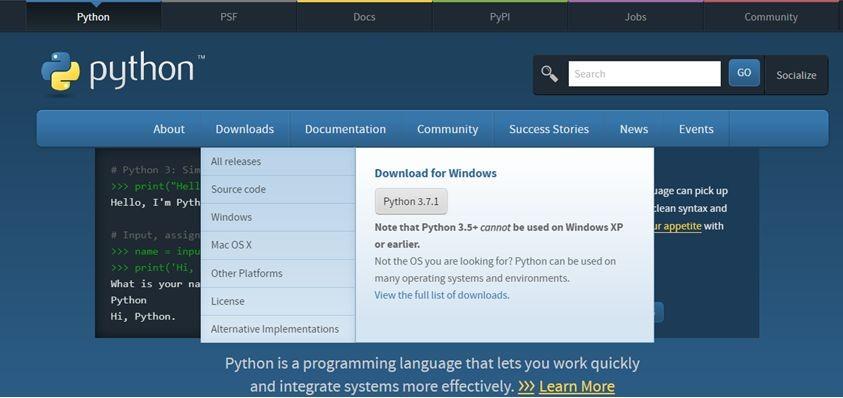


Figure 4 : Python download

## 3.4.2Installation(using Anaconda):

* + - * Python programs are also executed using Anaconda.
      * Anaconda is a free open source distribution of python for large scale data processing, predictive analytics and scientific computing.
      * Conda is a package manager quickly installs and manages packages.
      * In WINDOWS:
      * In windows
        + Step 1: Open Anaconda.com/downloads in web browser.
        + Step 2: Download python 3.4 version for (32-bitgraphic installer/64 -bit graphic installer)
        + Step 3: select installation type( all users)
        + Step 4: Select path(i.e. add anaconda to path & register anaconda as default python 3.4) next click install and next click finish
        + Step 5: Open jupyter notebook ( it opens in default browser)

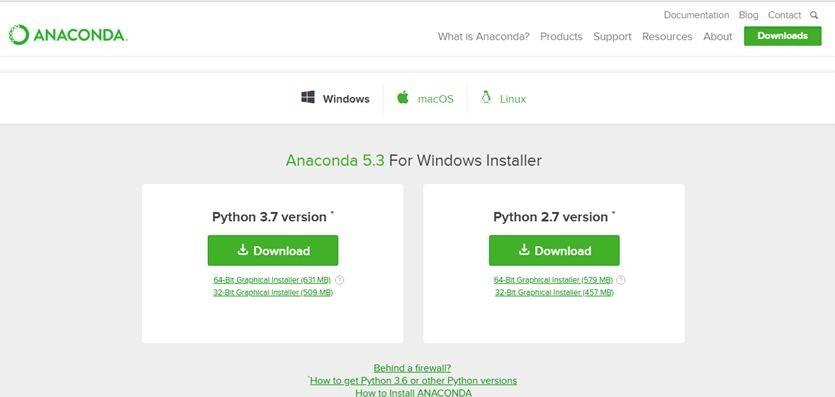


Figure 5 : Anaconda download



Figure 6 : Jupyter notebook

## 3.5PYTHON VARIABLE TYPES:

* Variables are nothing but reserved memory locations to store values. This means that when you create a variable you reserve some space in memory.
* Variables are nothing but reserved memory locations to store values.
* Based on the data type of a variable, the interpreter allocates memory and decides what can be stored in the reserved memory.
* Python variables do not need explicit declaration to reserve memory space. The declaration happens automatically when you assign a value to a variable.
* Python has various standard data types that are used to define the operations possible on them and the storage method for each of them.
* Python has five standard data types –
  + Numbers
  + Strings
  + Tuples
  + Dictionary

## 3.5.1Python Numbers:

* + - * Number data types store numeric values. Number objects are created when you assign a value to them.
      * Python supports four different numerical types − int (signed integers) long (long integers, they can also be represented in octal and hexadecimal) float (floating point real values) complex (complex numbers).

## 3.5.2Python Strings:

* + - * Strings in Python are identified as a contiguous set of characters represented in the quotation marks.
      * Python allows for either pairs of single or double quotes.
      * Subsets of strings can be taken using the slice operator ([ ] and [:] ) with indexes starting at 0 in the beginning of the string and working their way from -1 at the end.
      * The plus (+) sign is the string concatenation operator and the asterisk (\*) is the repetition operator.

## 3.5.3Python Lists:

* + - * A list contains items separated by commas and enclosed within square brackets([]).

* + - * To some extent, lists are similar to arrays in C. One difference between them is that all the items belonging to a list can be of different data type.
      * The values stored in a list can be accessed using the slice operator ([ ] and [:]) with indexes starting at 0 in the beginning of the list and working their way to end -1.
      * The plus (+) sign is the list concatenation operator, and the asterisk (\*) is the repetition operator.

## 3.5.4Python Tuples:

* + - * A tuple is another sequence data type that is similar to the list.
      * A tuple consists of a number of values separated by commas. Unlike lists, however, tuples are enclosed within parentheses.
      * The main differences between lists and tuples are: Lists are enclosed in brackets ( [

] ) and their elements and size can be changed, while tuples are enclosed in parentheses ( ( ) ) and cannot be updated.

* + - * Tuples can be thought of as read-only lists.
      * For example − Tuples are fixed size in nature whereas lists are dynamic. In other words, a tuple is immutable whereas a list is mutable. You can't add elements to a tuple. Tuples have no append or extend method. You can't remove elements from a tuple.
      * Dictionaries are enclosed by curly braces ({ }) and values can be assigned and accessed using square braces ([]).
      * You can use numbers to "index" into a list, meaning you can use numbers to find out what's in lists. You should know this about lists by now, but make sure you understand that you can only use numbers to get items out of a list.
      * What a dict does is let you use anything, not just numbers. Yes, a dict associates one thing to another, no matter what it is.

## 3.6PYTHON FUNCTION:

**3.6.1Defining a Function:**

You can define functions to provide the required functionality.Here are simple rules to define a function in Python. Function blocks begin with the keyword def followed by the function name and parentheses (i.e.()).

Any input parameters or arguments should be placed within these parentheses.You can also define parameters inside these parentheses

The code block within every function starts with a colon (:) and is indented. The

statement returns [expression] exits a function, optionally passing back an expression to the caller.

## 3.6.2Calling a Function:

Defining a function only gives it a name, specifies the parameters that are to be included in the function and structures the blocks of code. Once the basic structure of a function is finalized, you can execute it by calling it from another function or directly from the Python prompt.

## 3.7PYTHON USING OOP’s CONCEPTS:

**3.7.1Class:**

* + - * Class: A user-defined prototype for an object that defines a set of attributes that characterize any object of the class. The attributes are data members (class variables and instance variables) and methods, accessed via dot notation.
      * Class variable: A variable that is shared by all instances of a class. Class variables are defined within a class but outside any of the class's methods. Class variables are not used as frequently as instance variables are.
      * Data member: A class variable or instance variable that holds data associated with a class and its objects.
      * Instance variable: A variable that is defined inside a method and belongs only to the current instance of a class.
      * Defining a Class:
        + We define a class in a very similar way how we define a function.
        + Just like a function ,we use parentheses and a colon after the class name(i.e. ():) when we define a class. Similarly, the body of our class is

indented like a functions body is.



Figure 7 : Defining a Class

## 3.7.2 init method in Class:

* + - * The init method — also called a constructor — is a special method that runs when an instance is created so we can perform any tasks to set up the instance.
      * The init method has a special name that starts and ends with
      * two underscores: init ().

**GENDER DETECTION AND AGE PREDICTION**

**4.1 PROBLEM STATEMENT:**

To predict the Gender and Age of face images using Open CV and Convolutional Neural Networks.

**4.2 OBJECTIVE OF THE CASE STUDY**

In the modern world the age and gender prediction is most needed for computers in various aspects.Such as in validation of the clients working in a company.It is very difficult for an individual to check the gender and age for a mass group of people.In those situatuion a system which consist of prediction of age and gender is neccessary to save the time and work.Google,Instagram and other AI detectors of face and gender feature uses the machine learning algortihms and build the model.Then they deploy the machine learning algorithms to process the images.

**MODEL BUILDING**

**4.3 Pre-Processing the Data:**

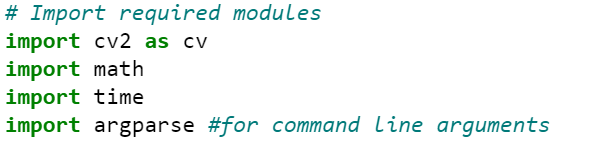
Preprocessing of the data actually involves the following steps:

**4.3.1: Getting the Face Images**

We can get the face images from a user at real time or we can pass a set of images to the

model.

**4.3.2: Importing the required packages**

****

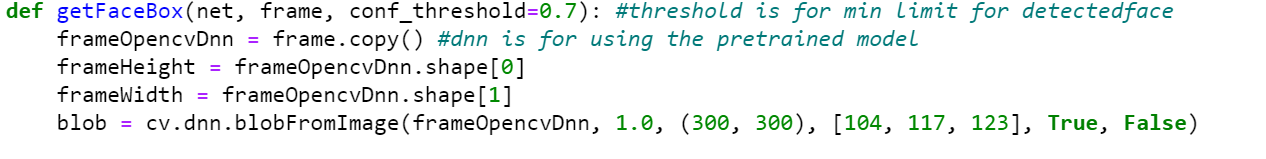
**Figure:1**

* OpenCV is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel.
* Math module provides access to the mathematical functions defined by the C standard.
* Python has defined a module, “time” which allows us to handle various operations regarding time.
* Argparse is the recommended command-line parsing module in Python standard library.

**4.3.3 Defining a function to detect the faces:**

From OpenCV 3.4 version we can use DNN module in which we can use pre-trained

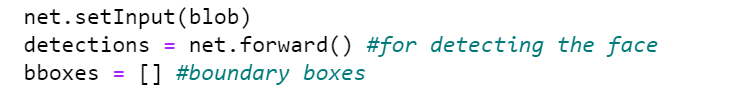
models. For face detections we have getFaceBox function.



**Figure:2**

* conf\_threshold is for validation,whether the detected face is minimum of 70%.The default value of threshold in OpenCV is 0.5.
* Blob is a library for computer vision to detect connected regions in binary digital images.
* blobFromImage creates 4-Dimensional blob from image.
* The parameters for blobFromImage are blobFromImage(image,scalefactor=1.0,size,mean,swapRB)

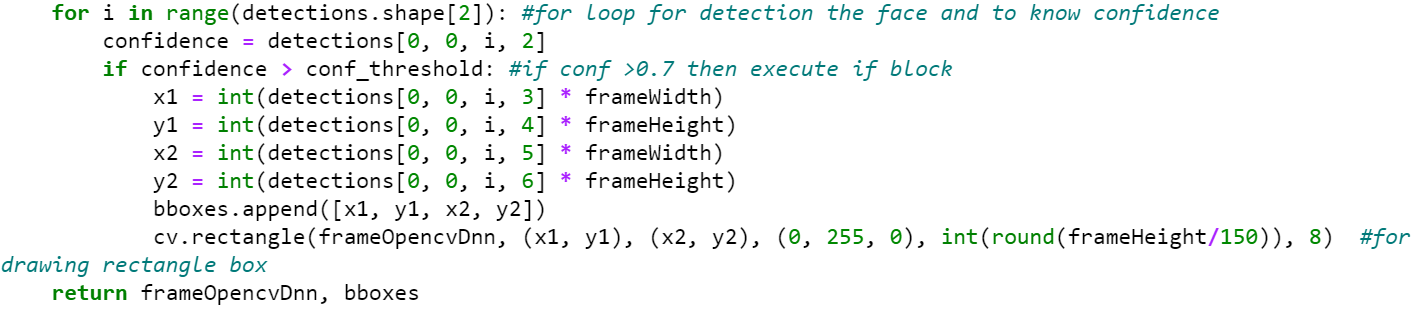
**4.3.4 Setting the input value for Network:**

****

**Figure:3**

* net.forward() method gives Numpy ndarray as output which can be used for plot box on the input image.
* bbox is a Python library that is intended to ease the use of 2D and 3D bounding boxes in areas such as Object Detection.

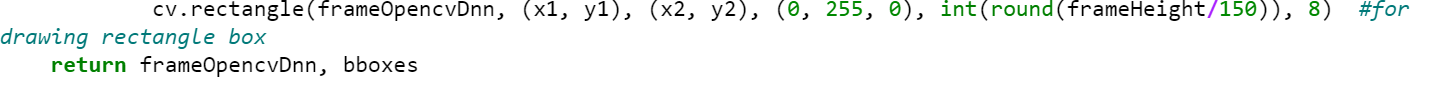
**4.3.5 Detection of faces and the confidence:**

****

**Figure:4**

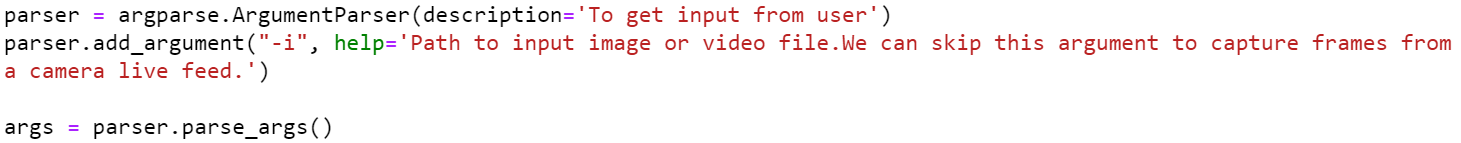
* Extracting the confidence (i.e. probability) associated with the prediction.
* Filtering out the weak detections by ensuring the confidence is greater than the minimum confidence.
* Computing the (x,y) coordinates of the bounding box for the object.

**4.3.6 Drawing the rectangle around the detected face:**

****

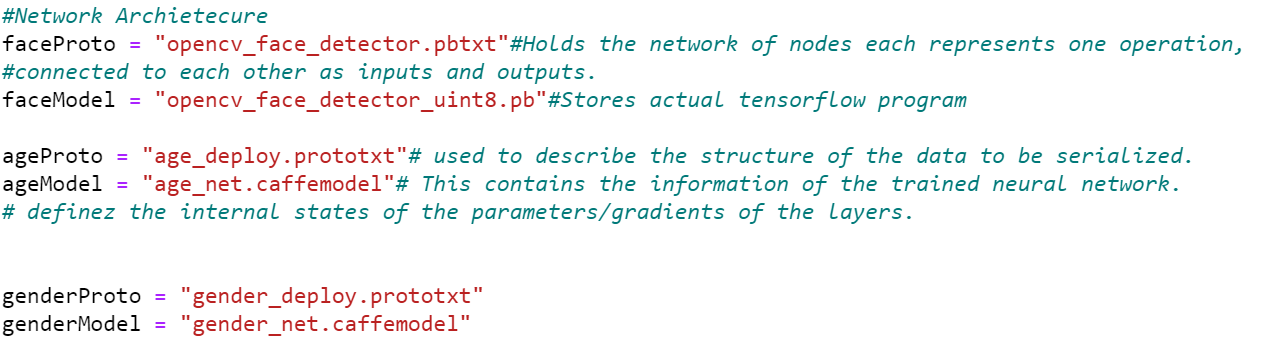
**Figure:5**

**4.3.7 Taking the input from user by argparse:**

****

**Figure:6**

**4.4 Model Building and Network Architecture:**

****

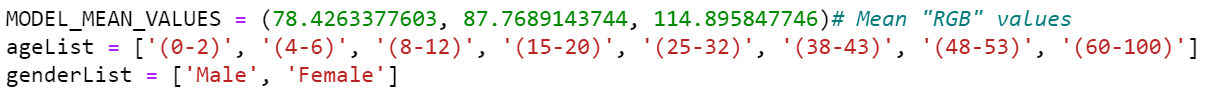
**Figure:7**

* faceProto is assigned with “face detector.pbtxt” pre-trained file which holds the network of nodes each represents one operation.
* faceModel is assigned with “face\_detector\_uint8.pb” pre-trianed tensorflow program
* ageProto is assigned with “agedeploy.prototxt” pre-trained model which is used for serialization of data.
* ageModel is assigned with “age\_net.caffemodel” pre-trianed model which contains the information of trained model.

1. A cafeemodel is used to integrate trained models into data pipelines.
2. It is used to deploy against the new data from user through command line arguments.

* Similarly genderProto and genderModel are assigned with respective pre-trianed models.

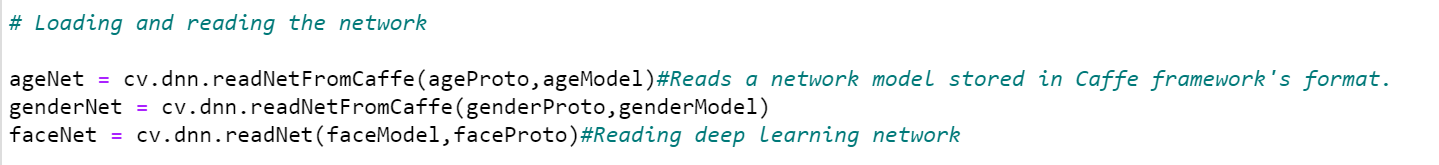
**4.4.1 Model mean values,age list and gender list:**

****

**Figure:8**

* Model mean values are used for mean subtraction.
* Mean subtraction is used to help combat illumination changes in the input images in our dataset.
* Age list and Gender list is used for predicting the age and gender from the given face image.

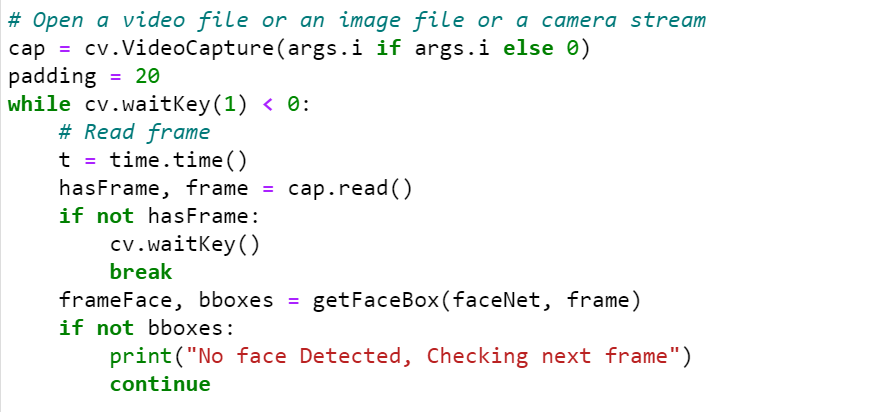
**4.4.2 Loading and Reading the network:**

****

**Figure:9**

* cv.dnn.readNetFromCaffe is used for reading the ageProto and ageModel from the stored caffemodel framework.
* cv.dnn.readNet is used for reading faceModel and faceProto networks.

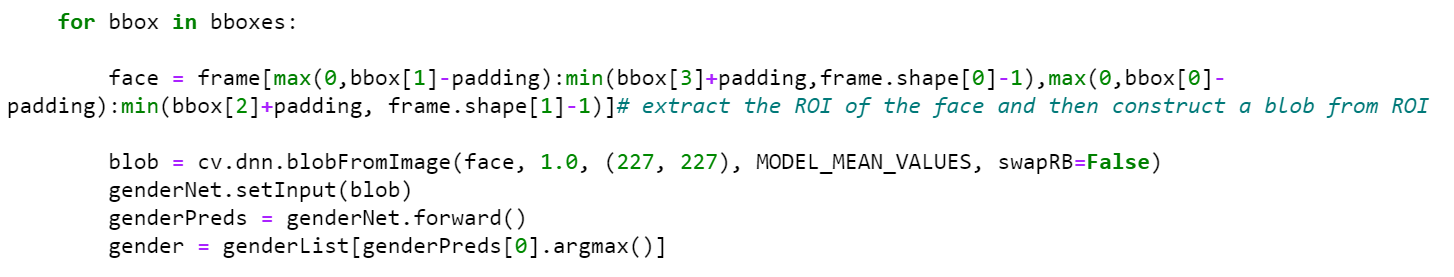
**4.5 Reading the Input:**

****

**Figure:10**

* OpenCV provides VideoCapture() method to get the image frame from the camera feed.
* Padding is used for adding borders for the frame.
* OpenCV also provides waitKey() method.The function waitKey() waits for a key event for a delay.It is calculated in milliseconds.

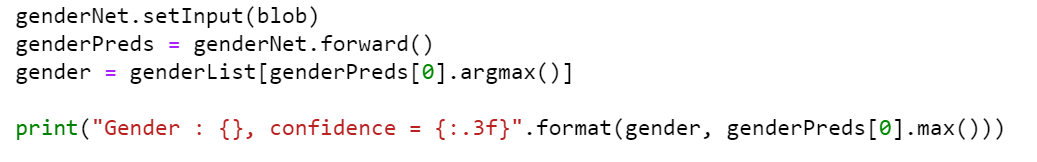
**4.6 Extracting the Region of Interest(ROI) and constructing a blob for ROI:**

****

**Figure:11**

* swapRB is used for BGR ordering.Here swapRB is false because the mean values assumes the order as RGB.

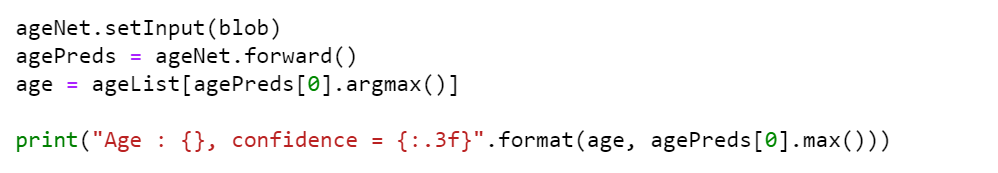
**4.6.1 Predicting the gender:**

****

**Figure:12**

* From the above ROI blob,the blob is passed as input the genderNet model, then from the genderlist the gender the is predicted.
* The print statement returns the gender with confidence in three digit format.

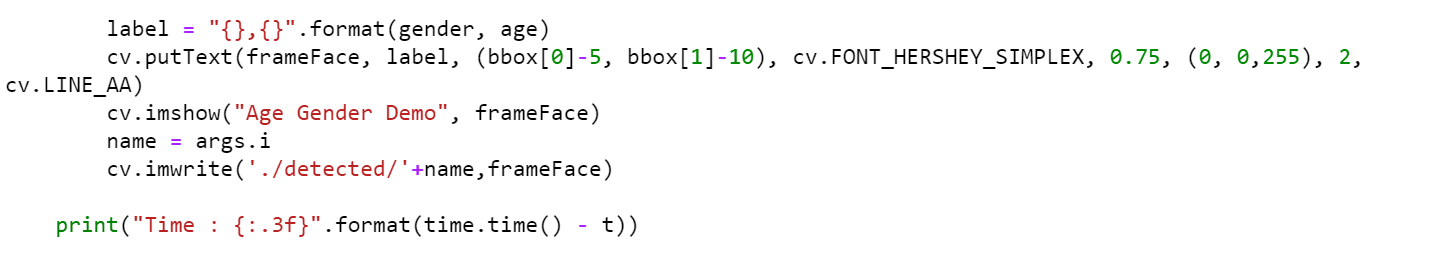
**4.6.2 Predicting the age:**

****

**Figure:13**

* For age prediction the ROI blob is passed as input to the ageNet model, then from agelist the model predicts the age.
* The print statement returns the age with confidence in three digit format.

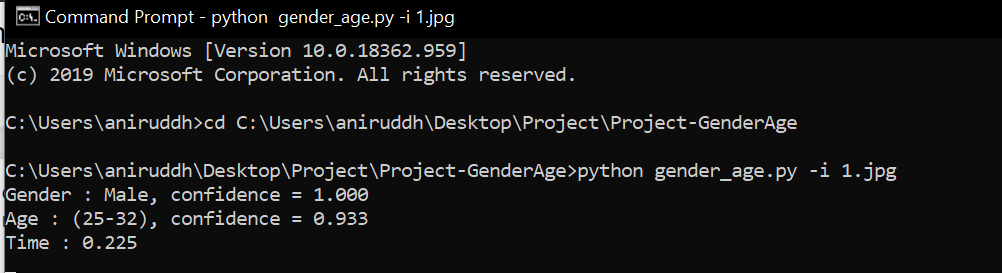
**4.7 Printing the image,gender,age and time :**

****

**Figure:14**

* Label contains the empty dictionary and the format as gender and then age.
* In OpenCV, cv.putText() method is used to draw a text string on any image.
* cv.imshow() method is used to display an image in a window.
* The cv.imwrite() method is used to save an image in local system.This method will save the according to the specified format in current working directory.
* By using time module, the time taken for the execution of program is printed.

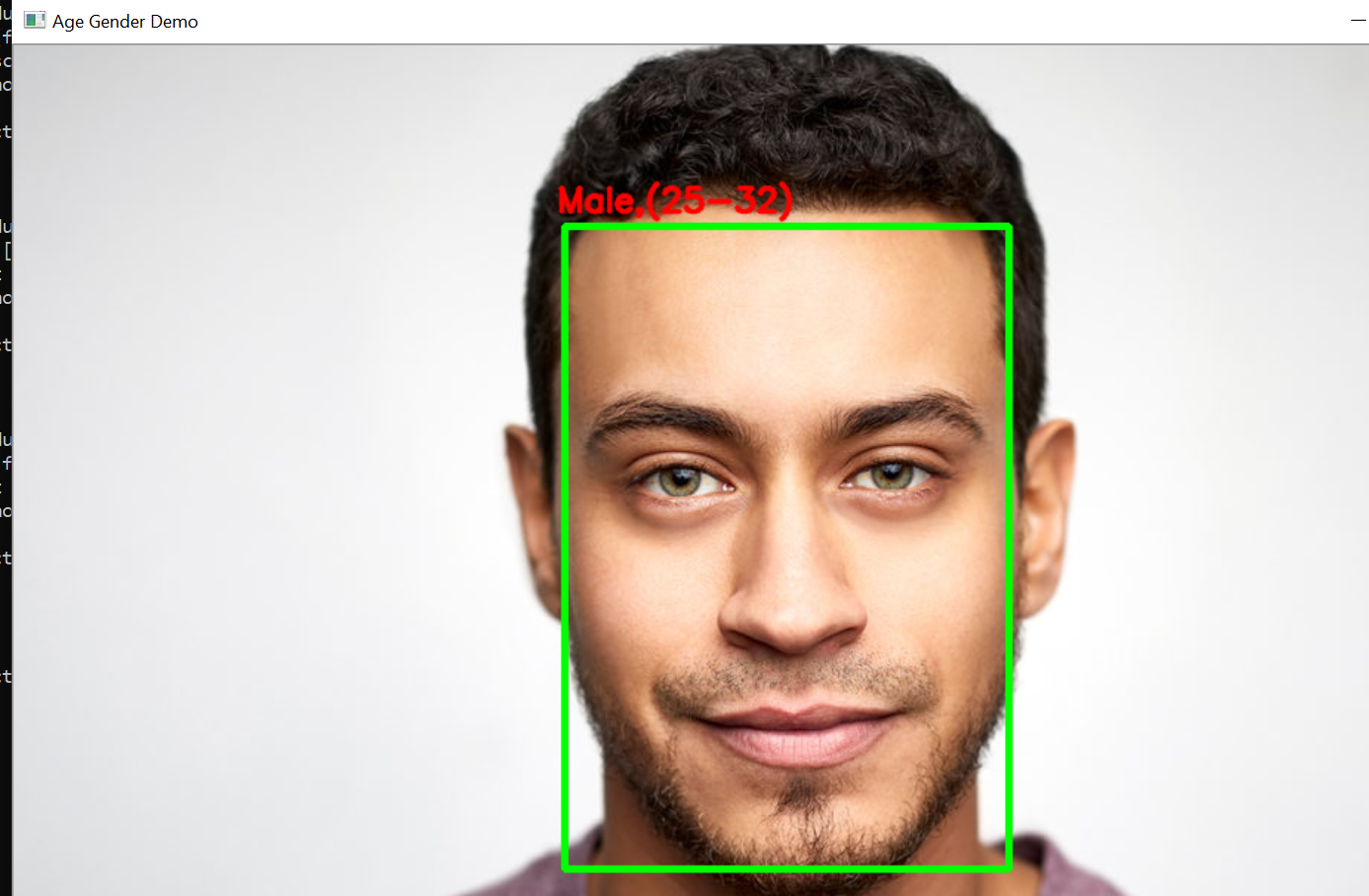
**4.7.1 Output for a sample face image:**

****

* For executing a “.py” file first the current working directory should be set and then

the following command should be executed.

python filename.py -i imagename.jpg



**5.Conclusion:**

The output of the face image shows the accurate result and gives better accuracy

results(i.e. confidence).

By this method or project one can easily predict the gender and age of a person through the face images.

**6. References:**

[1] <https://en.wikipedia.org/wiki/OpenCV>

[2] <https://docs.opencv.org/master/d6/d00/tutorial_py_root.html>

[3] <https://www.pyimagesearch.com/category/deep-learning/>