Age Estimation from Partial Dorsal and Palmar Hand Images

Abstract

Age estimation has potential applications in radiographical, clinical, and forensic practice because it can be used to determine the best treatment strategy. Previous studies estimate age through facial features, which poses a significant potential risk to privacy. An unexplored avenue in this domain is the utilization of close-up partial dorsal and palmar hand images for age estimation. This project aims to pioneer this approach, employing ML algorithms to learn and extract significant features from close-up skin images to estimate age. In addition, a comparative analysis was conducted on conventional ML algorithms (SVR, LGBMR, and RFR) and state-of-the-art DL techniques (DenseNet201) to uncover the potential benefits of using DL techniques over ML algorithms. The findings demonstrate that the Support Vector Regression algorithm performed slightly better than the Light Gradient Boosting Machine Regression and Random Forest Regression algorithms at age estimation. On the other hand, the DenseNet201 significantly outperformed the other models in continuous age estimation, with a Mean Squared Error of 0.66 years and Root Mean Squared Error of 0.81, indicating its effectiveness in capturing complex relationships in the data. Also, the DenseNet201 captured a higher proportion (87%) of the variance in the age values, accounting for the number of features used to fit the model, than the conventional ML algorithms.

1 Introduction

Accurately estimating age can be an essential tool in various fields, mainly in radiographical, clinical, and forensic practice. In radiography, age estimation can be used to determine the suitable timing for treatment based on tissue, bone, and organ maturity (Štern, Payer, and Urschler 2019). Age estimation is also vital for personalized healthcare interventions, disease risk assessment, and the development of targeted clinical treatments (Rew et al. 2019). In forensic science, age estimation can be used to determine the age of unknown individuals and age verification (Alarifi et al. 2019). The advancement of image processing and Artificial Intelligence (AI) algorithms provides a chance to bridge the gap between external appearance and internal aging processes.

An unexplored avenue in this domain is the utilization of partial dorsal and palmar hand images for age estimation. The majority of research on age estimation used naturalistic facial features to estimate age (Choi et al. 2013; Zaghbani, Boujneh, and Bouhlel 2018). The approach of using facial images presents potential privacy risks. Faces are highly unique to individuals, which makes the images easily linkable to personal identities. The facial data could also be used for profiling, such as inferring attributes like race, gender, and emotions (Han et al. 2015). Moreover, individuals may not always be aware that their facial images are being captured and analyzed.

In comparison, hand images lack the distinct features needed to identify a person easily. They also provide limited scope for profiling since they convey significantly less information about an individual's demographic characteristics. Furthermore, the use of hand images for age estimation would require active participation from the subject, such as placing their hands on a scanner. Hence, the individuals are aware of and consent to the capture of their hand images. Applying partials would further reduce any potential privacy risks that might arise from using dorsal and palmar hand images for age estimation.

Most of the studies on the use of dorsal and palmar hand images for age estimation are recent, which suggests that it is emerging research. Consequently, a significant proportion of them utilize both traditional ML and DL techniques for automated age estimation (Ray and Chakrabarti 2019; Moon and Lee 2022). These prior works found that using DL techniques allows researchers to extract high-level Deeply Learned Aging (DLA) features from images due to them being more complex than conventional ML algorithms. However, many of them centered on using full-hand images to estimate the age of the subject accurately. Given the focus on full dorsal and palmar hand images, a relevant research question for this study is: How effective is age estimation from DLA features extracted from partial dorsal and palmar hand images relative to conventional methods that use full hand images? Additionally, various researchers have found that DL techniques generally perform better than conventional ML algorithms at estimating age from skin images. Therefore, this study will attempt to answer the question: Do DL techniques outperform traditional ML algorithms at age estimation while using partial dorsal and palmar hand images as input?

This project aims to pioneer this approach, employing Deep Learning (DL) algorithms to learn and extract Deeply Learned Aging (DLA) features from partial dorsal and palmar hand images to estimate age. The features include skin aging phenotypes or potentially uncovering novel indicators of previously unidentified features. To enhance the project's depth, a comparative analysis of age estimation will be conducted using conventional Machine Learning (ML) algorithms and state-of-the-art DL techniques. The comparative approach aims to uncover the potential benefits of using DL techniques over ML algorithms. In order to answer the proposed research questions, the study will set out to achieve the following objectives:

- Identify existing methodologies and techniques for age estimation using dorsal and palmar hand images.
- Investigate the current state-of-the-art DL and ML algorithms for age estimation tasks.
- Develop a methodology for deriving partial dorsal and palmar hand images from full hand images.
- Design and implement a framework for extracting DLA features from partial hand images.
- Collect and pre-process a dataset of dorsal and palmar hand images suitable for age estimation experiments.
- Implement conventional ML algorithms for age estimation using partial hand images as input.
- Implement DL techniques for age estimation using partial hand images as input.
- Evaluate the performance of both ML and DL algorithms in terms of age estimation accuracy using appropriate evaluation metrics.
- Compare the effectiveness of DL-based age estimation with traditional ML methods when using partial dorsal and palmar hand images.
- Provide insights into the potential applications and limitations of utilizing partial hand images for age estimation in real-world scenarios.
- Propose recommendations for future research directions in the field of age estimation using dorsal and palmar hand images.

The research makes several contributions in the field of automated age estimation from dorsal and palmar hand images. For starters, it discovered a research gap in the field of automated age estimation using partial as opposed to full dorsal and palmar hand images. Additionally, the study proposed a framework for deriving partial dorsal and palmar hand images and then extracting DLA features from them. The framework was designed to facilitate a comparative analysis of some rarely used ML algorithms and a state-of-the-art DL model.

2 Literature Review

A wide range of conventional ML and DL algorithms have been employed to estimate the age of an individual using skin images. However, prior works are mainly aimed at face age synthesis, which is the reconstruction of facial images to demonstrate the proper effects of aging. For instance, Choi et al. (2013) presented a novel self-diagnostic application for classifying biological age based on cell features associated with wrinkles. The researchers collected 834 digital microscopic skin images from 238 subjects, which consisted of their face, neck, and hands. Region of Interest (ROI) based processing was used to select a 300 x 300 pixel subset of the input images. Contrast (or histogram) stretching normalization was used to enhance the images' contrast levels and achieve histogram equalization. In order to detect wrinkles, the normalized images were converted to binary images using the Otsu technique. De-noising was then used to reduce any salt- and-pepper noises in the binary images to prevent over-segmentation during the wrinkled detection process. Their wrinkle feature extraction captured several cell features, including the length, width, depth, and cell area, to train the SVM classification model used to estimate skin age with 90% accuracy (see Table 1 below). However, they relied solely on the SVM algorithm for classification, and their approach might not be effective with other ML and DL techniques.

Zaghbani Boujneh and Bouhlel (2018) compared the performance of a kNN, Wavelet Network (WN), and a novel chronological age regression model based on a Deep Supervised Sparse Autoencoder (DSSAE) on DLA features extracted from a Stacked Autoencoder model. They used face images from the FG-NET and MORPH II databases. AdaBoost face detection and alignment algorithms were employed for ROI-based pre-processing, followed by the resizing of the images to a fixed size of 32 x 32 pixels. The DSSAE was the best model across the FG-NET and MORPH II datasets.

A few studies were found to use skin images derived from other parts of the human body other than the face. Seker et al. (2014) built a low-cost skin biological age assessment SVR model using features extracted from a multi-spectral light analysis of intensity levels of the Red, Green, Blue (RGB), and Infrared (IR) bands. It effectively captured multi-layer cellular changes due to the biological skin aging process. The study employed inner-arm skin images gathered from 136 participants between the ages of 19 and 60. Several fairly recent works have employed dorsal and palmar hand images for age estimation.

Chakrabarty and Chatterjee (2019) developed a hybrid CNN-SVM model meant to classify an individual into three age groups. Their study used dorsal hand images from the Afifi hands dataset with approximately 11*K* images of both the dorsal and palm sides of 190 subjects ranging from 18 to 75 years. Pre-processing included image binarization using the adaptive threshold technique, resizing the images to a fixed size of 1000 x 1000 pixels, and pixel scaling each image to a value between 0 and 1 by dividing by 255. The researchers used their CNN to extract the DLA features used to train the hybrid classifier.

Lin et al. (2019) created a novel multi-CNN (MCNN) architecture tailored to attribute estimation, including chronological age estimation using hand images (dorsal and palmar) from the Afifi dataset. Their pre-processing steps involved data augmentation based on random cropping and rotation to add to the existing sample, resizing the images to 224 x 224 pixels, and pixel normalization in terms of a mean of 0 and a variance of 1. DLA feature extraction was done using a three-layered sequential CNN.

Similarly, Štern Payer and Urschler (2019) sought to evaluate and compare the performance of the Random Forest (RF) and DCNN techniques in biological age estimation using different generalizations of features extracted from hand MRIs. The images were sourced from the Boltzmann Institute for Clinical Forensic Imaging 3D hand MRI dataset from 328 subjects between 13 and 25 years and a 2D X-ray dataset from the Digital Hand Atlas Database, which consisted of 835 images of participants older than 10 years. Preprocessing involved the cropping and filtering-based enhancement of age-relevant structures in the hand MRIs. The pre-processed images were then resized to 80 x 80 pixels. Štern Payer and Urschler (2019) also conducted a comparative analysis of their hand-engineered feature generation and DCNN for DLA feature extraction. The DCNN technique outperformed the RF model while using the hand-crafted, and DLA features to train the regressors.

Abderrahmane Guelzim and Abdelouahad (2020) developed a method of human age prediction based on multi-class classification techniques of up to 17 labels, ranging from 18 to 75 years. They used a novel CNN and Gated Recurrent Unit (CNN-GRU) model architecture to solve the chronological age classification problem with dorsal and palmar hand images from the Afifi dataset. Crop-edge detection was used to eliminate the white background in the hand images without impacting the ROI. The authors also resized all the images to 224 x 224 pixels and conducted data balancing by adding more samples using data augmentation.

Moon and Lee (2022) compared and evaluated the performance of several CNN-based classification techniques based on features extracted either from a segmentation image processing technique using the watershed algorithm or one of several CNN architectures. The classifiers included a NASNet-Mobile, MobileNetV2, MobileNetV3-Small and EfficientNet-B0. Dorsal hand images were collated by the researchers using a Galaxy A3 smartphone from 68 healthy subjects in the age range of 10 to 80 years. Pre-processing involved resizing the dorsal hand images to a fixed size of 256 x 256 pixels. Consequently, image segmentation was conducted on the resized images using the watershed algorithm or CNN models, including the Fully CNN (FCN-8), Unet, ResUNet++, and fusion UNet. Overall, the MobileNetV3-Small model architecture was found to be the best CNN for the biological age estimation task.

Table 1: Previous studies on Age Estimation Schemes Based on Machine-Learning and Deep-Learning Techniques Using Skin Images from Various Parts of the Human Body.

Article	Input image	Image Size (pixels)	Age	Datasets	Classification	Performance / Evaluation metrics
Choi et al. (2013)	Face, neck, and hand images	300 x 300	Biological	Private microscopic 8-bit grayscale skin image dataset.	SVM.	Accuracy: ~90%
Seker et al. (2014)	Inner-arm image	N/A	Biological	Skin images were gathered from 136 participants between the ages of 19 and 60.	SVR.	RMSE: 0.16
Zaghbani Boujneh and Bouhlel (2018)	Face image	32 x 32	Chronological	FG-NET; MORPH II.	kNN, WN, DSSAE.	FG-NET kNN/WN/DSSAE MAE: 4.88/4.92/3.92 MORPH II kNN/WN/DSSAE MAE: 5.05/4.29/3.26
Chakrabarty and Chatterjee (2019)	Hand dorsal image	1000 x 1000	Chronological	Afifi hands dataset.	CNN-SVM.	Accuracy: 100% Recall: 100%
Lin et al. (2019)	Hand image	224 x 224	Chronological	Afifi hands dataset.	MCNN.	Accuracy: ~99%

Article	Input image	Image Size (pixels)	Age	Datasets	Classification	Performance / Evaluation metrics
Štern Payer and Urschler (2019)	Hand MR image	80 x 80	Biological	3D hand MRIs; 2D X-ray dataset.	RF, DCNN.	Hand-crafted features RF/DCNN MAE: 0.51/0.21 Automated features RF/DCNN MAE: 0.23/0.20
Abderrahmane Guelzim and Abdelouahad (2020)	Hand dorsal and palmar images	224 x 224	Chronological	Afifi hands dataset.	CNN-GRU.	Accuracy: ~99% Macro F1-score: 0.99 Loss: 0.037
Moon and Lee (2022)	Hand dorsal image	256 x 256	Biological	Back-of-the-hand images collated using a Galaxy A3 smartphone from 68 healthy subjects in the age range of 10 to 80 years.	NASNet-Mobile, MobileNetV2, MobileNetV3-Small, EfficientNet-B0.	NASNet-Mobile/ MobileNetV2/ MobileNetV3-Small/ EfficientNet-B0 Accuracy: 92%/93%/94%/93% Recall: 92%/93%/94%/94% Precision: 92%/92%/94%/93% F1-score: 92%/92%/94%/93%

Note: MAE measures the average absolute difference between predicted values and the actual values (Zaghbani, Boujneh, and Bouhlel 2018); RMSE is the square root of the average of squared differences between predicted values and actual values (Seker et al. 2014); Accuracy represents the proportion of correctly classified instances out of the total instances; Recall/Sensitivity measures the proportion of actual positive cases that the model correctly identified; Precision is the proportion of true positive cases out of all the cases that the model predicted as positive; F1-score represents the harmonic mean of precision and recall and is useful when there is an uneven class distribution in the data (Moon and Lee 2022).

3 Methodology

The following section outlines the experimental setup, data pre-processing steps, image feature extraction techniques, age estimation techniques, evaluation metrics, and the overall framework employed in the study. Figure 1(see appendix) illustrates the systematic processes of undertaking the research project.

3.1 Age Estimation Algorithms

3.1.1 Region of Interest Analysis

Identifying and isolating the region of interest (i.e., the hand) was necessary since the images were captured against a uniform white background. The process involved a grayscale conversion of the original dorsal hand image to eliminate every form of colour information, leaving different shades of grey. Simple thresholding was then used to convert the grayscale image to a binary image. It is a type of image segmentation that separates the image into two regions based on a specific threshold value. Pixels in the raw image with intensity values less than or equal to 220 were set to the maximum intensity value of 255, which represents white. Conversely, pixels with intensity values greater than 220 were set to zero, which represents black. The process can be mathematically defined as:

$$B(x,y) = \begin{cases} 0 & if\ G(x,y) > T(x,y) \\ 255 & if\ G(x,y) \leq T(x,y) \end{cases} \qquad (1)$$

where $T(x,y)$ is the threshold calculated for each pixel, $B(x,y)$ is the resulting pixel on the binary image, and $G(x,y)$ is a pixel from the source grayscale image. As a result, the areas of interest were set to white against a black background.

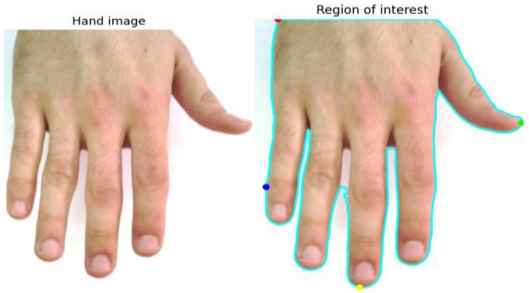


Figure 2: Example of Identifying the Region of Interest

The chain approximation contour detection method was then used to detect the borders of the hand (see the teal outline in Figure 2) from the binary image. The contour was then used to identify the extreme points (left: blue circle, bottom: yellow circle, right: green circle, and top: red circle) that define the boundary of the hand. Finding these extreme points was vital to refining the localisation of the hand regions. Once the hand was detected, a bounding box was generated around the region. The bounding box served as a spatial reference for cropping out the whole hand from the raw image. In step with the proposed framework, the bottom half of the image was then cropped out, leaving a partial of the hand for further analysis.

3.1.2 Feature Extraction

The features present in the dorsal and palmar hand images were extracted through transfer learning to extract DLA features. Specifically, representations learned by a densely connected convolution network (DenseNet201) were used to extract meaningful features from the study sample. The model was pre-trained on the ImageNet1*K* dataset, a subset of the larger ImageNet database typically used for deep-learning computer vision tasks (Russakovsky et al. 2015). As a result, it learned to isolate generic features from a wide variety of images, making it suitable for feature extraction.

DenseNet201 consists primarily of convolution, pooling, and fully connected layers. The convolution layers, which are grouped into dense blocks, are responsible for learning hierarchical representations of the input images (Huang et al. 2022). Unlike conventional CNN architectures, which tend to have their layers connected solely to the subsequent layer, DenseNet201 is essentially a feed-forward network. Each layer is connected to every other layer in a feed-forward mode, meaning they receive feature maps from all the previous layers and pass the output to all the subsequent layers. In addition, DenseNet201 concatenates feature maps from preceding layers to the current layer using skip (or shortcut) connections (Huang et al. 2017), allowing it to preserve and reuse learned features at earlier stages.

The feature extraction process involved feeding each input image through a series of dense blocks. Each block consisted of multiple convolution layers with batch normalisation and a Rectified Linear Unit (ReLU) activation function (Huang et al. 2017). A series of convolution and pooling operations were performed on the input to continuously extract features from the image. The output of the feature extraction process was a vector of numeric features that represent the partial dorsal and palmar hand images. Each element in the vector represented different levels of abstraction, starting from simple features like edges and textures to more complex information like object shapes and structures. The deep network architecture involves multiple layers of computations, typically following the pattern:

$\text{Layer}_{i+1}=\text{F}(\text{Layer}_i,\text{Layer}_{i-1},\ldots,\text{Layer}_1)$ where F denotes a composite function of operations such as convolutions, pooling, and activation functions applied to the concatenation of outputs from previous layers.

a) Dimensionality reduction was used to reduce the number of features extracted from the previous step while maintaining as much information as possible. Principal Component Analysis(PCA) can reduce the complexity of the feature space while retaining essential hand pose or gesture information. By applying PCA to the extracted features from DenseNet201, we can further decrease dimensionality while preserving important information. Figure 6 illustrates that the lower-dimensional data from the process had a total of 242 features, which is a significant drop from the 1921 DLA features from the transfer learning. It also improves model performance and reduces overfitting and computational efficiency.

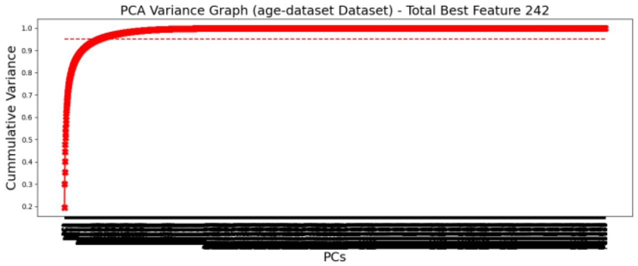


Figure 6: Explained Variance in PCA

3.1.3 Regression Algorithms

In order to evaluate the efficacy of using partial dorsal and palmar hand images for age estimation, a Support Vector Regression, Light Gradient Boosting Machine Regression (LGBMR), Random Forest (RFR), and the DenseNet201 models were trained on the dataset. Each supervised learning algorithm was used for the continuous age estimation. The SVM model served as a benchmark model, given their repeated use in several studies (Choi et al. 2013; Seker et al. 2014; Chakrabarty and Chatterjee 2019). Conversely, the preference for LGBMR and RFR algorithms stemmed from the observed research gap in their application in previous related works. Moreover, machine-learning and deep-learning algorithms could be applied for continuous age estimation.

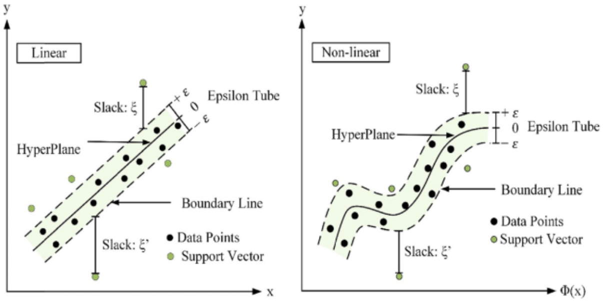
a) Support Vector Machine

Support Vector Machine handles high-dimensional data and has flexibility in using different kernel functions, It has been commonly used in various fields, including computer vision, bioinformatics, and finance (Choi et al. 2013). For classification tasks, SVM works by identifying the hyperplane that best divides the classes in the feature space within the maximum margin. For regression tasks, it predicts continuous values by finding the hyperplane that fits within a specified margin of tolerance(epsilon) from the actual data points, aiming to be as flat as possible (Seker et al. 2014). It uses a kernel function to transform the input space into a higher-dimensional space, where it might be easier to find a separating hyperplane (Chakrabarty and Chatterjee 2019). For linear regression function is as follows

$$f(x) = wx+b$$

Then the equation of decision boundary becomes w x+b = +a, wx+b = -a hyperplane that satisfies the SVR **-a < Y-wx+b < +a**.

- w is the weight vector.
- x is the input feature vector.
- *b* is the bias term.



(The diagram is available at: Ditthakit, P., Pinthong, S., Salaeh, N., Weekaew, J., Thanh Tran, T. and Bao Pham, Q. (2023). Comparative study of machine learning methods and GR2M model for monthly runoff prediction. *Ain Shams Engineering Journal*, 14(4), p.101941. doi:https://doi.org/10.1016/j.asej.2022.101941.)

The formula for non-linear regression where is the Radial Basis Function kernel which involves mapping input data into high-dimensional space a linear regression function is applied.

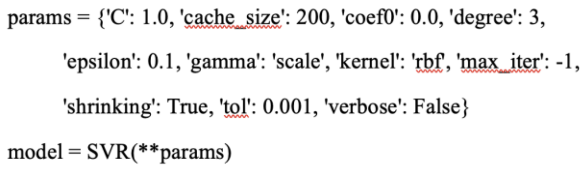
$$y=\sum_{i=1}^{n} (\alpha_i - \alpha_i{}^*) \exp(-\gamma \| x_i - x \|^2)+b$$

$$\gamma = \frac{1}{n.X.var()}$$

where n is the number of features and X.var() is the variance of the input features.

- α_i and α_i* are the Lagrange multipliers.
- b is the bias term.
- x_i are the support vectors.
- *x is the input vector for which we want to predict the output*

SVR is customised as follows:

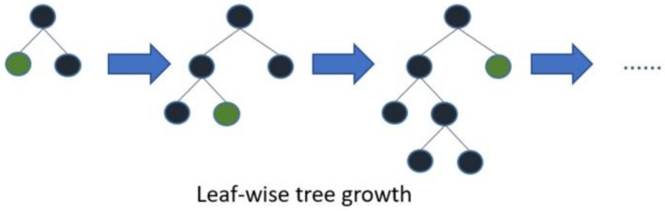


```
params = {'C': 1.0, 'cache_size': 200, 'coef0': 0.0, 'degree': 3,
        'epsilon': 0.1, 'gamma': 'scale', 'kernel': 'rbf', 'max_iter': -1,
        'shrinking': True, 'tol': 0.001, 'verbose': False}
model = SVR(**params)
```

The model utilizes the specified parameters to determine how it will fit the training data and make predictions. when model. fit(X_train, y_train) is called the SVR model works to find the optimal hyperplane in a high-dimensional space that best fits the training data, while still staying within a margin of tolerance controlled by a parameter called epsilon. The regularization parameter C helps balance the trade-off between fitting the data closely and maintaining a smooth model, generalizable. RBF kernel is employed to compute the similarity between pairs of input points in a transformed features space, enabling the capture of non-linear relationships between feature and target variables.

b) Light Gradient Boosting Machine

LGBM is an ensemble learning method, and it is widely known for its use in real-world applications that require high predictive accuracy and robustness against overfitting, such as feature extraction (Heaton 2016). LightGBMR uses a technique called gradient boosting. It builds multiple weak learners, usually decision trees, sequentially, where each base estimator corrects the errors of the previous one, resulting in a more accurate predictive model, it minimizes a loss function (e.g., mean squared error for regression and cross-entropy for classification) by sequentially adding new estimators to the model (Bansal et al. 2021). It employs a leaf-wise development strategy instead of level-wise growth until reaches the specified depth or leaf nodes, where each split is made only on the leaf node that has the highest reduction in loss. Additionally, it uses two novel techniques Gradient-based One-side Sampling(GOSS) and Exclusive Feature Bundling(EFB) to reduce memory consumption.



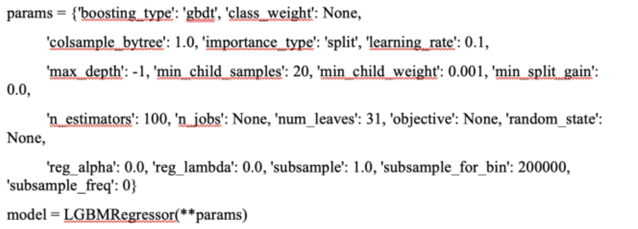
(Diagram is available at : Zahra Faska, Lahbib Khrissi, Khalid Haddouch and Nabil El Akkad (2023). A robust and consistent stack generalized ensemble-learning framework for image segmentation. *Journal of Engineering and Applied Science*, 70(1). doi:https://doi.org/10.1186/s44147-023-00226-4.)

The basic equation for gradient boosting involves iteratively minimizing the loss function by adding weak learners (decision trees in the case of LightGBM) to the ensemble. Here's the prediction function for gradient boosting is

$$\hat{y}=\sum_{t=1}^{T}\eta \mathrm{ht}(x)$$

- T is the number of trees(weak learners) in the ensemble (n_estimators=100).
- η is the learning rate which contributes to each tree to the final prediction(learning_rate=0.1).
- $\mathrm{ht}(x)=h_t(\ x)$ is the prediction from the t-th decision tree.

Each tree $h_t(\ x)$ is fitted to the negative gradient of the loss function of the previous ensemble of trees, typically starting with an initial prediction.
LightGBM Regression is customised as follows:

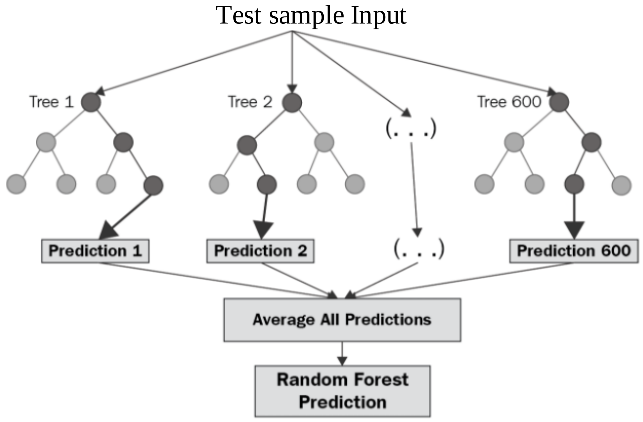


```
params = {'boosting_type': 'gbdt', 'class_weight': None,
          'colsample_bytree': 1.0, 'importance_type': 'split', 'learning_rate': 0.1,
          'max_depth': -1, 'min_child_samples': 20, 'min_child_weight': 0.001, 'min_split_gain':
0.0,
          'n_estimators': 100, 'n_jobs': None, 'num_leaves': 31, 'objective': None, 'random_state':
None,
          'reg_alpha': 0.0, 'reg_lambda': 0.0, 'subsample': 1.0, 'subsample_for_bin': 200000,
'subsample_freq': 0}
model = LGBMRegressor(**params)
```

The model utilizes the specified parameters and provides various regularization techniques to prevent overfitting. max_depth controls the maximum depth of the individual trees to limit model complexity min_child_weight controls min sum of instance weight needed in a child for further partitioning and reg_alpha and reg_lamda L1 and L2 regularization terms to penalize large coefficients. Each tree in the ensemble makes a prediction by traversing from the root to a leaf node. The final prediction is made by aggregating the predictions from all the trees in the ensemble.

c)Random Forest

RF is a supervised learning algorithm and bagging technique that uses an ensemble learning method for regression in machine learning. The trees in Random Forest Regression run in parallel, meaning there is no interaction between these trees while building the trees. It is commonly used in fields such as marketing, fraud detection, and medicine because it reduces the risk of overfitting, offering a high level of accuracy. It recursively splits the data based on feature attributes to create a tree-like structure where each internal node represents a feature, each branch represents a decision based on that feature (Heaton 2016), and each leaf node represents a class label (in classification) or a continuous value (in regression). The algorithm then selects the most informative feature at each node based on impurity measures such as Gini impurity or entropy. It continues the process recursively until a stopping criterion is met (Zhou and Feng 2017), such as reaching a maximum depth or a minimum number of samples per leaf.



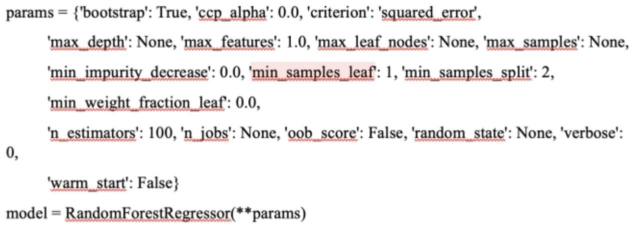
(Bakshi, C. (2020). *Random Forest Regression*. [online] Medium. Available at: https://levelup.gitconnected.com/random-forest-regression-209c0f354c84.)

The formula for the predicted function is:

$$\hat{y}=\frac{1}{T}\sum_{t=1}^{T}\mathrm{ht}(x)$$

- T is the number of trees (n_estimators=100).
- $\mathrm{ht}(x)=\mathrm{h}_t(x)$ is the prediction from the t-th decision tree.

Random Forest Regression is customised as follows:

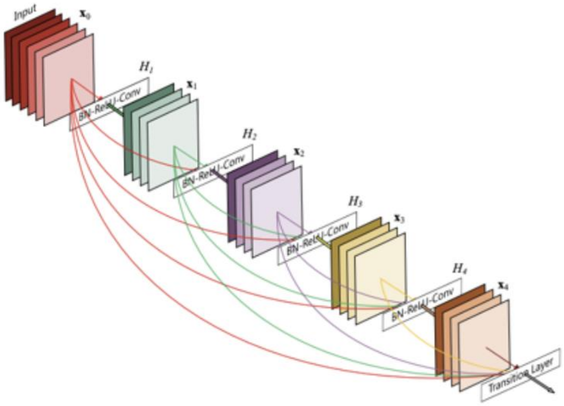


```
params = {'bootstrap': True, 'ccp_alpha': 0.0, 'criterion': 'squared_error',
          'max_depth': None, 'max_features': 1.0, 'max_leaf_nodes': None, 'max_samples': None,
          'min_impurity_decrease': 0.0, 'min_samples_leaf': 1, 'min_samples_split': 2,
          'min_weight_fraction_leaf': 0.0,
          'n_estimators': 100, 'n_jobs': None, 'oob_score': False, 'random_state': None, 'verbose':
0,
          'warm_start': False}
model = RandomForestRegressor(**params)
```

n_estimators create n number of trees in the forest, each tree being trained on a different random sample of the training data is called bootstrap sampling/bagging. Each tree is grown independently to its full depth (unless restricted by other parameters like 'max_depth', min_sample_split', etc) is called randomization. During the construction of each tree, at each split in a tree, a random subset of features is considered for splitting, which randomization helps in making the trees more diverse and reduces overfitting by introducing variability in the models. For a new input, each tree in the forest makes predictions. The final prediction of the Random Forest is the average of the predictions from all the trees in the forest.

d)DenseNet201

The DenseNet201 model used for age estimation employs the same algorithm used for DLA feature extraction. Convolutional network architecture is known as densely connected conventional networks. Each dense block takes not only the output of the previous block as input but also the outputs of all the preceding blocks. It has 201 layers and has high computation demands during the training and inference phases. It is used in computer vision tasks and performs better with fewer parameters and helps in overfitting.

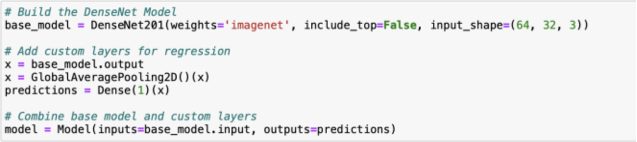


DenseNet Structure

$$a^{[l]} = g([a^{[0]}, a^{[1]}, a^{[2]}, \dots\dots\dots\dots, a^{[l-1]}])$$

(diagram available at: Baldha, S. (2022). *Introduction to DenseNets (Dense CNN)*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2022/03/introduction-to-densenets-dense-cnn/.

The DenseNet201 model was customized for the age estimation project as follows:



```
# Build the DenseNet Model
base_model = DenseNet201(weights='imagenet', include_top=False, input_shape=(64, 32, 3))

# Add custom layers for regression
x = base_model.output
x = GlobalAveragePooling2D()(x)
predictions = Dense(1)(x)

# Combine base model and custom layers
model = Model(inputs=base_model.input, outputs=predictions)
```

The DenseNet201 model is initialized with pre-trained weights trained for regression. Data is split 80:20 into training and testing sets. Adding a new fully connected dense layer at the end of the network with a single neuron for predicting a single numerical value. The model is compiled using Adam optimizer and MSE loss. It's trained for 15 epochs with a batch size of 32 and evaluated on validation data. Finally, its performance is assessed using MSE, RMSE, R squared score, and test loss.

3.2 Dataset

The “11K Hands” dataset collated by Afifi (2019) was used in this study. It consisted of a collection of 11,076 dorsal and palmar hand images from 190 subjects between the ages of 18 and 75 years. Therefore, each person had an average of 58 photographs taken of their hands at different orientations. Each image looks to have been taken uniformly from the camera against a white background. The 1600 x 1200 pixels images comprised both the dorsal and palmar sides of the right and left hands. Accompanying information, such as the participant’s true age, was included in a text file that had metadata associated with each image file.

3.2.1 Data Augmentation

Data augmentation was then used on the partials to enhance the diversity and size of a dataset by applying various transformations to the original data. Three types of image transformations were employed:

- Horizontal flipping – This transformation is implemented using the “Fliplr” method from the Python Imaging library. It horizontally flips images with a probability randomly chosen from the specified range (0.5 to 1.0). Here the probability of 0.5 means that there is a 50% chance that an image will be flipped horizontally.
- Rotation – The “Affine” method is used to rotate images by a specified range of degrees. Here the images can be rotated clockwise or counterclockwise within the range of -20 to 20 degrees. This parameter controls the extent of rotation applied to the images.
- Gaussian Blur – Gaussian blur is applied using the “GaussianBlur” method. It smoothens the images by averaging the pixel values within a neighbourhood defined by the standard deviation (sigma). The sigma parameter is randomly chosen from the specified range of 0 to 1.0, controlling the amount of blur applied to the images.

By combining these all three augmentation techniques, we introduce approximately 3000 variability into the dataset, which helps to improve the model’s performance and generalization ability. The ranges specified for each parameter allow for flexibility in the augmentation process, enabling the generation of a diverse set of augmented images while preserving the original dataset’s characteristics.

Original Images

Cropped Images

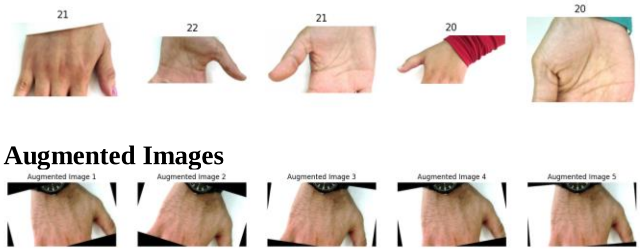


Figure 3: Sample Images from Data Augmentation

3.2.2 Data Balancing

Due to the skewed nature of the age data, there was a severe age-distribution imbalance. Looking at Figure 4 below, it appears that most of the images taken were from subjects aged 26 years and younger.

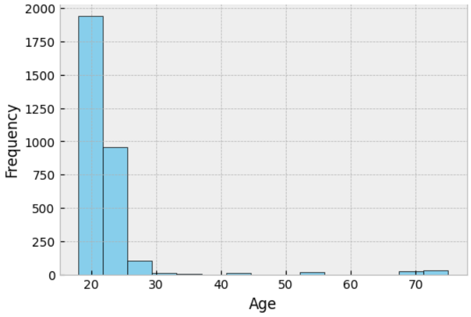


Figure 4: Distribution of Subject's Ages

In order to achieve a balanced distribution, the analysis only considered records from the ages of 19 to 26 years(having more than two persons hand images). The dataset was then downsampled per the minority class, which had a total of 795 images (see Figure 5) considered data augmentation techniques for 19,24, 25 & 26 years.

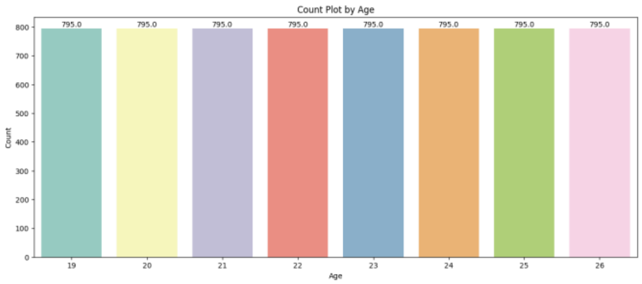
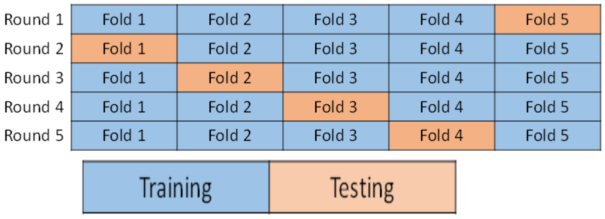


Figure 5: Distribution of Balanced Age Dataset

3.3 Experimental Setup

The experiment was implemented using Python(v3.10.12) on Google Colab, which had 54.8GB of available RAM to support the execution of the experiment. The image processing was conducted using OpenCV(v.4.8.0), is a Python computer vision library. A transfer learning model used for feature extraction was deployed using TensorFlow(v.2.15.0), an open-source library commonly used to develop computer vision and Natural Language Processing (NLP) applications. Keras(v.2.15.0), which is the front end of TensorFlow, was used to build high-level neural networks. It streamlined the process of building and fitting the DL models via a simpler interface and a wide range of pre-trained models. The Scikit-Learn(v.1.2.2) machine-learning library was used to develop and evaluate the performance of the regression used for age estimation.

The k-fold cross-validation technique was used in the experiment. In each iteration, the features and label set was divided into k = 5 folds using a stratified sampling approach, ensuring that the data distribution was maintained. During each iteration, each fold was split four to one, with the four folds being used for training while the remaining fold was used to gauge the models' performance (see Figure 7). During each iteration, features were extracted from DenseNet201 and these extracted features then underwent dimensionality reduction using PCA to retain 95% of the variance.The resulting predictions were compared against the actual target values to compute the evaluation metrics. These metrics were subsequently stored for each fold and iteration. After completing all the iterations, the results from each fold were aggregated. The evaluation metrics used included the MSE, RMSE and R squared score. It should be noted that the 5-fold cross-validation was not used to evaluate the DenseNet201 model here the data was split 80% for training and 20% for testing for age estimation.



Round 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

Training Testing

Figure 7: 5-Fold Cross-Validation Procedure

3.3.1 Performance Metrics

The average MSE, RMSE, and R^2 scores were calculated across all folds for each iteration, providing a comprehensive assessment of the model's performance.

Mean Squared Error is the average of the squared difference between predicted and actual values. The MSE was calculated as follows:

$$MSE = \frac{1}{N}\sum_{i=0}^{N}(y_i - \hat{y}_i)^2 \tag{2}$$

where N is the sample size, and $\hat{y}_i$ and y_i are the true and predicted age, respectively.

The RMSE represents the square root of the MSE. It was used to measure the standard deviation of the residuals, it is the difference between the actual and predicted age values. The metric was calculated as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum\nolimits_{i=0}^{N}(y_i - \hat{y}_i)^2} \tag{3}$$

R^2 Score or Coefficient of Determination refers to the proportion of the variance in the dependent variable, which the regression models explain. It is a scale-free score, i.e., irrespective of the values being small or large, the value of the R^2 Score will be less than one.

$$R^2 = 1 - \left[\frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y}_i)^2}\right] \tag{4}$$

where $\bar{y}$ is the average of the actual age values in the dataset. The metric was used as a measure of how each of the models could predict the age of the subjects from the partial dorsal and palmar hand skin images.

4 Results

The following section presents the results of the training and performance evaluation of the Support Vector Regression, Light Gradient Boosting Machine Regression, Random Forest Regression, and DenseNet201 models. The ML algorithms, i.e., SVR, LGBMR, and RFR, were fitted and evaluated using a 5-fold cross-validation approach. Hence, the feature and label set was split into five folds were used on the models over five iterations. In the case of the DenseNet201, the data was split 80% for training and 20% for testing. it took 15 epochs to fit and evaluate its performance.

4.1 Support Vector Regressor

Figure 8 illustrates the performance of the SVR model by the cross-validation iteration and fold. It indicates that the MSE and RMSE were directly proportional to each other, meaning that they increased and decreased together at a constant ratio. On the other hand, the R^2 score was inversely proportional to the MSE and RMSE. Hence, an increase in MSE and RMSE corresponded to a decrease in the R^2 score.

Figure 9 shows that the model's performance was fairly consistent across the five iterations. However, Figure 8 indicates that it varied across the five folds, with the model testing poorly on fold 2 relative to the other folds. It appears to have tested best on folds 1 and 4, evidenced by the low MSE and RMSE and high R^2.

Table 2 presents the average MSE, RMSE, and R^2 score of the SVR model after each iteration. The SVR model had an MSE of about 1.33 years and RMSE of 1.15 years. The R^2 of .75 indicates that the feature set explained approximately 75% of the variance in the age values.

Table 2: Average SVR Performance by Iteration

Iteration	MSE	RMSE	R^2
1	1.33	1.15	.75
2	1.33	1.15	.75
3	1.32	1.15	.75
4	1.33	1.15	.75
5	1.33	1.15	.75
Overall	**1.33**	**1.15**	**.75**

4.2 Light Gradient Boosting Machine Regressor

Figure 10 illustrates the performance of the LGBMR model by the cross-validation iteration and fold. It shows that the MSE and RMSE maintained a direct proportion relationship. Likewise, the R^2 score remained inversely proportional to the MSE and RMSE. Hence, an increase in MSE and RMSE corresponded to a decrease in the R^2 score.

Figure 11 shows the somewhat consistent performance of the LGBM model across the five iterations. But, it varied across the five folds, with the LGBM model testing poorly on fold 2 while testing best on fold 1, which appears to have the lowest MSE and RMSE values and highest R^2 scores.

Table 3 outlines the average MSE, RMSE, and R^2 score of the LGBM model after each iteration. The SVR model had an MSE of about 1.56 years and RMSE of 1.25 years. The R^2 of .70 indicates that the feature set explained approximately 70% of the variance in the age values.

Table 3: Average LGBMR Performance by Iteration

Iteration	MSE	RMSE	R^2
1	1.58	1.26	.70
2	1.57	1.25	.70
3	1.57	1.25	.70
4	1.58	1.26	.70
5	1.54	1.24	.71
Overall	**1.56**	**1.25**	**.70**

4.3 Random Forest Regressor

Figure 12 illustrates the performance of the RFR model by the cross-validation iteration and fold. Like the SVR and LGBM models, the RF's MSE and RMSE were directly proportional to each other. On the other hand, the R^2 score was inversely proportional to the MSE and RMSE.

Figure 13 shows that the model's performance was fairly consistent across the five iterations. However, Figure 8 indicates that it varied across the five folds, with the model testing poorly on fold 2 relative to the other folds. It appears to have tested best on fold 4, evidenced by the low MSE and RMSE and high R^2.

Table 4 presents the five-iteration average MSE, RMSE, and R^2 scores of the RFR model. The RFR model had an MSE of about 1.90 years and an RMSE of 1.38 years. The R^2 of .64 indicates that the feature set explained approximately 63% of the variance in the age values.

Table 4: Average RF Performance by Iteration

Iteration	MSE	RMSE	R^2
1	1.93	1.39	.63
2	1.91	1.38	.64
3	1.90	1.38	.64
4	1.93	1.39	.63
5	1.91	1.38	.64
Overall	**1.90**	**1.38**	**.64**

4.4 DenseNet201

Figure 14 illustrates the model performance of the pre-trained DenseNet201. The model's performance fluctuates up to the seventh epoch before plateauing. This suggests that the model is learning the patterns in the training data effectively and is likely, not overfitting, as the performance stabilizes.

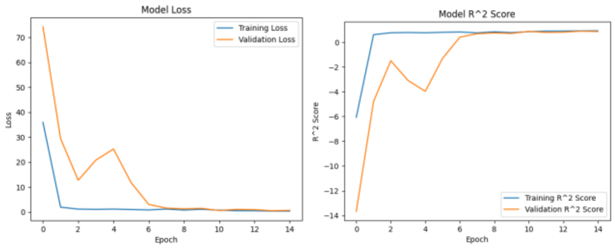


Figure 14: Training and Validation Performance

Table 5 outlines the MSE, RMSE, R^2 of the DenseNet201 model. It had an MSE of about .66 years and an RMSE of .81 years. Consequently, the feature set explained about 87% of the variability in the age values. As a result, the DenseNet201 model exhibits strong generalisation ability and performs well not only on training data but also on unseen data.

Table 5: DenseNet201 Performance

MSE	RMSE	R^2
0.66	0.81	.87

5 Discussion

The findings demonstrate that the Support Vector Regression algorithm performed slightly better than the Light Gradient Boosting Machine Regression and Random Forest Regression algorithms in age estimation. On the other hand, the DenseNet201 significantly outperformed the other models in continuous age estimation, with an MSE of 0.66 years and RMSE of 0.81, indicating its effectiveness in capturing complex relationships in the data. Similarly, the DenseNet201 captured a higher proportion (87%) of the variance in the age values, accounting for the number of features used to fit the model, than the conventional ML algorithms of SVR, RFR, and LightGBMR.

Distinguishing characteristics, such as wrinkles, skin texture, and vein prominence, were found to be present in the hand portion and may be related to aging. These characteristics could be relevant across different ethnic and demographic groupings, resulting in models that more effectively generalize to a heterogeneous population. Additionally, it was discovered that there were notable differences in the positioning and orientation of the dorsal or palmar hand images. Standardising the region of interest to the partial of the hand helped reduce the variability in the dataset and develop more robust age estimation models.

Furthermore, processing the entire palmar or dorsal hand image makes use of more computer power than processing just the partial. Reducing the computational complexity by narrowing down the ROI was intended to speed up the age estimation models' training and inference processes. Specifically, it made it possible for the DenseNet201 model that had already been trained to extract more task-relevant features, which could improve performance in downstream analysis tasks. Furthermore, focusing on partial images might improve privacy protection by preventing the analysis and storage of complete dorsal and palmar hand images that may contain identifiable details like fingerprints, nail shape, and texture.

The results of this study provide insights into the development of age estimation using partial dorsal and palmar hand images as input. In particular, employing a mix of hand detection and feature extraction with transfer learning to facilitate extracting meaningful features that could be used to build models that perform better or have comparable performance to well-established techniques. The findings supported the inference that the partial of the dorsal and palmar hand images contain distinctive features such as wrinkles, skin texture, and prominent veins that are relevant to ageing. Furthermore, the approach was integral to standardising the input images, which helped reduce variability in the dataset and contributed to the development of more robust models. The image normalisation and resizing steps further expedited the feature extraction by standardising the intensity levels and image sizes across the dataset. The goal of resizing the images to 32 x 64 pixels was to reduce the computational complexity of the models to allow for faster training and inference processes for the models.

Despite the smaller image resolution, the performance of the regressors was comparable to that of previous works that used larger images. The quality of features extracted through transfer learning contributed to the superior performance of the predictive models. Feature extraction using the pre-trained DenseNet201 enabled the extraction of hierarchical representations of dorsal and palmar hand images. The deep-learning model's dense connectivity and skip connections allowed it to gradually capture complex patterns and features relevant to age estimation and classification.

The performance difference between the ML and DL algorithms is consistent with the results of previous studies that conducted a comparative analysis of the two. For instance, Štern Payer and Urschler (2019) found that their DCNN performed better at age estimation than an RF model. Similarly, Zaghbani Boujneh and Bouhlel (2018) reported that their novel DSSAE model architecture outperformed more conventional age estimation techniques, such as WN and kNN. These studies noted that DL techniques use a network of interconnected layers to learn and understand the complex patterns in the skin images used as input. Consequently, they scale with data, whereas the ML algorithms tend to plateau at a certain performance threshold. Traditional ML techniques, such as the SVR, LGBMR, and RFR, also lack complexity in their algorithms, which results in lower accuracy in outcome predictions. In addition, ML models have scalability drawbacks due to their limited number of input variables, which makes it harder to build robust and scalable systems.

DenseNet201 can extract intricate patterns from images and offers a robust approach to age estimation. However, It may encounter some challenges when faced with partial dorsal and palmar hand images containing unknown features not well represented in the training data. Factors such as lighting conditions, watches, rings, diverse hand poses or orientations, and duplicate images as well may also pose difficulties for accurate age estimation. While machine learning algorithms can extract higher dimensional features that enhance the accuracy of their predictions, they may struggle to capture the intricate details present in images compared to deep learning models (DenseNet201). Nonetheless, inaccuracies may arise when dealing with poor quality and duplicate images.

However, DL models usually require more data to improve their performance, whereas ML algorithms tend to rely on less, given their underlying data structures. DL algorithms generally need more computational resources, such as storage, RAM, and graphical or computational processing units, to both train and the subsequent use of the capabilities learned, i.e., inference, due to their intrinsic complexity (Hu et al. 2017). Therefore, the performance of DL techniques could be hampered in situations where the datasets are small and simple, and there is a lack of adequate computational resources to allow the fitting and subsequent use of the models.

6 Conclusion

Numerous studies on the skin surface microstructure report an association between aging and changes in skin characteristics (Choi et al. 2013; Seker et al. 2014; Moon and Lee 2022). Some analyzed the relationship between skin conditions and aging by determining characteristics, such as length, width, and area of wrinkled cells from human skin (Choi et al. 2013). Consequently, researchers recognized that these skin features could be extracted to conduct automated age estimation using AI, particularly ML algorithms, such as the SVR, PART, PLS, and kNN. Recently, deep CNNs have been found to be very effective in various computer vision studies. One of the main advantages of conventional ML algorithms was the ability to extract higher dimensional features that scaled up the accuracy of their predictions. Thus, researchers used DL techniques to extract DLA features from skin images that could be used to train both traditional ML models and DL models.

In previous skin-related studies, the majority of them involve face age synthesis, which is the reconstruction of facial images to illustrate the effects of aging better. Some studies use skin images derived from parts of the human body other than the face, such as the inner arm, brain, and hand MRIs. Several fairly recent related works employed dorsal and palmar hand images for automated age estimation. However, none of these studies considered using a partial of the images to reduce the higher computational requirements that come with using DL techniques for both feature extraction and actual age estimation. A partial means using about half of the image to obtain the DLA features and subsequently conduct the age estimation. Therefore, this project proposed a framework for deriving partial images from full-hand images and then extracting DLA features from them using a pre-defined deep CNN model.

It involved an ROI analysis to capture the hand via the contour detection approach. The bottom half of the image was then cropped out, leaving a partial of the hand for further analysis. The study also used data augmentation to increase the variety of the dataset while still retaining the pertinent information. In order to address the data distribution imbalance in the age values, the data was downsampled to match the number of records in the minority class. A DenseNet201 was then used to extract DLA features from the partial. Dimensionality reduction was vital to reducing the high dimensional feature set to a subset of the most important features used to conduct a comparative analysis of three conventional ML algorithms, SVR, LGBMR, and RFR, and one DL model, the DenseNet201.

The SVR model served as a baseline model owing to how popular its use was in previous studies on age estimation. The RFR and LGBMR models performed slightly worse than the SVR technique, which could explain why the algorithms were rarely employed in prior related works. In comparison, the DenseNet201 model performed significantly better than the SVR, evidenced by its lower MSE and RMSE and higher R^2 score. The findings indicate that DL techniques perform better in general than ML algorithms in age estimation. More importantly, the use of the pre-trained model demonstrates the effectiveness of knowledge transfer, which served to improve the accuracy of the models in the target domain, especially with the limited training set size.

More importantly, the partials of the dorsal and palmar hand images appear to have had more distinctive features that are relevant to ageing, like wrinkles, skin texture, and the prominence of veins. Using the features enhanced the accuracy of the estimation models, resulting in competitive performance metrics. In addition, the features are applicable across different demographic groups and ethnicities, and this improved the generalization capability of the models, mitigating the potential for bias. Moreover, partials could help protect an individual's privacy by preventing the analysis or storage of full hand images that might contain more identifiable features, such as fingerprints, nail shape, and texture.

Narrowing down the ROI allows pre-trained models like the DenseNet201 to extract features that are more relevant to the age estimation task effectively. Pertinent features enhance the performance in downstream analysis tasks, like regression, and dimensionality reduction. Also, in a real-world application, partials would make data annotation easier and less time-consuming than dealing with an entire hand image. Storing the partial hand images also consumes less space than full hand images, resulting in more efficient management of large datasets.

One of the limitations of the study is that it used a limited dataset with a small sample size (n = 795) of ages ranging from 19 to 26 years. It might not have adequately represented the population. The lack of diversity could have resulted in a biased age estimation model that performs better on certain groups than others. Additionally, the study was limited to a single dataset, lacking a comprehensive validation on external datasets, which could impact the generalizability of the findings. The limitation was also observed in other similar studies on age estimation using dorsal and palmar hand images (Ray and Chakrabarti 2019; Lin et al. 2019; Abderrahmane, Guelzim, and Abdelouahad 2020), and it is due to the lack of reliable public databases of hand images.

In the future, the project should collect a more diverse dataset with a wide range of hand images from various demographics and ethnicities to help improve the generalizability of the age estimation models. Also, it could incorporate a thorough validation study in real-world settings to assess the performance of the models across different scenarios and populations. Investigating additional features present in full-hand images and exploring their relevance to aging could lead to more comprehensive age estimation models. Addressing the limitations of the study and following through on the recommendations would help advance the field of age estimation using partial hand images while ensuring that all practical considerations are adequately addressed.

7 References

Abderrahmane, Mohamed Ait, Ibrahim Guelzim, and Abdelkaher Ait Abdelouahad. 2020. "Human Age Prediction Based on Hand Image Using Multiclass Classification." In 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI), 1–5. IEEE. doi:10.1109/ICDABI51230.2020.9325591.

Afifi, Mahmoud. 2019. "11K Hands: Gender Recognition and Biometric Identification Using a Large Dataset of Hand Images." *Multimedia Tools and Applications*, March, 1–20. doi:10.1007/s11042-019-7424-8.

Alarifi, Jhan, John Fry, Darren Dancey, and Moi Hoon Yap. 2019. "Understanding Face Age Estimation: Humans and Machine." In *2019 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 1–5. IEEE. doi:10.1109/CITS.2019.8862107.

Bansal, Monika, Munish Kumar, Monika Sachdeva, and Ajay Mittal. 2021. "Transfer Learning for Image Classification Using VGG19: Caltech-101 Image Data Set." *Journal of Ambient Intelligence and Humanized Computing*, September, 1–12. doi:10.1007/s12652-021-03488-z.

Bakshi, C. (2020). *Random Forest Regression*. [online] Medium. Available at: https://levelup.gitconnected.com/random-forest-regression-209c0f354c84.

Baldha, S. (2022). *Introduction to DenseNets (Dense CNN)*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2022/03/introduction-to-densenets-dense-cnn/.

Chakrabarty, Navoneel, and Subhrasankar Chatterjee. 2019. "A Novel Approach to Age Classification from Hand Dorsal Images Using Computer Vision." In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 198–202. IEEE. doi:10.1109/ICCMC.2019.8819632.

Choi, Young-Hwan, Yoon-Sik Tak, Seungmin Rho, and Eenjun Hwang. 2013. "Skin Feature Extraction and Processing Model for Statistical Skin Age Estimation." *Multimedia Tools and Applications* 64 (2): 227–47. doi:10.1007/s11042-011-0987-7.

Ditthakit, P., Pinthong, S., Salaeh, N., Weekaew, J., Thanh Tran, T. and Bao Pham, Q. (2023). Comparative study of machine learning methods and GR2M model for monthly runoff prediction. *Ain Shams Engineering Journal*, 14(4), p.101941. doi:https://doi.org/10.1016/j.asej.2022.101941.

Han, Hu, Charles Otto, Xiaoming Liu, and Anil K Jain. 2015. "Demographic Estimation from Face Images: Human vs. Machine Performance." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (6): 1148–61. doi:10.1109/TPAMI.2014.2362759.

Heaton, Jeff. 2016. "An Empirical Analysis of Feature Engineering for Predictive Modeling." In *SoutheastCon 2016*, 1–6. IEEE. doi:10.1109/SECON.2016.7506650.

Huang, Gao, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. 2017. "DenseNet Densely Connected Convolutional Networks." In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2261–69. IEEE. doi:10.1109/CVPR.2017.243.

Huang, Gao, Zhuang Liu, Geoff Pleiss, Laurens van der Maaten, and Kilian Q Weinberger. 2022. “Convolutional Networks with Dense Connectivity.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44 (12): 8704–16. doi:10.1109/TPAMI.2019.2918284.

Hu, Zhenzhen, Yonggang Wen, Jianfeng Wang, Meng Wang, Richang Hong, and Shuicheng Yan. 2017. “Facial Age Estimation with Age Difference.” *IEEE Transactions on Image Processing : A Publication of the IEEE Signal Processing Society* 26 (7): 3087–97. doi:10.1109/TIP.2016.2633868.

Lin, Yi-Chun, Yusei Suzuki, Hiroya Kawai, Koichi Ito, Hwann-Tzong Chen, and Takafumi Aoki. 2019. “Attribute Estimation Using Multi-CNNs from Hand Images.” In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 241–44. IEEE. doi:10.1109/APSIPAASC47483.2019.9023260.

Moon, Cho-I, and Onseok Lee. 2022. “Skin Microstructure Segmentation and Aging Classification Using CNN-Based Models.” *IEEE Access : Practical Innovations, Open Solutions* 10: 4948–56. doi:10.1109/ACCESS.2021.3140031.

Ray, Paramita, and Amlan Chakrabarti. 2019. “A Mixed Approach of Deep Learning Method and Rule-Based Method to Improve Aspect Level Sentiment Analysis.” *Applied Computing and Informatics*, March. doi:10.1016/j.aci.2019.02.002.

Rew, Jehyeok, Young-Hwan Choi, Hyungjoon Kim, and Eenjun Hwang. 2019. “Skin Aging Estimation Scheme Based on Lifestyle and Dermoscopy Image Analysis.” *Applied Sciences* 9 (6): 1228. doi:10.3390/app9061228.

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, et al. 2015. “ImageNet Large Scale Visual Recognition Challenge.” *International Journal of Computer Vision* 115 (3): 211–52. doi:10.1007/s11263-015-0816-y.

Seker, H, V Uslan, A B Orun, E Goodyer, and G Smith. 2014. “Prediction of Skin Ages by Means of Multi-Spectral Light Sources.” *Conference Proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society* 2014: 6736–39. doi:10.1109/EMBC.2014.6945174.

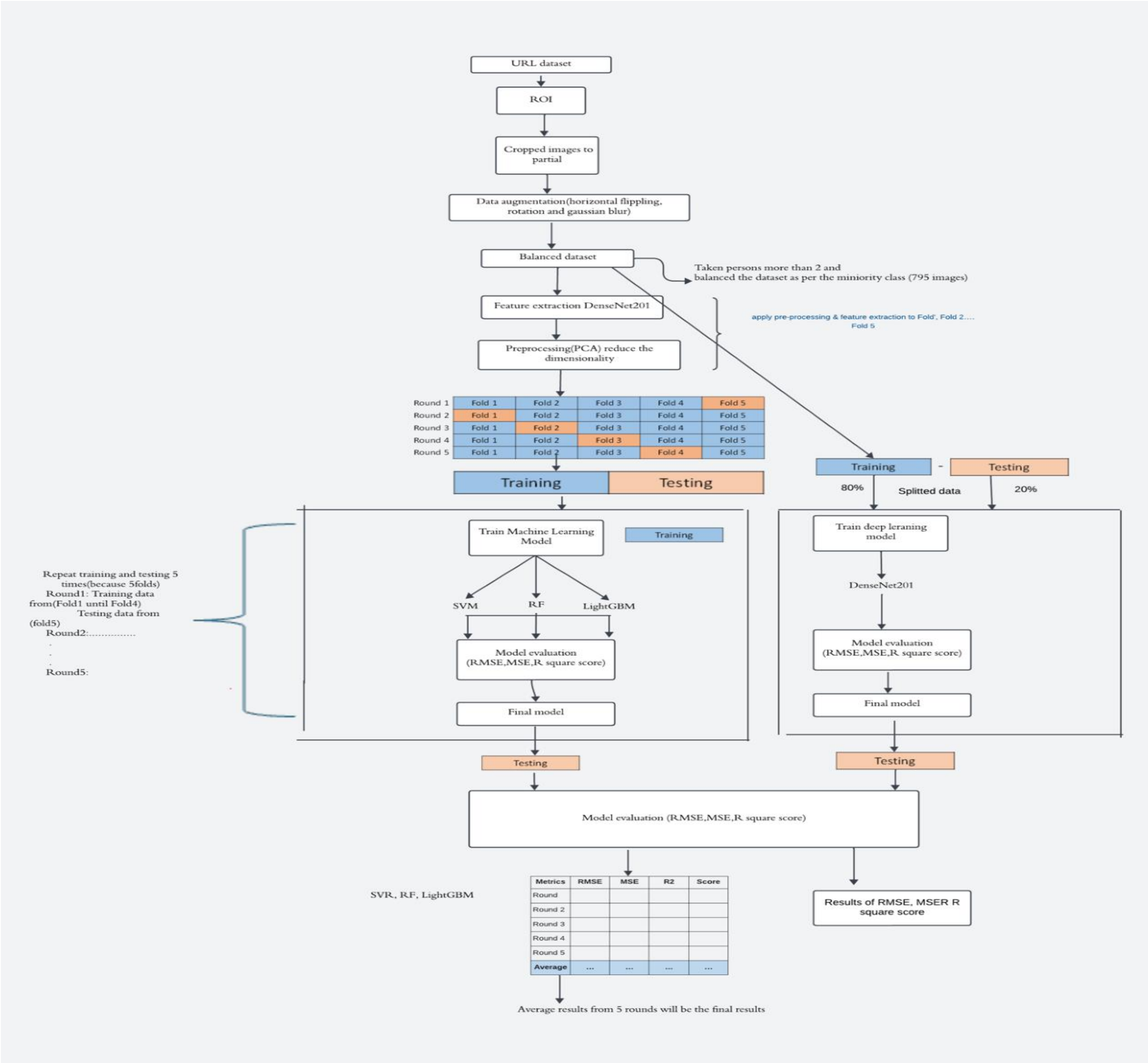
Štern, Darko, Christian Payer, and Martin Urschler. 2019. “Automated Age Estimation from MRI Volumes of the Hand.” *Medical Image Analysis* 58 (December): 101538. doi:10.1016/j.media.2019.101538.

Zaghbani, Soumaya, Noureddine Boujneh, and Med Salim Bouhlel. 2018. “Age Estimation Using Deep Learning.” *Computers & Electrical Engineering* 68 (May): 337–47. doi:10.1016/j.compeleceng.2018.04.012.

Zhou, Zhi-Hua, and Ji Feng. 2017. “Deep Forest: Towards an Alternative to Deep Neural Networks.” In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, edited by Carles Sierra, 3553–59. California: International Joint Conferences on Artificial Intelligence Organization. doi:10.24963/ijcai.2017/497.

Zahra Faska, Lahbib Khrissi, Khalid Haddouch and Nabil El Akkad (2023). A robust and consistent stack generalized ensemble-learning framework for image segmentation. *Journal of Engineering and Applied Science*, 70(1). doi:https://doi.org/10.1186/s44147-023-00226-4.)

8 Appendix


URL dataset
ROI
Cropped images to partial
Data augmentation(horizontal flippling, rotation and gaussian blur)
Balanced dataset
Taken persons more than 2 and balanced the dataset as per the miniority class (795 images)
Feature extraction DenseNet201
Preprocessing(PCA) reduce the dimensionality
apply pre-processing & feature extraction to Fold', Fold 2.... Fold 5
Training
Testing
Training
-
Testing
80%
Splitted data
20%
Train Machine Learning Model
Training
Repeat training and testing 5 times(because 5folds)
Round1: Training data from(Fold1 until Fold4)
Testing data from (fold5)
Round2:...............
.
.
.
Round5:
SVM
RF
LightGBM
Model evaluation (RMSE,MSE,R square score)
Final model
Train deep leraning model
DenseNet201
Model evaluation (RMSE,MSE,R square score)
Final model
Testing
Testing
Model evaluation (RMSE,MSE,R square score)
SVR, RF, LightGBM
Results of RMSE, MSER R square score
Average results from 5 rounds will be the final results

Round 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Round 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

Metrics	RMSE	MSE	R2	Score
Round				
Round 2				
Round 3				
Round 4				
Round 5				
Average	...	...	...	...

Figure 1: Methodological Flow Chart

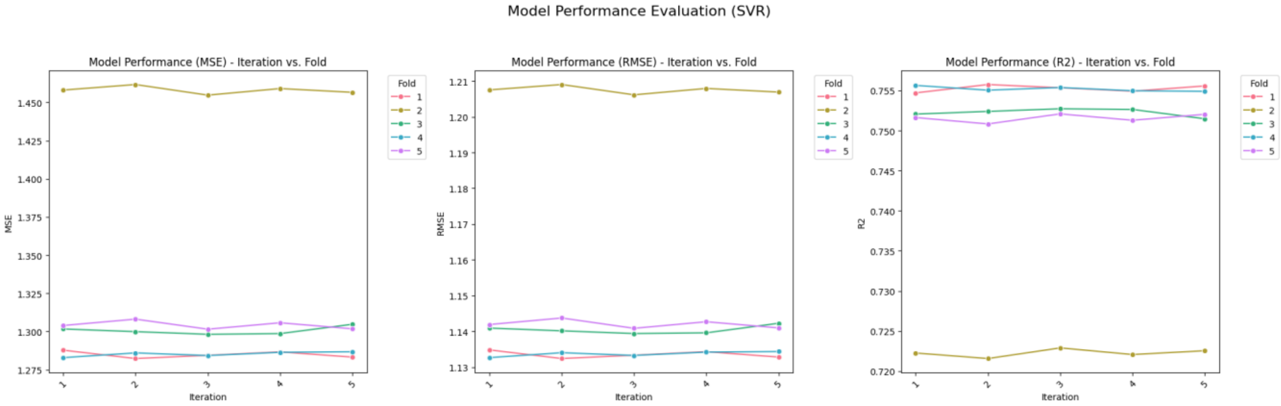


Figure 8: SVR Performance by Iteration and Fold

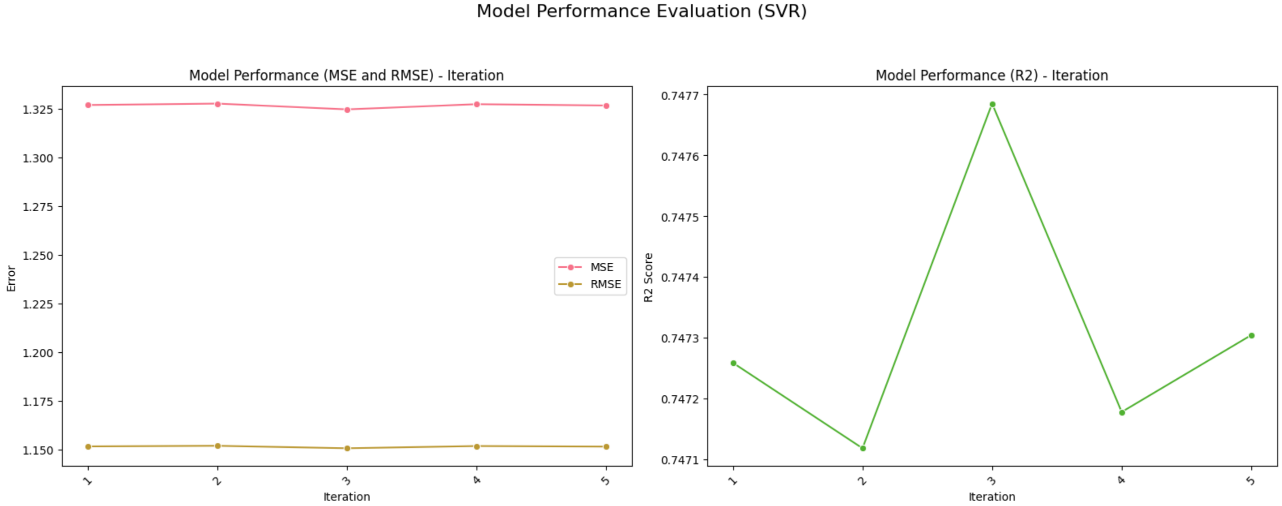


Figure 9: Average SVR Performance by Iteration

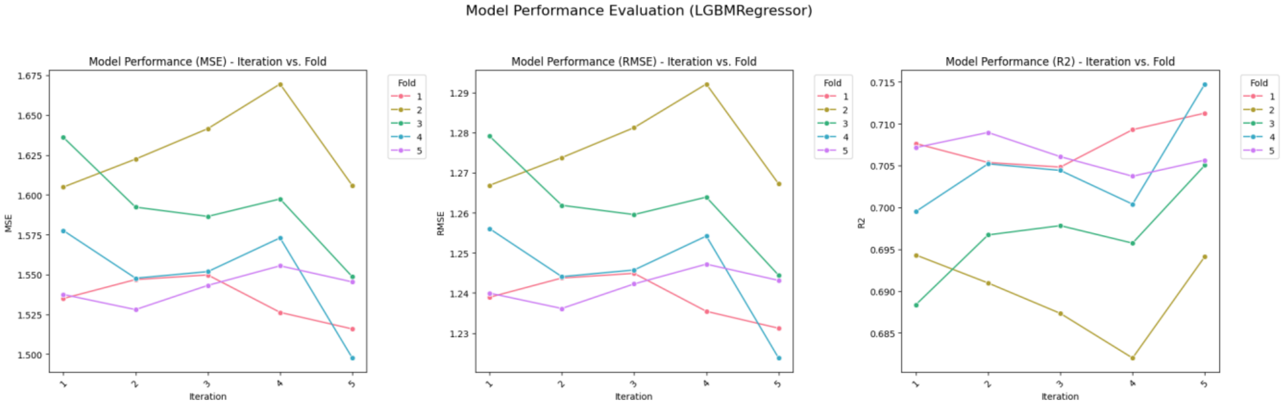


Figure 10: LGBM Performance by Iteration and Fold

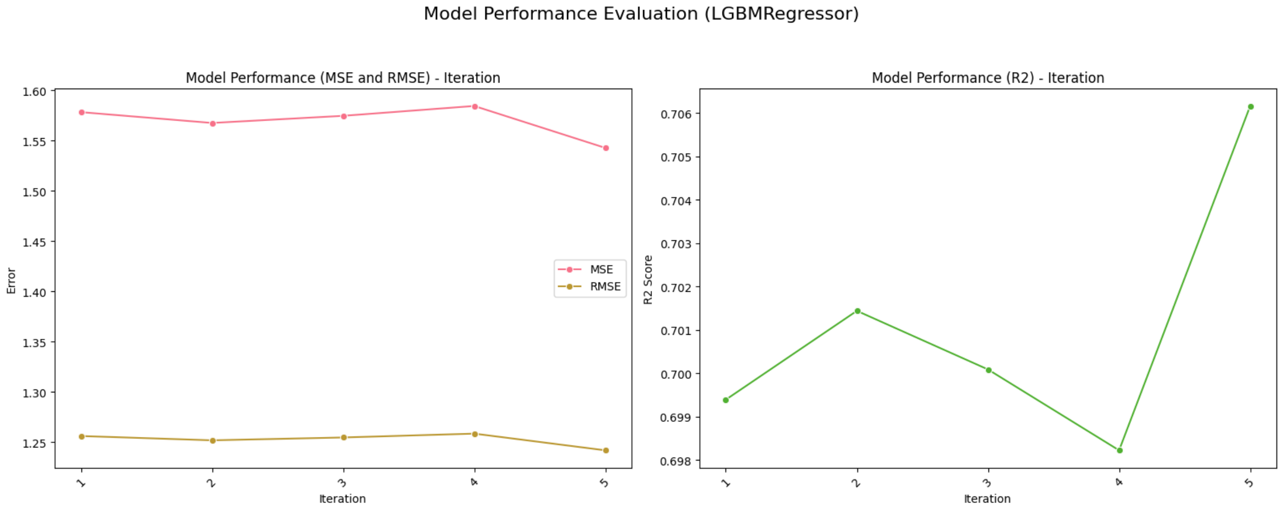


Figure 11: Average LGBM Performance by Iteration

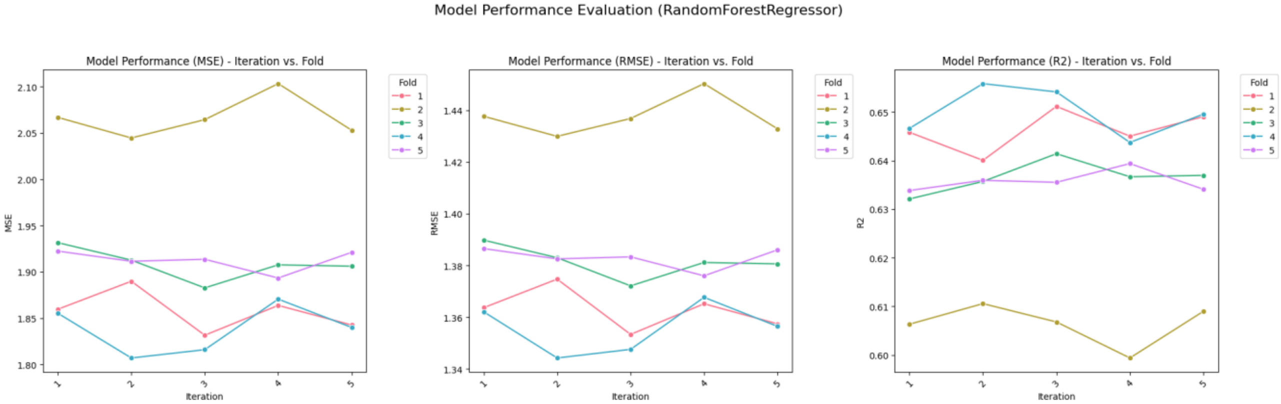


Figure 12: RF Performance by Iteration and Fold

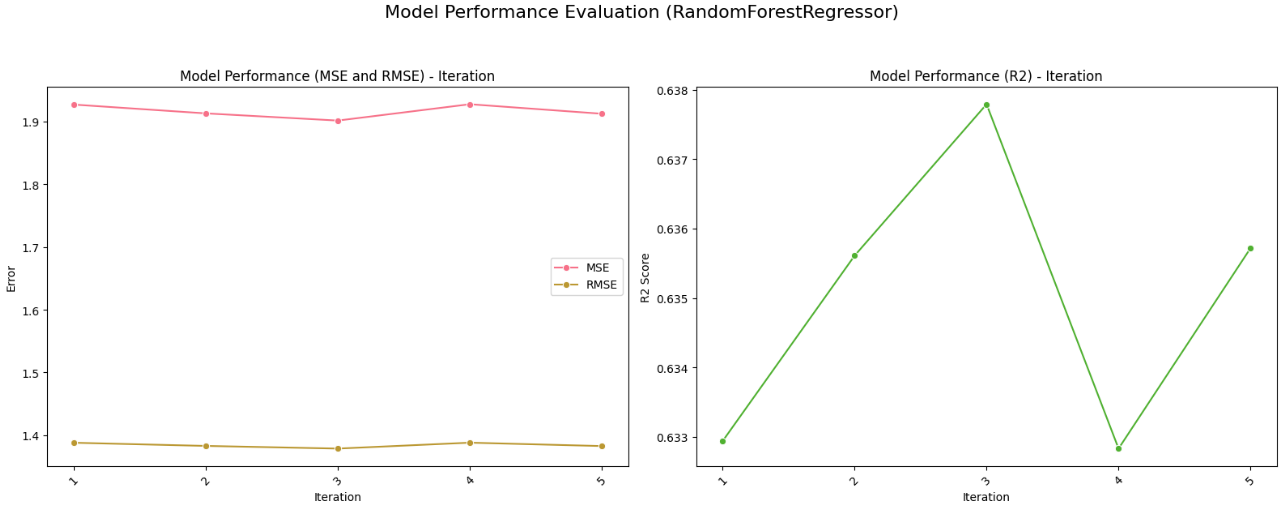


Figure 13: Average RF Performance by Iteration