Deep Face Recognition Estimating Body Mass Index

Submitted in the partial fulfillment for the award of

the degree of

BACHELOR OF ENGINEERING

IN

Artificial Intelligence & Machine Learning

Submitted by:
Bhagath Reddy:18BCS6196
Aditya Gupta : 18BCS6152

Under the Supervision of:
Mrs. Shifali Sharma

Department of AIT-CSE

Outline

- Introduction to Project
- Problem Formulation
- Objectives of the work
- Methodology used
- Results and Outputs
- Conclusion
- Future Scope
- References

Introduction to Project:

The majority of people don't know how to calculate BMI or find it difficult to calculate their BMI daily and don't know whether they are healthy or not.

Together using a person's gender, age and weight stats that can influence on many aspects of their life. It can affect their health as having a high BMI is linked to increased risk of both cardio stroke and diabetes

The human face exhibits information pertaining to identity, a person's disposition, as well as attributes such as gender, age and ethnicity. Biometric emphasis has predominantly been placed on facial recognition.

Introduction to Project:

The attributes such as gender, age, height and weight gaining popularity due to semantic interpretation, like they can show you description which can understandable for people example, describing -old, -male, -short.

Many health problems arise if a person is obese so if a person can estimate their BMI by just uploading their photo, they will be able to know about their health and focus on improving their health .

Problem Formulation:

1:Preprocessing

Input preprocessing pipeline:

loads image to resize and convert an array to create feature model

labeling BMI, Age, Gender

create models for random sample which generate fitting models

2: MTCNN face detector (MTCNN):

alignment: train data to crop face and detect faces

to make predictions

3:VGGFace face prediction (VGGFace): with transfer learning and VGG16 multi tasking to learn multiple tasks here we combine 3 tasks together from keras

Problem Formulation:

4. Training regression models and finding out the best one.

   Used python face recognition library ResNet-34 but with fewer layers and the number of filters reduced by half.
   It maps a “face” into a feature vector which can comprise various Features like : facial-Height, Width ,color, lips width, nose height
   These detected faces are provided as input to regression models.

5. Web application: Using anvil platform to connect python notebook to upload image to test and call the trained model to predict and display

Objectives:

1.To apply Data Image Pre-processing Operations. To apply Facial Recognition algorithms

2 Extraction of face embeddings using a pre-trained face-net architecture.

3 Training regression models and finding out the best one.

4 To Predict BMI weight, height of an individual using his/her face photograph.

Methodology used:

In CNN convolution blocks are followed by the dense layers to make prediction. In naive model the 3 models are required to train separately, increasing maintenance cost

|`[input image] => [VGG16] => [dense layers] => [BMI]`|

|`[input image] => [VGG16] => [dense layers] => [AGE]`|

|`[input image] => [VGG16] => [dense layers] => [SEX]`|

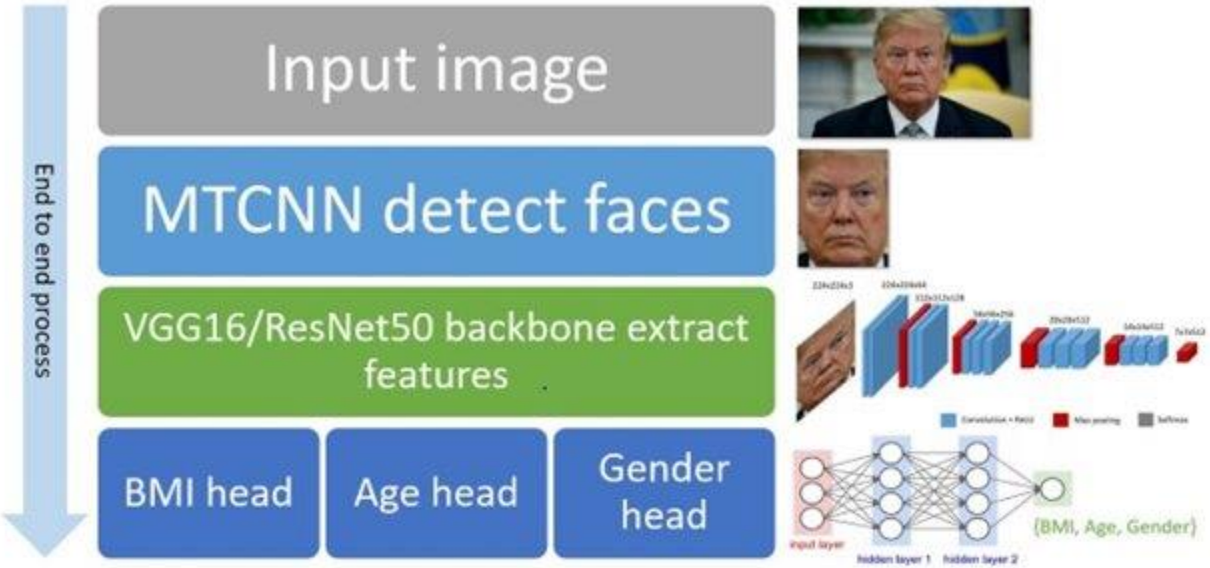
Since we are going to predict `BMI`, `Age`, `Sex` from the same image, we can share the same backbone for the three different prediction heads and hence only one model will be maintained

`[input image] => [VGG16] => [separate dense layers] x3 => weighted([BMI], [AGE], [SEX])`|

Methodology used:

The most simplified multi-task learning structure,

Deep Network with shared convolutional and task-specific fully connected layers with matrix priors VGG-Face,is used as feature extractor to extract the activation vector of the fully connected layer in the CNN architecture.



- *Backbone is loaded with pretrained weights from VGGFace*
- *Prediction (BMI, Age, Gender) heads are 2 fully-connected layers*

Methodology used:

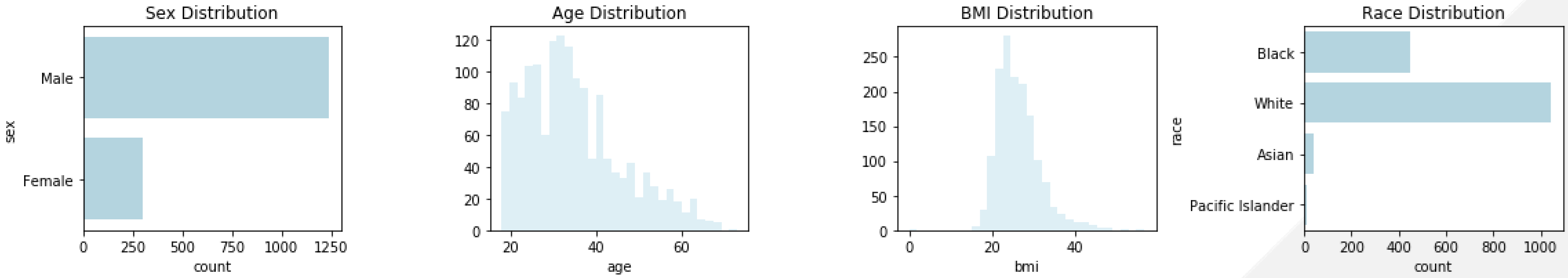
.We approached the problem by observing performing of various loss functions:

- Simple Linear Regression: simple linear equation y=wx+b, we can calculate MSE as: MSE = 1/N ∑ (yi − (w.xi + b))^ 2
- Ridge Linear Regression: adds a penalty for large variations in w parameters.   RSSRIDGE = 1/N ∑ (yi − (w.xi + b)) 2 + ʎ ∑ wj
- Random Forest Regressor: gets perfectly trained on that particular sample data and hence the output depend on multiple decision trees. Cost = sum (y – prediction)2
- Kernel ridge regression: combines Ridge regression and classification (linear least squares with l2-norm regularization)

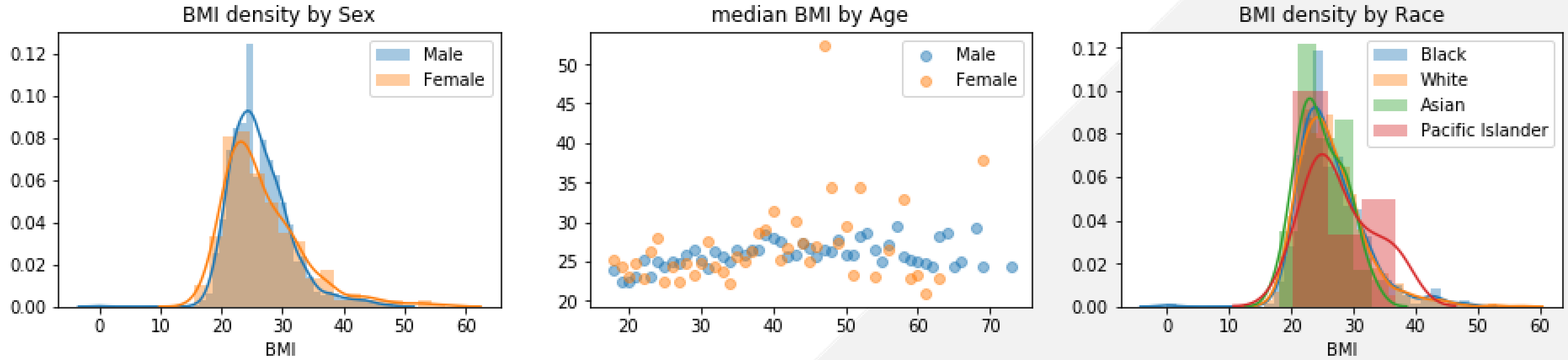
Results and Outputs:

DATA PRE-PROCESSING (USING MTCNN)

MTCNN face alignment



Sex Distribution
Male
Female
sex
count
Age Distribution
age
BMI Distribution
bmi
Race Distribution
Black
White
Asian
Pacific Islander
race
count

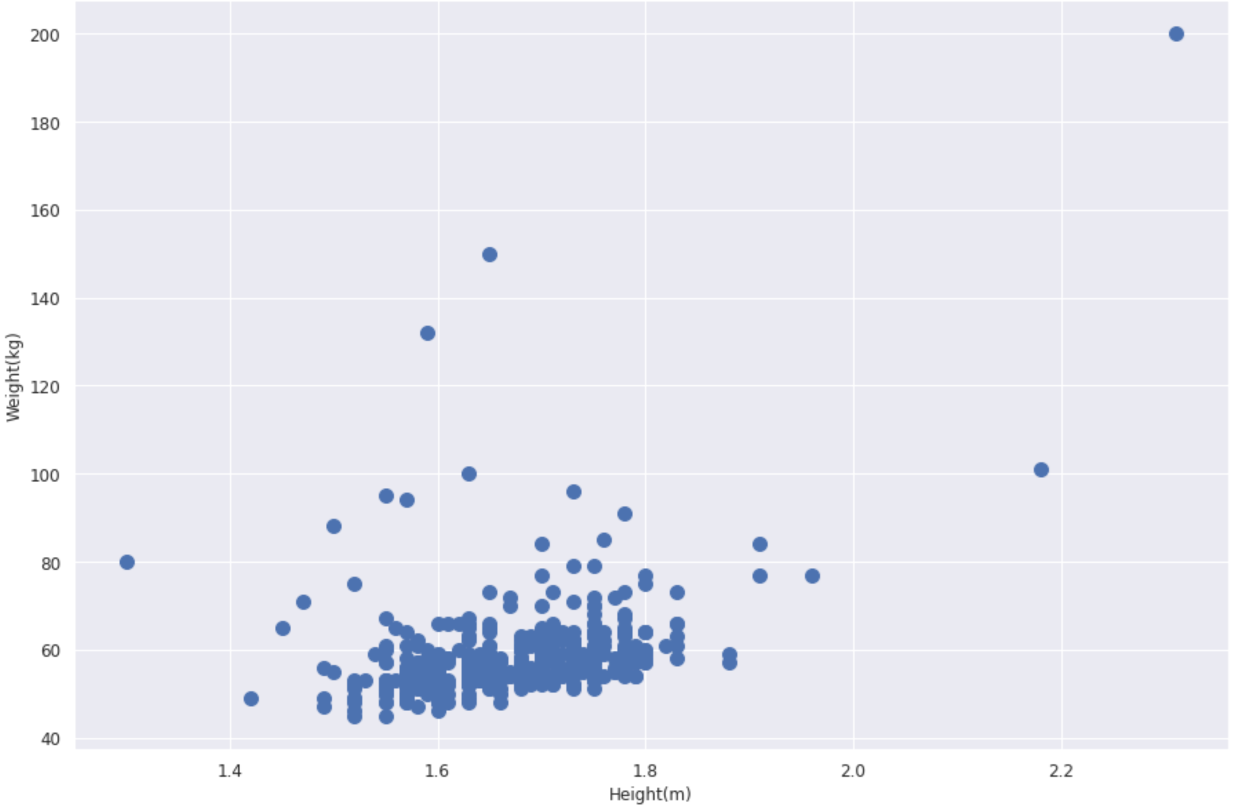


BMI density by Sex
Male
Female
BMI
median BMI by Age
Male
Female
BMI density by Race
Black
White
Asian
Pacific Islander
BMI

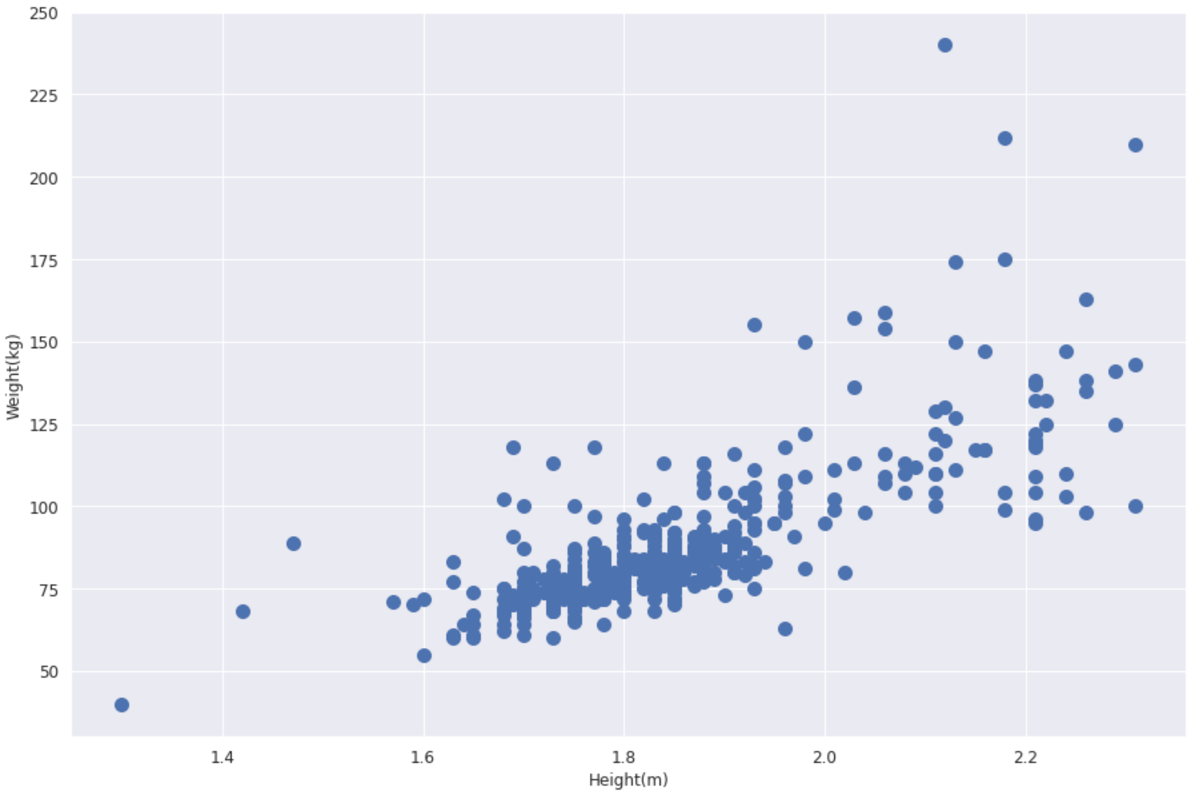
Relationship between parameters:

- Weight vs Height

females



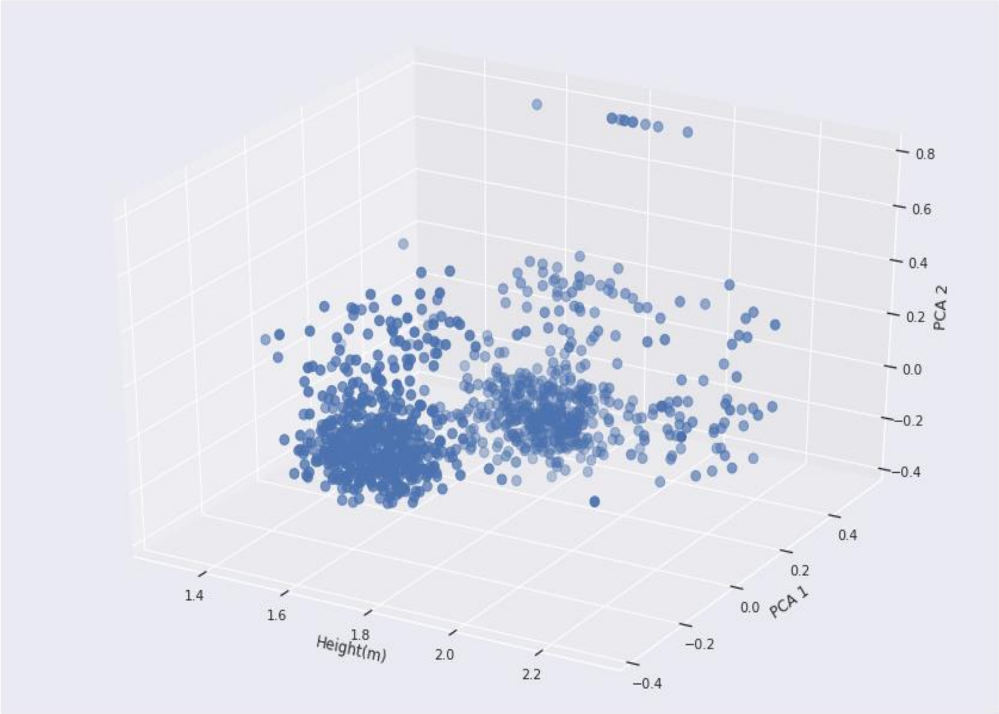
Males



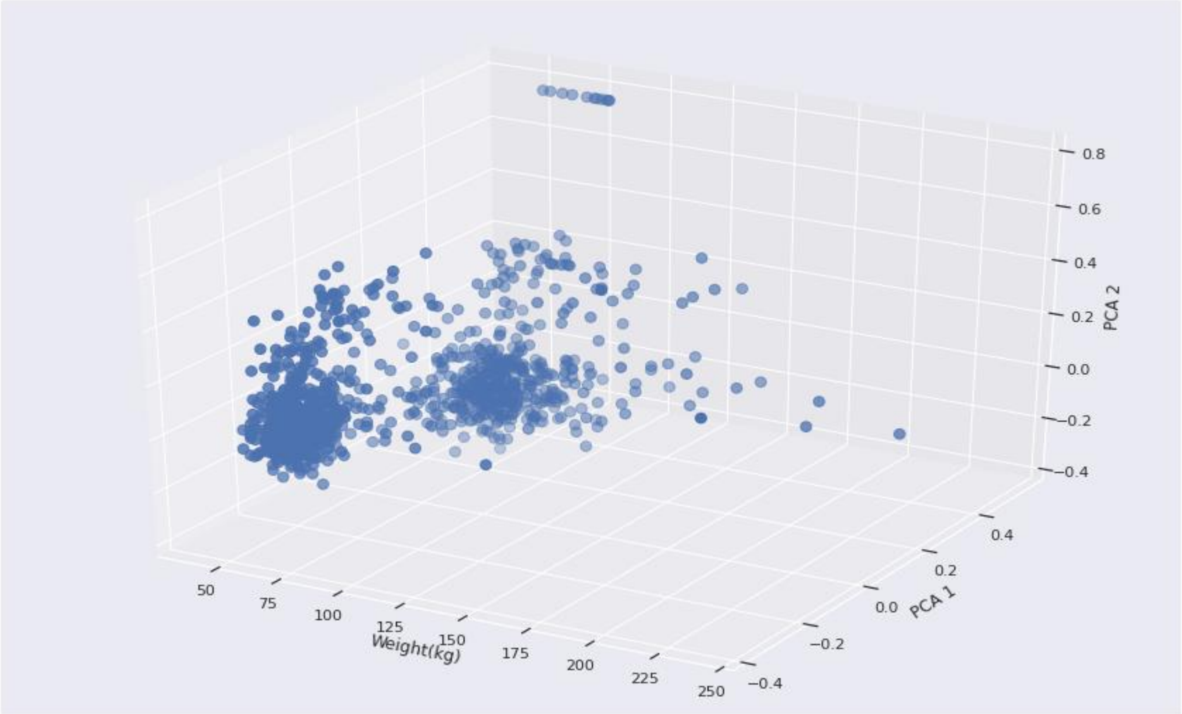
Visualization of Face Embedding

PCA was used for dimension reduction of 128-d feature vector to 2-d.

vs Height



vs weight

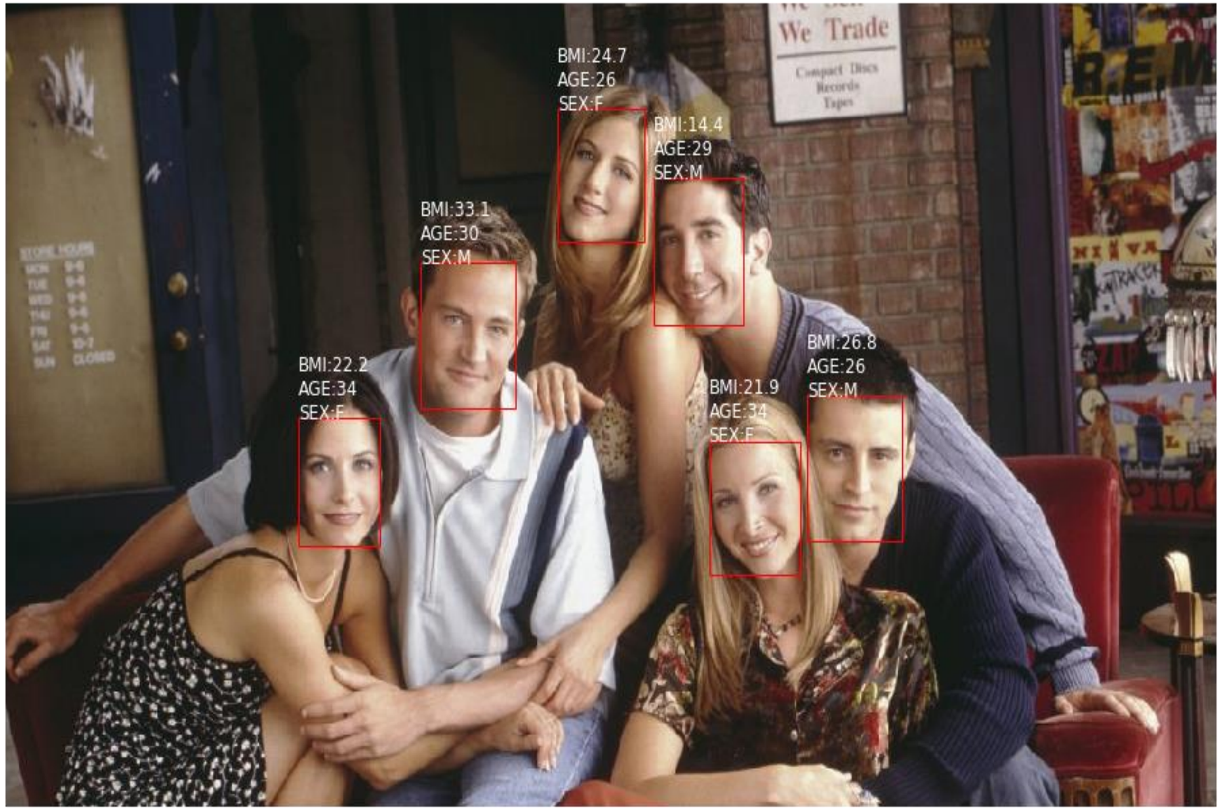


Training regression models and find best one.

5. Data/Models	Height Model	Weight Model	BMI Model
Simple Linear Regression	Mean square error = 0.01 Variance score = 0.31 Average error = 0.0509 Accuracy = 90.22%	Mean square error = 13.57 Variance score = -193.80 Average error = 3.6763 Accuracy = 13.32%	Mean square error = 6.58 Variance score = -240.37 Average error = 2.5612 Accuracy = 18.02%
Ridge Regression	Mean square error = 0.00 Variance score = 0.36 Average error = 0.0479 Accuracy = 90.83%	Mean square error = 0.03 Variance score = 0.60 Average error = 0.1130 Accuracy = 97.39%	Mean square error = 0.01 Variance score = 0.47 Average error = 0.0839 Accuracy = 97.36%
Random Forest	Mean square error = 0.01 Variance score = 0.30 Average error = 0.0483 Accuracy = 90.78%	Mean square error = 0.04 Variance score = 0.48 Average error = 0.1306 Accuracy = 96.98%	Mean square error = 0.02 Variance score = 0.31 Average error = 0.0950 Accuracy = 97.01%

Data/Models	Height Model	Weight Model	BMI Model
Random Forest with tuned hyper parameter	Mean square error = 0.00 Variance score = 0.34 Average error = 0.0472 Accuracy = 90.96%	Mean square error = 0.03 Variance score = 0.59 Average error = 0.1124 Accuracy = 97.41%	Mean square error = 0.02 Variance score = 0.43 Average error = 0.0830 Accuracy = 97.40%
Kernel Ridge Regression	Mean square error = 0.01 Variance score = 0.29 Average error = 0.0531 Accuracy = 90.18%	Mean square error = 0.03 Variance score = 0.54 Average error = 0.1243 Accuracy = 97.10%	Mean square error = 0.02 Variance score = 0.32 Average error = 0.1042 Accuracy = 97.60%

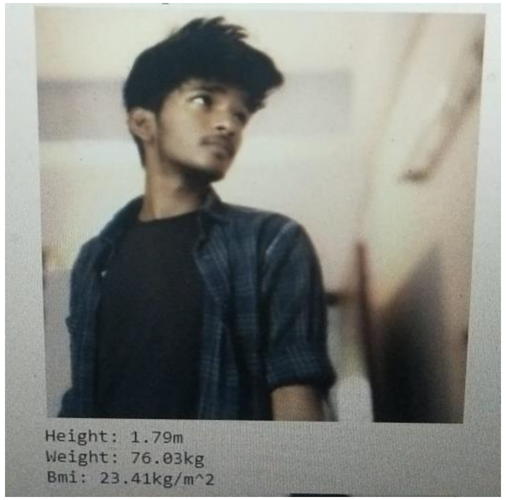
Predict multiple faces



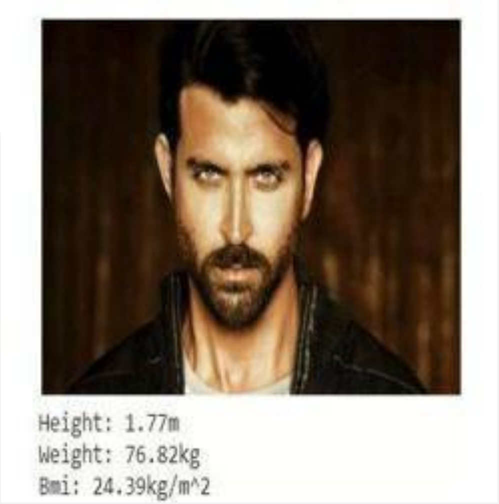
Single face


Height: 1.79m
Weight: 82.91kg
Bmi: 25.3kg/m^2


eight: 1.66m
eight: 57.81kg
mi: 20.78kg/m^2

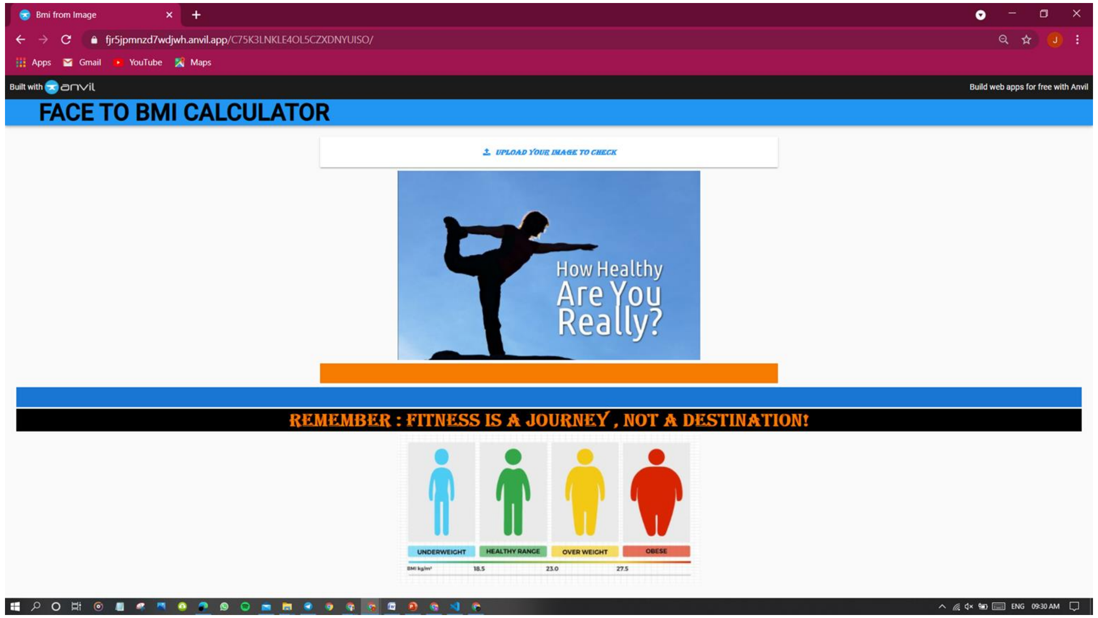

Height: 1.79m
Weight: 76.03kg
Bmi: 23.41kg/m^2

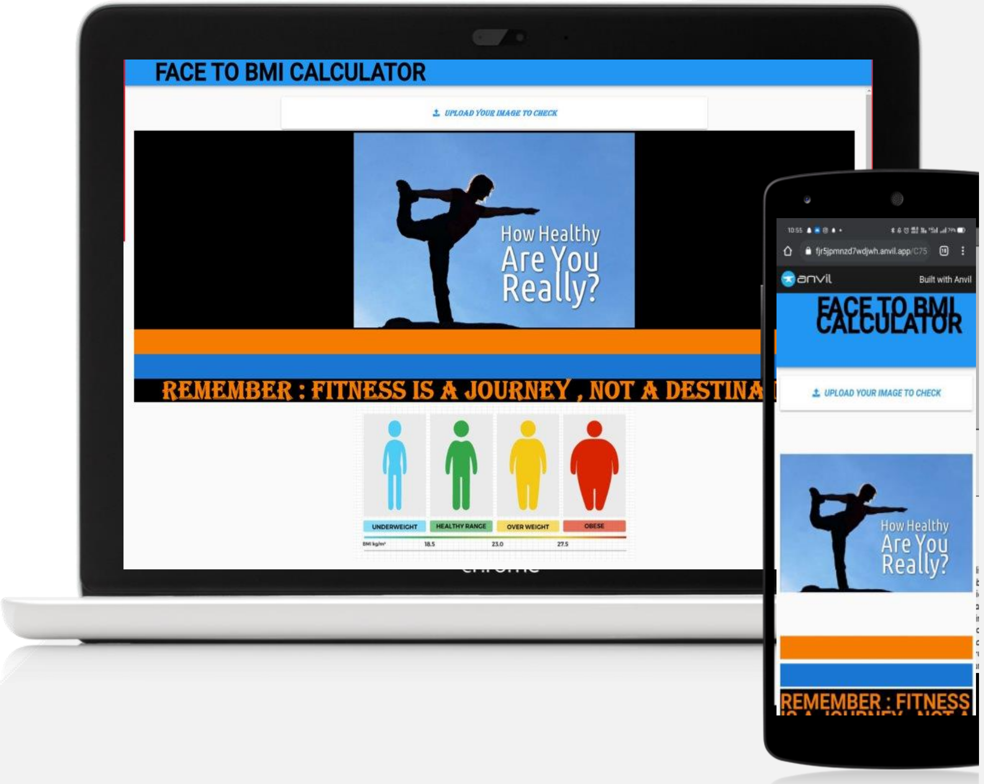

Height: 1.83m
Weight: 87.19kg
Bmi: 25.33kg/m^2


Height: 1.77m
Weight: 76.82kg
Bmi: 24.39kg/m^2

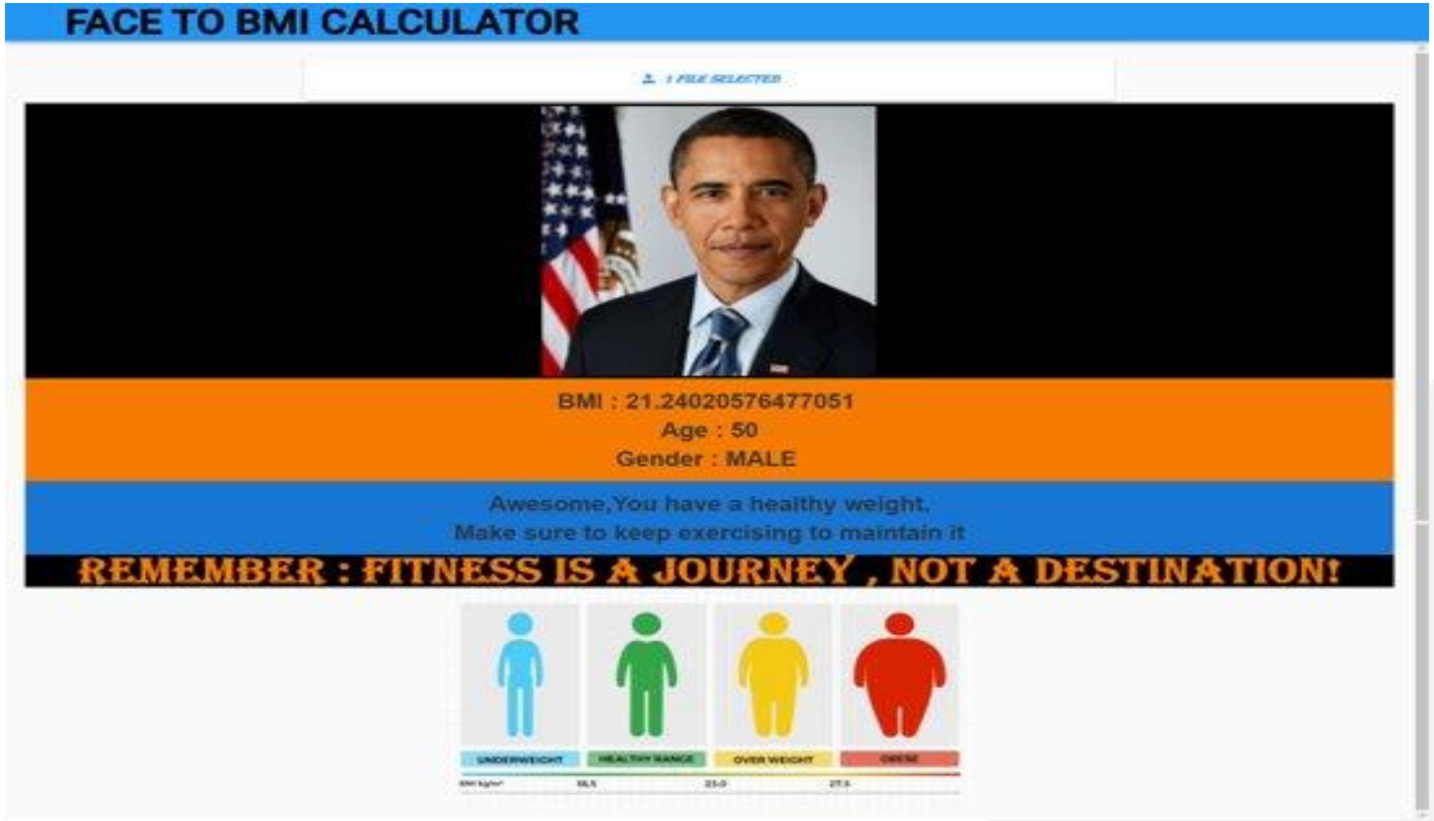
Using anvil for web application

Anvil- Front End





Output:


FACE TO BMI CALCULATOR
BMI : 21.24020576477051
Age : 50
Gender : MALE
Awesome,You have a healthy weight.
Make sure to keep exercising to maintain it
REMEMBER : FITNESS IS A JOURNEY , NOT A DESTINATION!

Conclusion:

It was found that Random Forest Regressor with tuned hyper parameters outperformed all the models in terms of the mean squared error and explained variance.

BMI can be experimentally verified from the predicted height and weight as:

$$BMI = weight\ [kg] / height^2\ [m^2]$$

This project presented a novel approach for estimating height, weight and BMI on single-shot facial images,

based on regression models. Experiments conducted on the dataset, resulted in absolute mean error of 0.083

and variance score of 0.43.

Future Scope:

- We did not observe a significant gender-bias in estimating height, weight and BMI.
- However, more work is necessary in this regard.
- Future work will involve the additional study of ethnicity in order to improve utilization of facial appearance
- for height, weight and BMI estimation.
- The web applications can be useful for Gyms , Fitness Trainers and Fitness Apps can use this as a marketing strategy.
- We can prompt the potential customers to check their BMI by loading their image and alert the customer about the problems they might have to face due to excess weight and inspire them to join

References:

1. MTCNN: https://github.com/ipazc/mtcnn (1, 2) Zhang, K., Zhang, Z., Li, Z., and Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10):1499–1503.

2. VGGFace: https://github.com/rcmalli/keras-vggfac

3. J. Bekios-Calfa, J. M. Buenaposada, and L. Baumela, “Robust gender recognition by exploiting facial attributes dependencies,” PRL, 2013.

2. J. C. Klontz and A. K. Jain, “A case study on unconstrained facial recognition using the boston marathon bombings suspects,” Technical Report, no. MSU-CSE-13-4, 2013

3. A. Dantcheva, C. Velardo, A. D’angelo, and J.-L. Dugelay, “Bag of soft biometrics for person identification,” Multimedia Tools and Applications, vol. 51, no. 2, pp. 739–777, 2011.

4. L. Wen and G.-D. Guo, “A computational approach to body mass index prediction from face images,” Image and Vision Computing, vol. 31, no. 5, pp. 392–400, 2013.

5. A. Dantcheva, P. Bilinski, F. Bremond, "Show me your face and I will tell you your height, weight and body mass index," Proc. of 24th IAPR International Conference on Pattern Recognition (ICPR), (Beijing, China), August 2018.

6. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

7. “Using Computer Vision to Infer Body Mass Index on Social Media”, by Enes Kocabey, Mustafa Camurcu, Ferda Ofli, Yusuf Aytar, Javier Marin, Antonio Torralba, Ingmar Weber at ICWSM'17.