Face to BMI: Using Fine-Tuned CV and ML Algorithm to Infer Body Mass Index

Presenter: Richard Yang

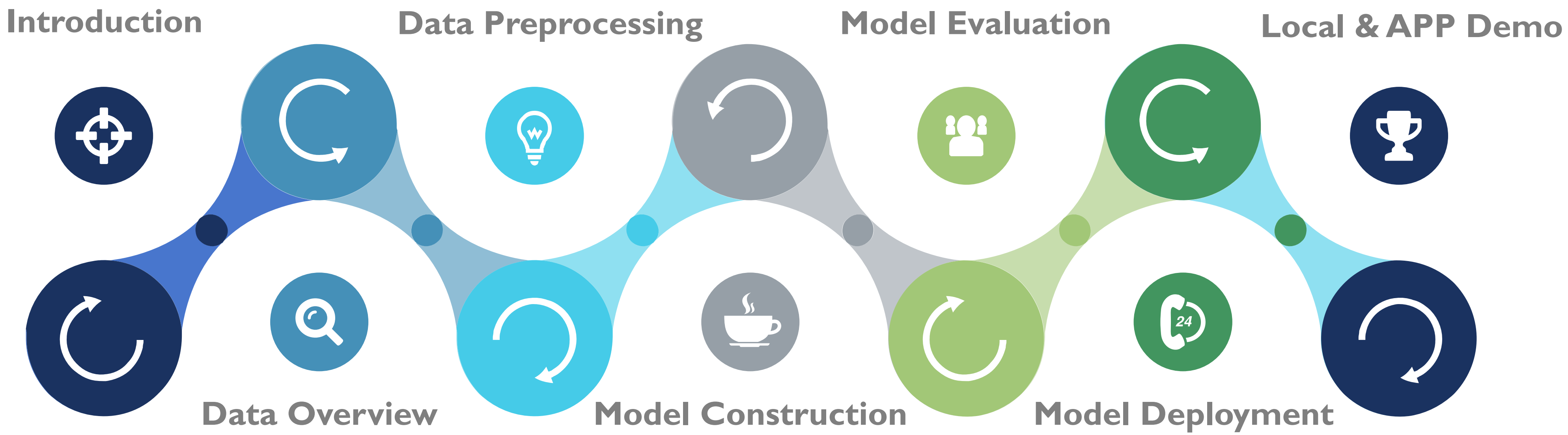
Machine Learning and Predictive Analytics

M.S in Applied Data Science

University of Chicago

Date: May 22, 2023

Agenda



Introduction
Data Overview
Data Preprocessing
Model Construction
Model Evaluation
Model Deployment
Local & APP Demo
24

Introduction

•The **VGG model**, known for its exceptional performance in image classification, is capable of extracting high-level features from facial images.

•The model is fine-tuned using a dataset of labeled facial images with corresponding BMI values.

•The developed system offers a non-invasive and efficient means of estimating BMI, revolutionizing healthcare practices.

•The implications of this technology include personalized fitness recommendations, early detection of obesity-related conditions, and tailored interventions for healthier lifestyles.

Data Overview

Data Label	Image File
4206 records	3963 valid image

- Each record includes attributes such as BMI, gender, training/testing indicator, and image file name.
- BMI values represent the target variable, indicating body weight relative to height.
- The gender attribute identifies the gender of the individual associated with each record.
- The dataset is divided into training and testing subsets based on the "is_training" attribute.
- Image files serve as visual input for the VGGFace model used in fine-tuning and BMI prediction.

Data Preprocessing

Two different preprocessing methods have been applied to prepare the dataset for training and evaluation.

- Images are resized to 224x224 pixels, a standard size for VGGFace models.
- Images are converted to arrays and preprocessed using Keras' preprocess_input function.
- Preprocessed images are stored in the 'images_kimage' list.
- Dataset is split into training and testing sets based on the 'is_training' column.
- Training and testing images are converted to NumPy arrays, 'train_images' and 'test_images'.

- The second preprocessing method involves using the VGGFace ResNet50 model for feature extraction.
- The ResNet50 model is created and the "avg_pool" layer is used to obtain extracted features.
- Images are processed in batches, loaded, resized, converted to arrays, and preprocessed.
- Processed images are fed into the ResNet50 model for feature extraction using the predict method.
- Extracted features from all batches are concatenated to obtain the final feature matrix for training.

Model Construction

Approach 1: Feature Extraction + SVR Model

1. Extracted facial features using the pre-trained VGGFace model.

2. Trained a Support Vector Regression (SVR) model with RBF kernel and epsilon value of 0.2.

3. Used the extracted features as input to the SVR model to map to corresponding BMI values.

Approach 2: Fine-tuning VGGFace Model(Resnet50)

1. Froze the pre-trained layers to retain their weights and feature extraction capabilities.

2. Added custom dense layers on top, including dropout and final dense layer for BMI prediction.

3. Compiled the model with Adam optimizer, MSE loss, and a learning rate of **0.00001**.

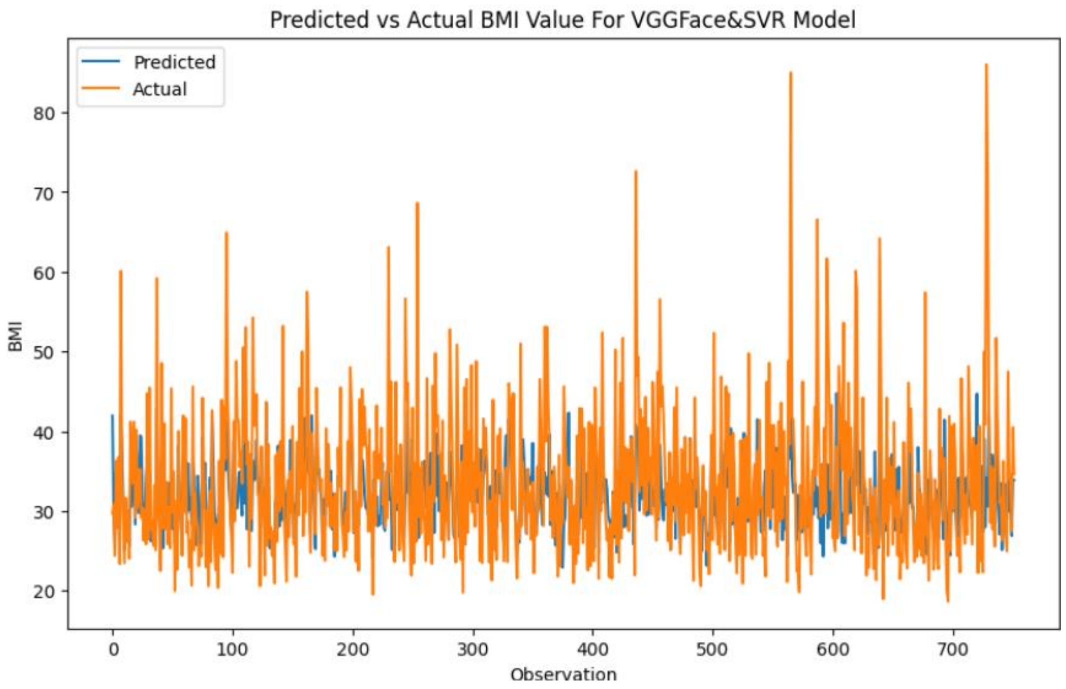
4. Trained the model for **20 epochs**

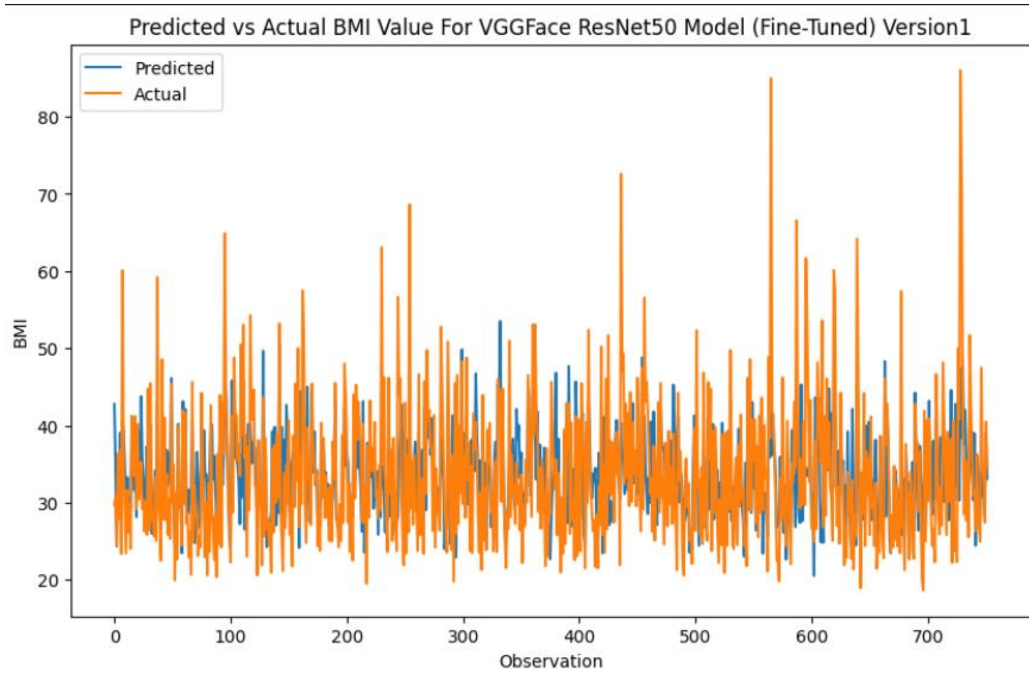
Model Evaluation 1 – Evaluation Metrics

	Feature Extraction + SVR Model	Fine-tuning VGGFace Model(Resnet50)
Mean Squared Error	54.550	50.869
Root Mean Squared Error	7.386	7.132
Mean Absolute Error	5.177	5.190
R-Squared	0.357	0.401
Correlation Coefficient	0.652	0.634

Model Evaluation 2

Line Chart Visualization of Predicted VS Actual BMI Value

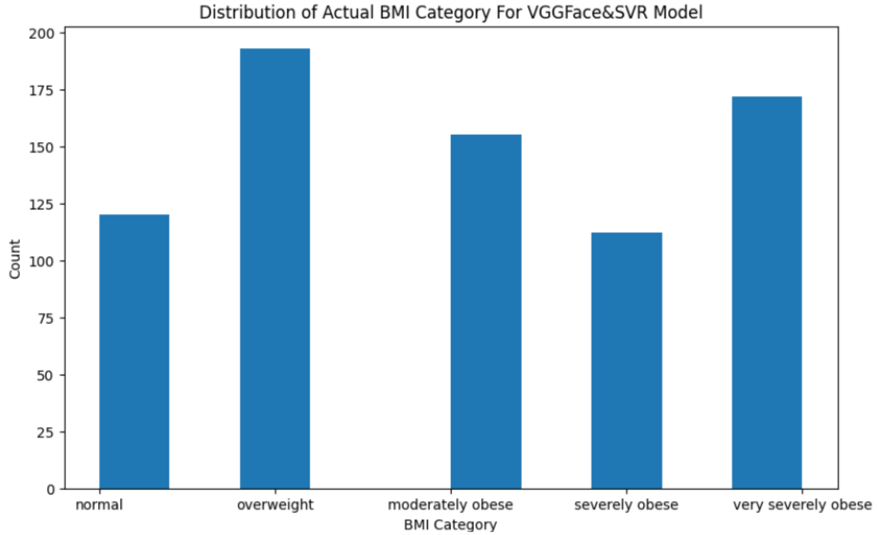


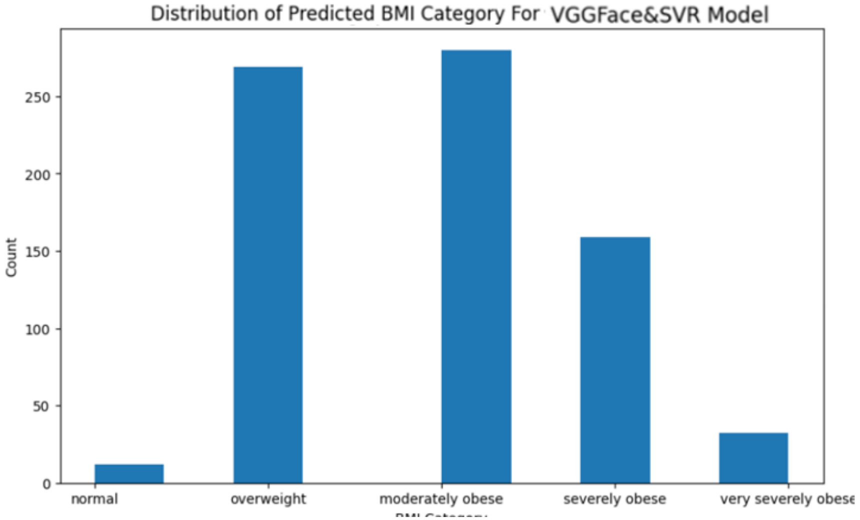


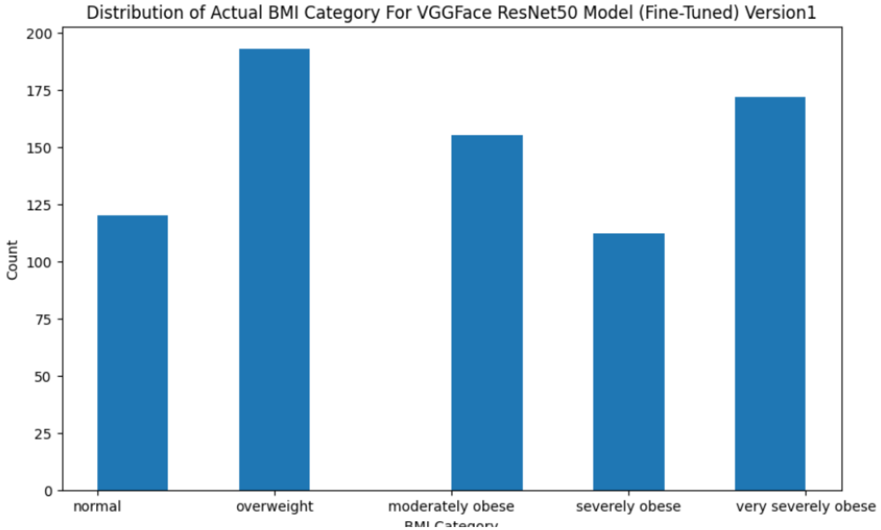
According to Figures, the predicted BMI values from the fine-tuned model exhibited a greater overlapping range with the actual BMI values, indicating a closer alignment between the predicted and actual values.

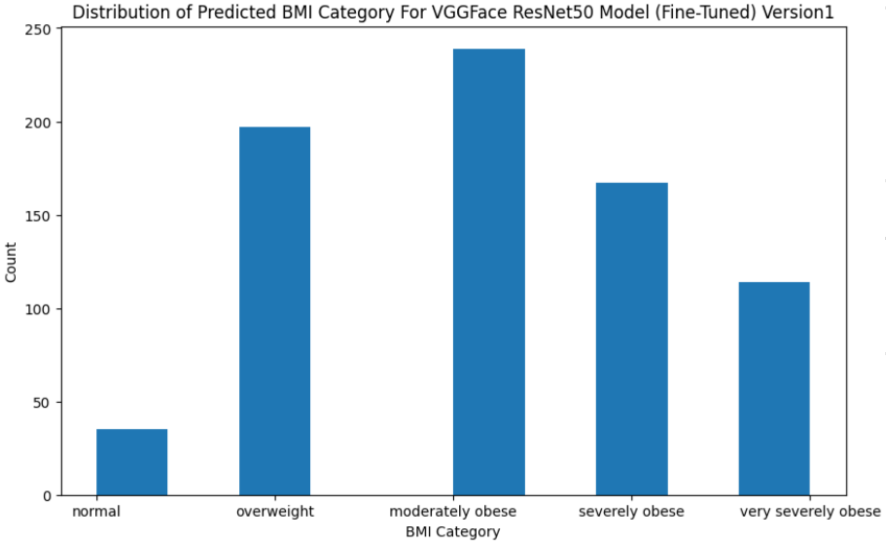
Model Evaluation 3

Distribution of Predicted VS Actual BMI Category for two models









The fine-tuned model has better performance, especially in predicting the 'normal', and 'very severely obsess' BMI category.

For the fine-tuned VGGFace Model, the model was able to adapt and learn features that are more relevant to BMI estimation from facial images. This customization allowed the model to capture subtle facial cues and patterns that are indicative of BMI, leading to improved predictive performance.

Model Deployment: Real-time video capture on a local PC

To deploy the BMI estimation model developed using two different approaches, I utilized two different methods: **real-time video capture on a local PC camera** and a **web application** developed with Streamlit.

1. Preprocesses the captured frame by resizing it to 224x224 pixels and normalizing the pixel values.

2. Detects faces in the frame using the Haar Cascade classifier.

3. Extracts the face region from each detected face.

4. Uses the fine-tuned model to predict the BMI for each face region.

5. Draws bounding boxes around the detected faces and adds the BMI prediction text to the frame.

6. Displays the frame with the bounding boxes and BMI predictions in real-time.

The real-time video capture approach offers the advantage of immediate and interactive estimation, allowing users to view their BMI estimations in real time.

Model Deployment: Web Application

The web application deployment provides convenience and accessibility, enabling users to upload images from various sources and obtain BMI estimations.

1. I deployed a Web Application using Streamlit, The model and the necessary packages are stored in the Git LFS: Git Large File Storage

2. The web app allows users to upload an image for BMI estimation.

3. The uploaded image is processed using similar steps as in the first approach, including face detection, face region extraction, and BMI prediction using the fine-tuned model.

4. The web app then displays the uploaded image with bounding boxes around the detected faces and the corresponding BMI predictions

In summary, it is important to note that both methods are susceptible to noise present in the images. Factors such as wearing glasses, lighting conditions, and the angle of the face in the video or uploaded image can introduce variability and potentially affect the accuracy of the BMI estimation.

Reference

Kocabey, E., Camurcu, M., Ofli, F., Aytar, Y., Marin, J., Torralba, A., & Weber, I. (2017, March 9). Face-to-BMI: Using computer vision to infer body mass index on social media. arXiv.org. https://arxiv.org/abs/1703.03156

Appendix

Web Application: https://zhiliny2-mltest-flaskapicopy-yf5j4j.streamlit.app/

Github Repository For App Deployment: https://github.com/zhiliny2/mltest

THANK YOU!