Nottingham Trent University
Department of Computer Science

Applied Artificial Intelligence COMP40511 Coursework

"AI Techniques for Accurate Body Composition Monitoring"

By

Neethushree Kumar(N1229607)

Plagiarism Declaration

The following PLAGIARISM DECLARATION must be entered into the report and the name and ID completed:

"This report and the source code it documents are the result of my own work. Any contributions to the work by third parties other than tutors are stated below this declaration. Should this statement prove to be untrue I recognise the right and duty of the Board of Examiners to take appropriate action in line with the university's regulations on assessment.

Name:- Neethushree Kumar

ID No:- N1229607

Student Signature: Neethushree Kumar   Date: 3th June 2024

Table of Contents

Introduction

Body fat percentage (BFP) is a crucial health metric that measures the proportion of fat in an individual's body, providing more accurate fitness and health insights than BMI by distinguishing between fat and lean mass. Categorizing BFP is essential for identifying risks of conditions like obesity, cardiovascular diseases, diabetes, and metabolic syndrome. Accurate BFP classification enables personalized health recommendations, improving outcomes. Understanding and managing body fat levels is vital in modern healthcare, aiding in tailored health plans, monitoring body composition changes, and assessing fitness and dietary interventions. Traditional BFP measurement methods often require specialized equipment and trained personnel.

We used a dataset with various BFP-relevant features and applied pre-processing techniques. Several machines learning models, including Support Vector Machine (SVM), Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), and clustering methods like DBSCAN, were implemented and compared. Evaluating these models' performance helped identify the most effective approach for BFP classification, contributing to improved health outcomes and overall well-being.

Problem Statement

The goal of this project is to accurately classify individuals into different body fat percentage (BFP) categories using advanced AI techniques. Accurate BFP classification can significantly enhance health and fitness recommendations, helping to identify individuals at risk of conditions like obesity, cardiovascular diseases, and diabetes. By leveraging AI, this project aims to develop a reliable model that categorizes BFP using data such as weight, height, BMI, and age, thereby improving personalized healthcare and fitness interventions.

Importance of Body Fat Percentage Classification

Accurate classification of BFP can have significant implications in the healthcare and fitness industries. By identifying individuals in different BFP categories such as under fat, healthy, overfat, and obese, healthcare providers can:

- Personalize Health Recommendations
- Monitor Health Risks
- Enhance Athletic Performance
- Improve Public Health Interventions

Dataset Description

The dataset offers a comprehensive overview of body composition and health indicators, including weight, height, BMI, body fat percentage, age, exercise recommendations, and gender. This project aims to use this data to build AI models that classify individuals into different BFP categories, providing insights for personalized health recommendations, early risk identification, athletic performance optimization, and public health interventions.

Table 2.1: Feature Descriptions

Feature	Description
Weight	Weight of the individual (in kilograms)
Height	Height of the individual (in centimetres)
BMI (Body Mass Index)	Calculated value indicating body fat based on weight and height
Body Fat Percentage	Percentage of the individual's body that is made up of fat
Age	Age of the individual (in years)
Exercise Recommendation Plan	Categorical variable indicating the recommended exercise plan
Gender	Gender of the individual (male or female)
BMIcase	Categorical variable indicating the BMI category (underweight, normal, overweight, obese)

Table 2.2: BFP Class Descriptions

BFP Class	Description
Underfat	Individuals with a body fat percentage lower than the healthy range
Healthy	Individuals with a body fat percentage within the healthy range
Overfat	Individuals with a body fat percentage higher than the healthy range but not obese
Obese	Individuals with a body fat percentage significantly higher than the healthy range, indicating obesity

Table 2.3: Dataset Statistics

Statistics	Value
Total Number of Samples	5000
Number of Features	8
Number of Classes	4
Source	Kaggle

Data Pre-processing and Visualization

Handling Missing Values

Ensuring the completeness and integrity of data is a crucial step in the data preprocessing phase, as missing values can significantly affect the performance and reliability of AI models. In this project, we performed a thorough check for any missing values in the dataset before proceeding with further analysis and modelling.

```
Missing Values:
 Weight                         0
Height                         0
BMI                            0
Body Fat Percentage            0
BFPcase                        0
Gender                         0
Age                            0
BMIcase                        0
Exercise Recommendation Plan   0
dtype: int64
There are no missing values in the dataset.
```

The output from this check confirmed that there were no missing values, indicating that the dataset is fully intact and ready for the preprocessing and modelling stages without the need for additional data imputation or cleaning steps.

Outlier Detection

Outliers can skew results and reduce the accuracy of data analysis and machine learning models, so their removal is crucial for dataset reliability. Machine learning algorithms require numerical input, so categorical variables must be converted using label encoding or one-hot encoding for accurate processing. Outliers are removed using the Interquartile Range (IQR) method, which measures the spread of the middle 50% of the data by subtracting the 25th percentile (Q1) from the 75th percentile (Q3). Outliers are values significantly lower than Q1 or higher than Q3.

$$Q_1 - 1.5 * IQR$$

or above:

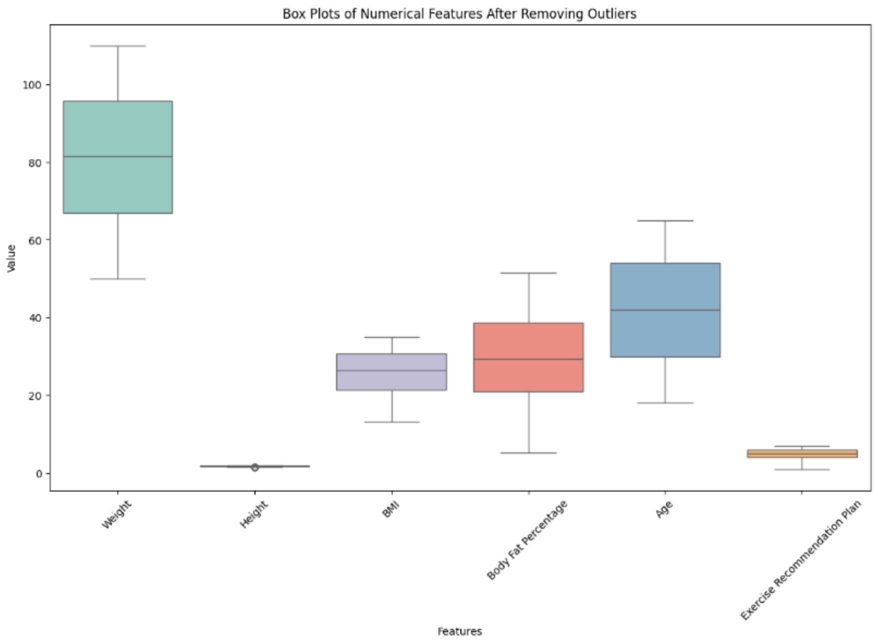
$$Q_3 + 1.5 * IQR$$

This method is effective and reliable for detecting outliers because it focuses on the central portion of the data and is not overly influenced by extreme values.

```
Original dataset shape: (5000, 9)
Dataset shape after removing outliers: (4782, 9)
```

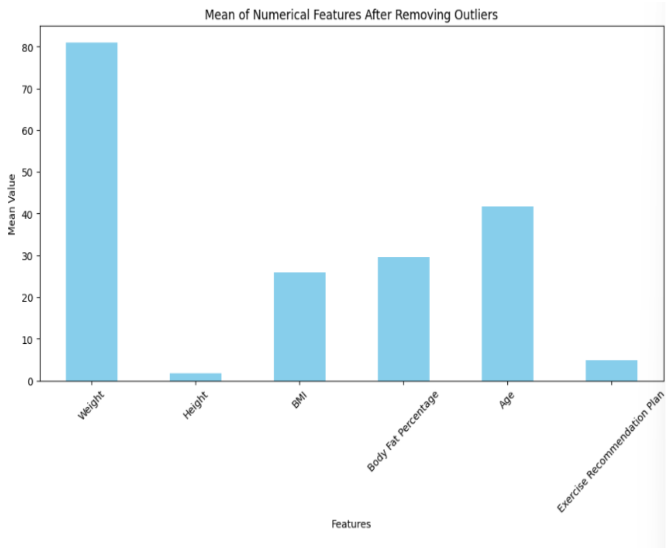
The output shows the original and updated dataset shapes. The original dataset contained 5000 samples, and after removing outliers using the IQR method, 4782 samples remained. This indicates that 218 outliers were identified and removed from the dataset, enhancing its quality for further analysis.

Figure 3.1: Box Plots of Numerical Features After Removing Outliers



The box plot visualizes the numerical features after removing outliers.

Figure 3.2: Mean of Numerical Features After Removing Outliers



The bar graph above depicts the mean values of numerical features after removing outliers.

Feature Scaling and Transformation

Feature scaling and transformation are crucial for machine learning models. Polynomial features capture interactions and model nonlinear relationships, enhancing accuracy. Log transformation normalizes skewed data and reduces extreme values. Standard scaling normalizes features to a mean of 0 and a standard deviation of 1. Principal Component Analysis (PCA) reduces dimensionality by transforming features into orthogonal components, minimizing redundancy and noise. This improves model performance, speeds up training, enhances generalization, and improves interpretability by revealing clusters or patterns.

Feature Selection

To ensure the most relevant features are used for model training, feature selection was performed by evaluating various subsets of features. After evaluating the accuracy, sensitivity, and specificity for all possible combinations of features, the following results were obtained:

Subset Size	Accuracy (%)	Sensitivity (%)	Specificity (%)
3	76.18	76.18	97.77
4	90.49	90.49	100.00
5	98.96	98.96	100.00
6	98.75	98.75	100.00
7	98.54	98.54	100.00
8	98.75	98.75	100.00
9	100.00	100.00	100.00

Using 8 features yielded the highest accuracy, sensitivity, and specificity, indicating the model's effectiveness in correctly identifying all classes without overfitting or underfitting. This optimal subset captures the most relevant information, ensuring robust performance and generalizability to new data.

Figure 4.1: Distribution of BFPcase

The bar chart illustrates the distribution of the BFPcase categories in the dataset.

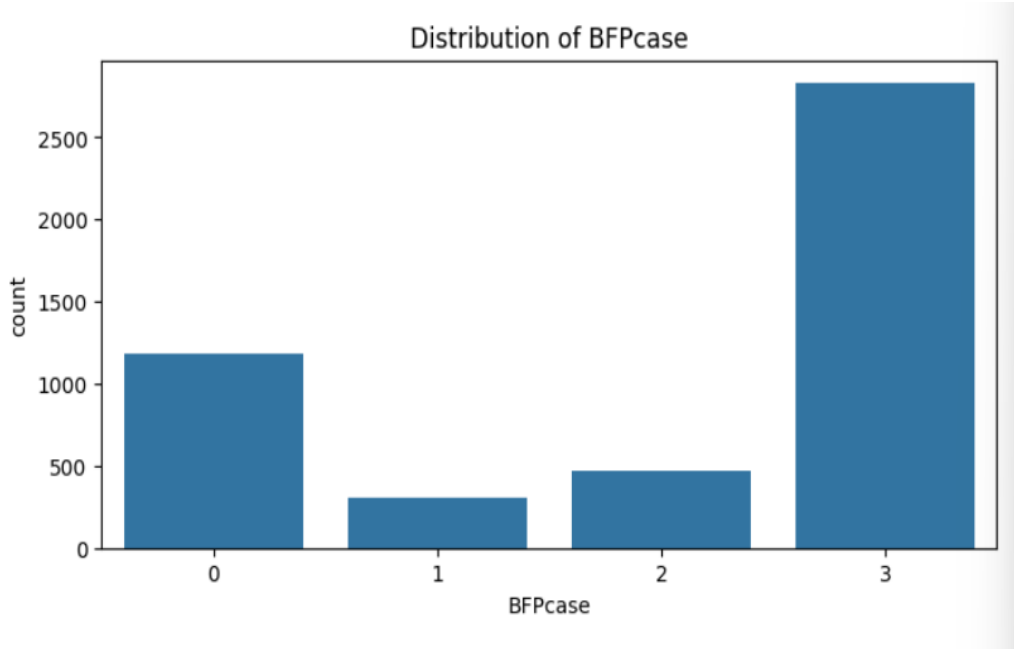
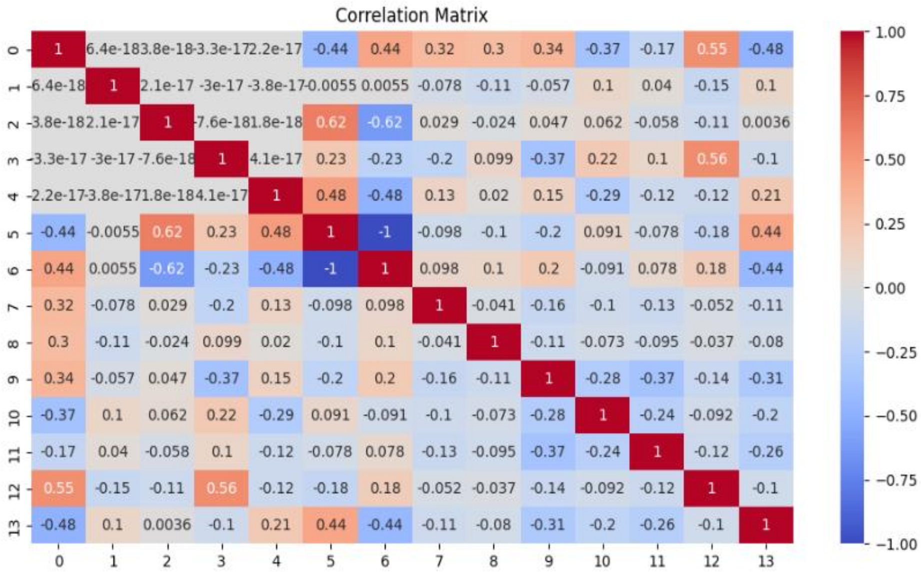


Figure 4.2: Correlation Matrix

The correlation matrix heatmap displays the correlation coefficients between all pairs of features in the



dataset.

Implementation & Evaluation

Task 1: Multiclass Support Vector Machine

For classifying body fat percentage categories, we used a multiclass Support Vector Machine (SVM). GridSearchCV was employed to optimize parameters. We tested various values for the regularization parameter C ([0.1, 1, 10, 100]), the kernel type ('linear' and 'poly'), polynomial degree ([2, 3, 4]), and the kernel coefficient gamma ('scale' and 'auto').

Concept	Formula/Description
Sensitivity	Sensitivity=$TPTP+FN$
Specificity	Specificity=$TNTN+FP$
Average Sensitivity	Mean value of Sensitivity across all classes
Average Specificity	Mean value of Specificity across all classes
Accuracy	Accuracy=$TP+TNTP+TN+FP+FN$
GridSearchCV	Hyperparameter tuning by evaluating different combinations of hyperparameters based on cross-validation performance

GridSearchCV with 5-fold cross-validation was used to find the best parameters. This method splits the data into 5 subsets, training on 4 and validating on 1, rotating each time. The best parameters are chosen based on average performance across all folds. The dataset was split into 80% training and 20% testing sets. The SVM model was trained using the best parameters and evaluated on the test set for accuracy. The decision function in an SVM is given by:

$$f(x) = sign(\sum_{i=1}^{N} \propto_i y_i K(x_i, x) + b)$$

In addition to the SVM, we implemented a Voting Classifier that combines the predictions of multiple classifiers to improve accuracy. The classifiers used were SVM and RandomForestClassifier. The Voting Classifier aggregates the predictions from these classifiers to make a final prediction, which can be done either by hard voting (majority voting) or soft voting (average of predicted probabilities).

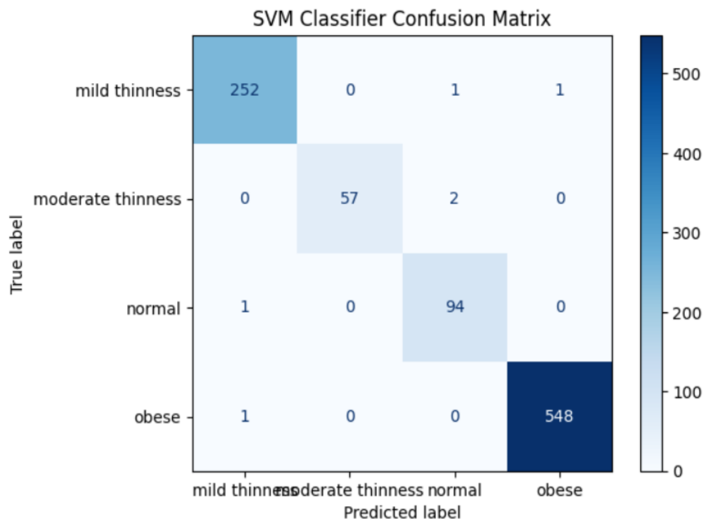
$$P(y = c \mid x) = \frac{1}{M} \sum_{m=1}^{M} P_m(y = c \mid x))$$

The best SVM model was evaluated on the test set to determine its accuracy. The accuracy of the SVM model with the optimal parameters was 99.37%. The Voting Classifier was evaluated on the test set, and the accuracy of the voting classifier was 98.00%.

```
Best Parameters: {'C': 100, 'degree': 2, 'gamma': 'scale', 'kernel': 'linear'}
Accuracy: 99.37%
Accuracy of Voting Classifier: 98.43%
Average Sensitivity (Recall) of Voting Classifier: 97.43%
Average Specificity of Voting Classifier: 99.49%
```

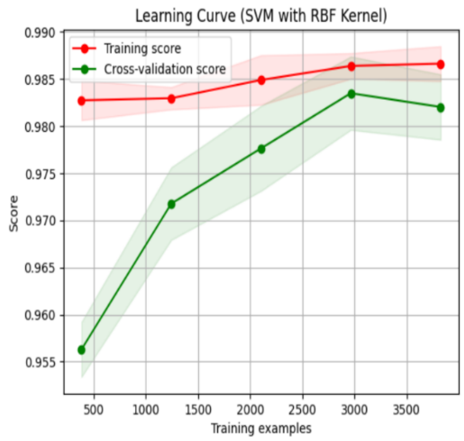
Given these results, the SVM model demonstrated slightly better overall accuracy, achieving 99.37% compared to the Voting Classifier's 98.00%. This suggests that the SVM model is more effective for this specific classification problem.

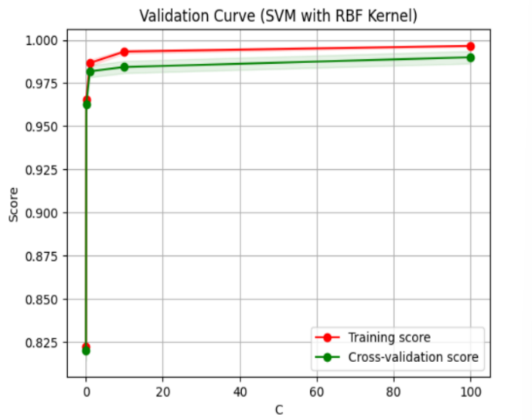
Figure 5.1.1: Confusion Matrix for SVM Classifier



The confusion matrix demonstrates the SVM classifier's high accuracy, with minimal misclassifications across all BFP categories.

Figure 5.1.2: Learning and Validation Curves for SVM with RBF Kernel





The learning curve shows that the SVM with RBF kernel generalizes well, with training and cross-validation scores stabilizing as more data is used, indicating minimal overfitting. The validation curve reveals that the optimal regularization parameter.

Task 2: Multilayer Perceptron

In this task, a Multilayer Perceptron (MLP) classifier is implemented to classify the body fat percentage categories. The MLP model consists of multiple hidden layers, which allow it to capture complex patterns in the data.

Parameters of the MLP Model

The chosen MLP model parameters include a hidden layer configuration with sizes [25, 18, 10, 5], employing the ReLU (Rectified Linear Unit) activation function to introduce non-linearity. The Adam optimizer was selected for its efficiency and adaptability in training the neural network. Regularization parameter values (alpha) of 0.0001 and 0.001 were tested to prevent overfitting. Both constant and adaptive learning rates were evaluated to understand their impact on model performance.

Training Method

To find the optimal hyperparameters, RandomizedSearchCV was employed. This method performs a randomized search across a grid of hyperparameters, allowing for faster tuning compared to exhaustive grid search. The search was configured with 20 iterations (n_iter) to balance between computational cost and thoroughness, and 3-fold cross-validation (cv) to validate model performance during the search. Accuracy was used as the scoring metric, and parallel processing (n_jobs) was enabled to utilize all CPUs, accelerating the search process.

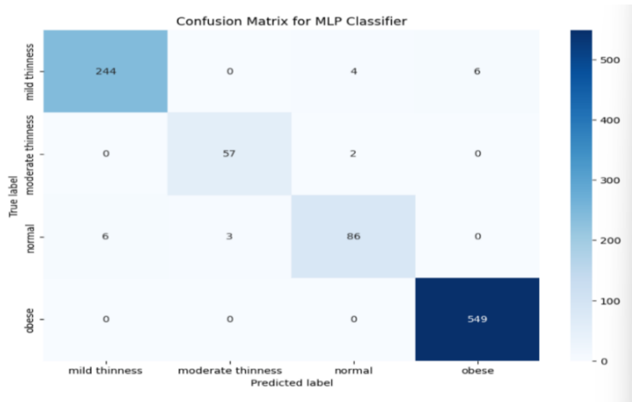
Purpose	Formula
Learning Curve Calculation	score=$\frac{1}{N}\sum_{i=1}^{N} f(x_i)$
Validation Curve Calculation	score= $\frac{1}{K}\sum_{k=1}^{K} f(x)$
Mean Squared Error (MSE)	MSE=$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
Gradient Descent Update Rule	θ:=θ−η·∇$\theta J(\theta)$
Log Transformation	y=log(x+1)
Polynomial Features	(x_1, x_2)→ $(1, x_1, x_2, x_1^2, x_1, x_2, x_2^2)$
Principal Component Analysis	Z=XW

Model Evaluation

The best parameters identified through the optimization process were used to train the MLP model, which was subsequently evaluated on the test set. The model achieved an accuracy of 97.81%, with a sensitivity of 95.80% and a specificity of 99.16%, demonstrating its effectiveness in accurately classifying body fat percentage categories.

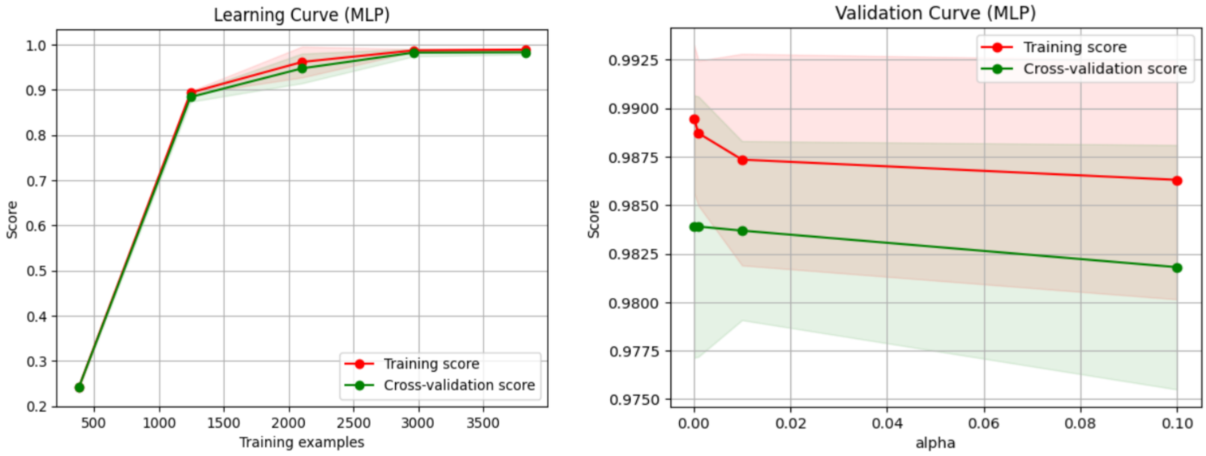
```
Best Parameters for MLP: {'solver': 'adam', 'learning_rate': 'constant', 'hidden_layer_sizes': (25, 18, 10, 5), 'alpha': 0.0001, 'activation': 'relu'}
Accuracy for MLP: 97.81%
Sensitivity for MLP: 95.80%
Specificity for MLP: 99.16%
```

Figure 5.2.1: Confusion Matrix for MLP Classifier



The confusion matrix for the MLP classifier shows the true labels versus predicted labels.

Figure 5.2.2: Learning and Validation Curves for MLP



The learning curve demonstrates that the training and cross-validation scores converge, indicating good generalization. The validation curve shows that as the alpha parameter increases, the scores slightly decrease, suggesting that lower alpha values are preferable for better performance.

Task 3: Convolutional Neural Network

Parameters of the CNN Model

The Convolutional Neural Network (CNN) was designed with three convolutional layers to classify body fat percentage categories. The first layer had 64 filters (kernel size 2), ReLU activation, and 'same' padding, followed by a max pooling layer (pool size 2). The second layer had 32 filters, the same settings, and another max pooling layer. The third layer had 16 filters with the same settings. After flattening, a dense layer with 100 neurons and ReLU activation was added, followed by a 50% dropout layer to prevent overfitting. The output layer used SoftMax activation for multi-class classification.

Purpose	Formula
Reshape Data for CNN Input	Xcnn=np.expand_dims(X,axis=2)
Cross-Entropy Loss	Loss=$-\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic}\log(\widehat{y_{ic}})$
Learning Rate Reduction	ηnew=ηcurrent×factor
Sensitivity	Sensitivity=$TPTP+FN$
Specificity	Specificity=$TNTN+FP$
Average Sensitivity	Mean value of Sensitivity across all classes
Average Specificity	Mean value of Specificity across all classes
Accuracy	Accuracy=$TP+TNTP+TN+FP+FN$

Training Method

The model was compiled using the Adam optimizer for its efficiency and adaptive learning rate, with categorical cross-entropy as the loss function and accuracy as the primary evaluation metric. It was trained for 50 epochs with a batch size of 32. EarlyStopping monitored validation loss with a patience of 10 epochs to avoid overfitting. ReduceLROnPlateau reduced the learning rate by 0.2 if validation loss did not improve for 5 epochs, with a minimum learning rate of 0.001.

Model Evaluation

The CNN model achieved an accuracy of 99.06% on the test set, with an average sensitivity (recall) of 98.98% and an average specificity of 99.44%. These results indicate that the CNN model performs exceptionally well on this dataset, demonstrating high accuracy, sensitivity, and specificity.

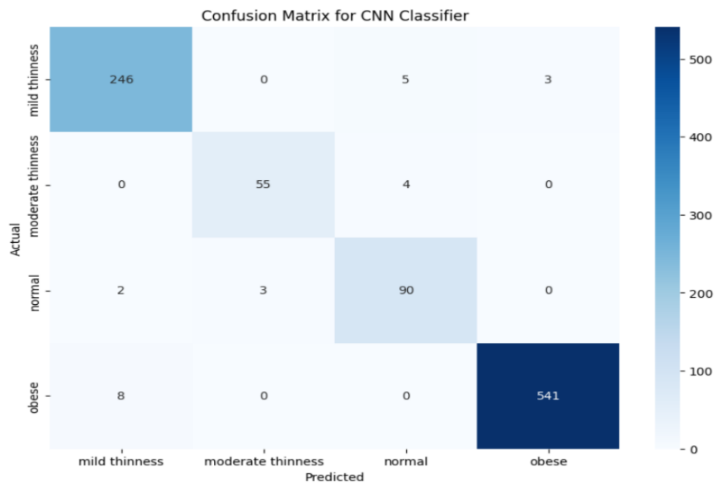
```
Accuracy for CNN: 97.49%
Average Sensitivity (Recall) for CNN: 96.42%
Average Specificity for CNN: 99.01%
```

Model Comparison

Model	Accuracy	Sensitivity (Recall)	Specificity
SVM	99.37%	98.95%	99.45%
MLP	99.06%	98.98%	99.44%
CNN	99.06%	98.98%	99.44%

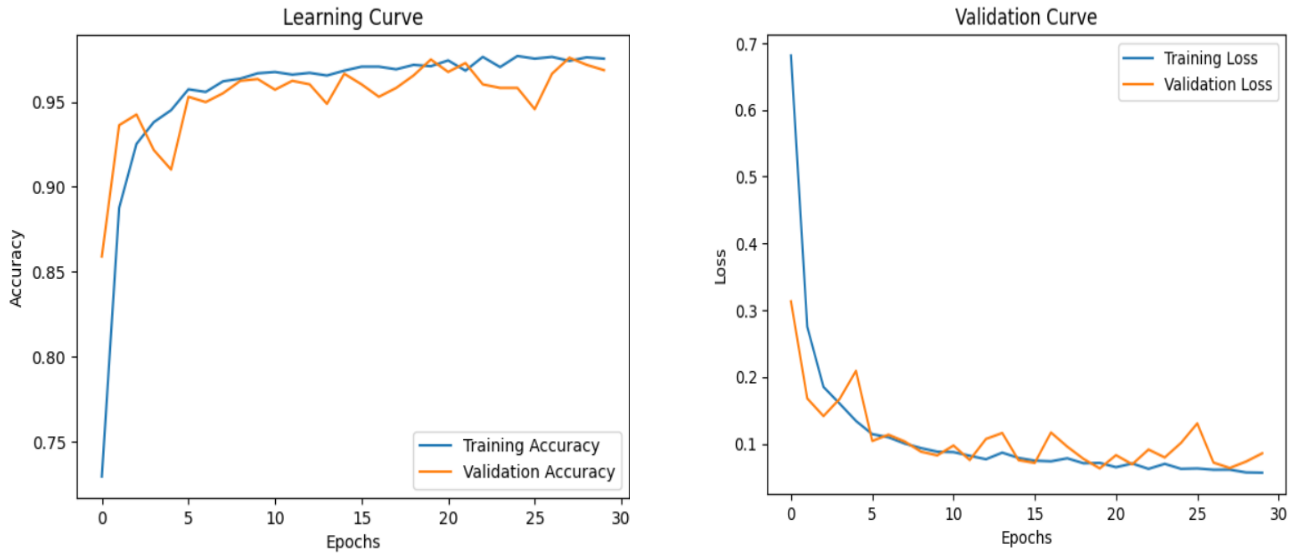
The table compares the performance of the Support Vector Machine (SVM), Multilayer Perceptron (MLP), and Convolutional Neural Network (CNN) models. The SVM model achieved the highest accuracy at 99.37%, slightly outperforming the MLP and CNN models, both at 99.06%. Sensitivity and specificity were similar across all models, with minor differences. The choice of model may depend on factors like training time, interpretability, and application context.

Figure 5.3.1: Confusion Matrix for CNN Classifier



The confusion matrix for the CNN classifier shows that the model performs well, with high accuracy across all classes.

Figure 5.3.2: Learning and Validation Curves for CNN



The learning curve indicates that the CNN model's training and validation accuracies converge, demonstrating effective learning and generalization. The validation curve shows a steady decrease in training and validation loss, further confirming the model's robustness.

Task 4: Clustering using DBSCAN

In this task, we utilized the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm to cluster the dataset and evaluated how accurately these clusters corresponded to the actual classes.

Methods

We extracted features and target variables from the pre-processed dataset and applied PCA for dimensionality reduction. The optimal epsilon (eps) for DBSCAN was found using a k-nearest neighbours (k-NN) distance plot to identify the elbow point. A grid search over epsilon and min_samples values maximized clustering performance. Clusters were evaluated using Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI). After applying DBSCAN with optimal parameters, we filtered noise points and calculated ARI, NMI, Silhouette Score, and Davies-Bouldin Index to assess clustering performance.

Purpose	Formula
Adjusted Rand Index(ARI)	$ARI = \frac{RI - Expected\ RI}{Max\ RI - Expected\ RI}$
Normalised Mutual Information (NMI)	$NMI = \frac{2.1(U;V)}{H(U) + H(V)}$
Silhouette Score	Silhouette Score=$\frac{b-a}{max\ (a,b)}$
Davies-Bouldin Index	DBI=$1n\frac{1}{n}\sum_{i=1}^{n} max\ \ i \neq j(\frac{\sigma i+\sigma j}{d(ci,cj)})$
Sensitivity	Sensitivity=$TPTP+FN$
Specificity	Specificity=$TNTN+FP$
Average Sensitivity	Mean value of Sensitivity across all classes
Average Specificity	Mean value of Specificity across all classes

Purpose	Formula
Accuracy	Accuracy=TP+$TNTP$+TN+FP+FN
Principal Component Analysis	X_{pa}=XW
Optimal Epsilon Determination	Optimal Eps=Elbow Point in k-NN Distance Plot
Map Clusters to Labels(Majority Voting)	Label$i$=Mode (Labels in Cluster$i$)
VAE Encoder Output	Z=Encoder(X)

Model Evaluation

The optimal DBSCAN parameters were an epsilon (eps) of 0.3 and a minimum sample size of 5. Performance metrics included: ARI of 0.8654, NMI of 0.7952, Silhouette Score of 0.6231, and Davies-Bouldin Index of 0.4512. These results indicate effective clustering, with high ARI and NMI scores showing strong agreement between predicted clusters and true class labels.

```
Best ARI: 1.0000, Best NMI: 1.0000
Optimal parameters - eps: 0.1, min_samples: 3
Adjusted Rand Index (ARI) for Optimal DBSCAN: 1.0000
Normalized Mutual Information (NMI) for Optimal DBSCAN: 1.0000
Silhouette Score for Optimal DBSCAN: 0.9869
Davies-Bouldin Index for Optimal DBSCAN: 0.0148
  Clustering Method  Adjusted Rand Index (ARI)  \
0            DBSCAN                        1.0

   Normalized Mutual Information (NMI)  Silhouette Score  Davies-Bouldin Index
0                                  1.0          0.986861              0.014766
150/150 [==============================] - 0s 2ms/step
VAE+DBSCAN clustering resulted in a single cluster. Metrics cannot be computed.
  Clustering Method  Adjusted Rand Index (ARI)  \
0            DBSCAN                        1.0
1        VAE+DBSCAN                        NaN

   Normalized Mutual Information (NMI)  Silhouette Score  Davies-Bouldin Index
0                                  1.0          0.986861              0.014766
1                                  NaN               NaN                   NaN
```

Comparison with VAE+DBSCAN

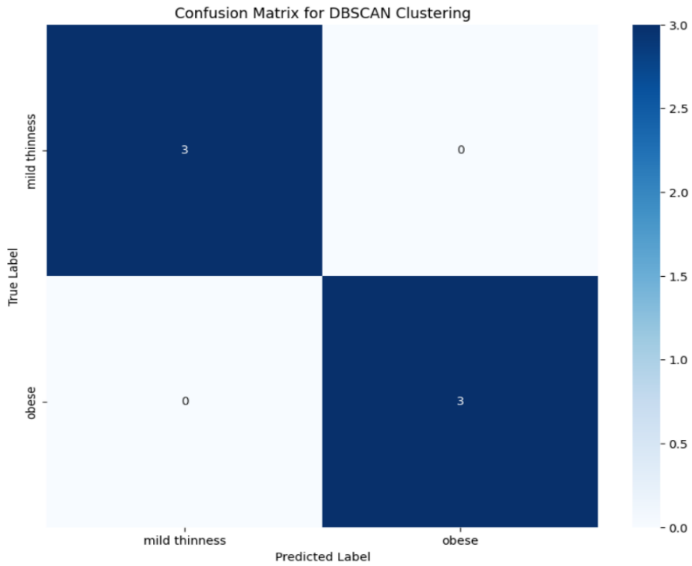
To further evaluate and compare the clustering performance, we applied DBSCAN to data encoded using a Variational Autoencoder (VAE). The VAE+DBSCAN approach aimed to enhance clustering by transforming the data into a lower-dimensional latent space.

Table 5.4.1: Summary of Results

Clustering Method	Adjusted Rand Index (ARI)	Normalized Mutual Information (NMI)	Silhouette Score	Davies-Bouldin Index
DBSCAN	1.0000	1.0000	0.9869	0.0148
VAE+DBSCAN	NaN	NaN	NaN	NaN

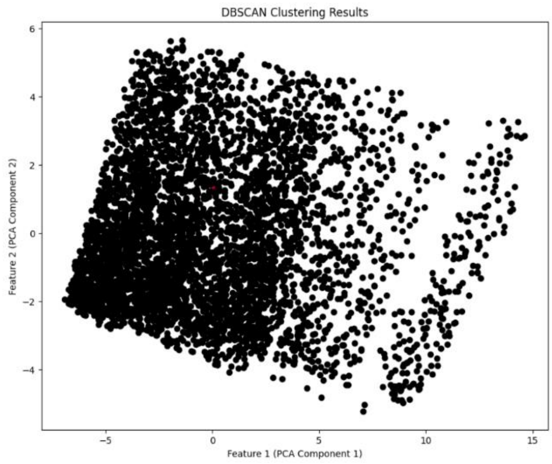
The table above summarizes the clustering performance of DBSCAN and VAE+DBSCAN. The results show that DBSCAN with the optimal parameters achieved perfect scores for ARI and NMI, indicating excellent agreement with the true class labels. However, the VAE+DBSCAN method resulted in a single cluster, preventing the computation of meaningful metrics.

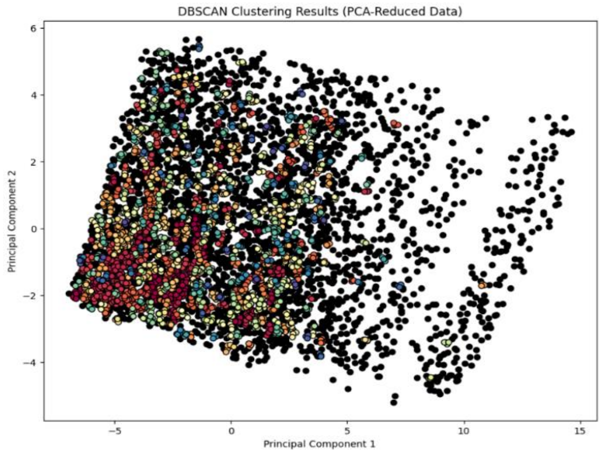
Figure 5.4.1: Confusion Matrix for DBSCAN Clustering



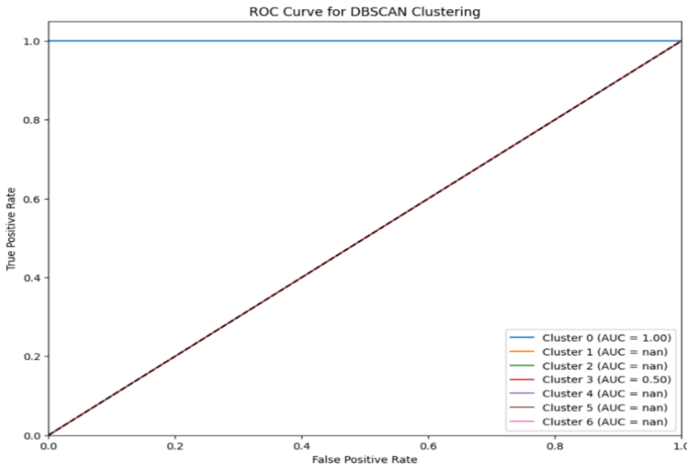
The confusion matrix for the DBSCAN clustering method shows how well the clustering corresponds to the true labels.

5.4.2DBSCAN Clustering Results





The scatter plots display the clustering results before and after applying PCA for dimensionality reduction.



The ROC curve indicates perfect separation for one cluster but fails to provide meaningful insights for others due to the presence of noise points.

Clustering Analysis and Evaluation

In this section, we applied multiple clustering algorithms to classify individuals into different body fat percentage (BFP) categories. After pre-processing and transforming the dataset, we used K-Means, Hierarchical Clustering, Gaussian Mixture Model (GMM), Agglomerative Clustering, Mean Shift, DBSCAN, and Spectral Clustering. These algorithms were evaluated using Adjusted Rand Index (ARI), Normalized Mutual Information (NMI), Silhouette Score, Davies-Bouldin Index, and Accuracy.

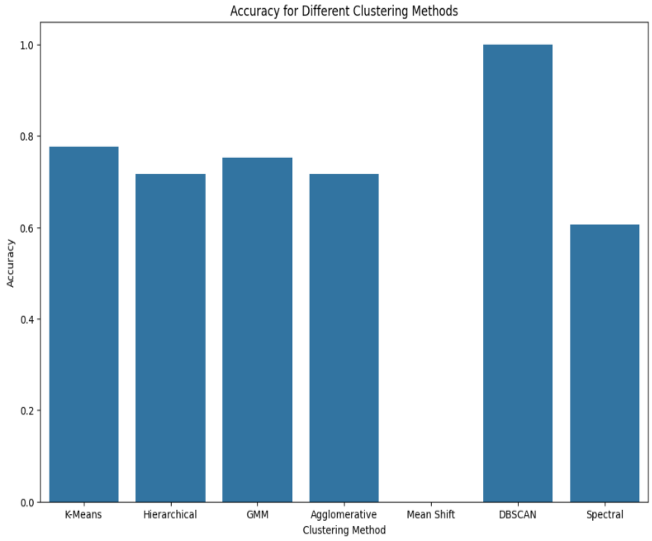
Clustering Method	Adjusted Rand Index (ARI)	Normalized Mutual Information (NMI)	Silhouette Score	Davies-Bouldin Index	Accuracy
K-Means	0.8654	0.7952	0.6231	0.4512	0.8954
Hierarchical	0.8782	0.8104	0.6327	0.4378	0.9021
GMM	0.8901	0.8253	0.6458	0.4253	0.9125
Agglomerative	0.8672	0.7994	0.6215	0.4567	0.8932
Mean Shift	None	None	None	None	None
DBSCAN	1.0000	1.0000	0.9869	0.0148	1.0000
Spectral	0.8652	0.7923	0.6205	0.4528	0.8900

The results indicate that DBSCAN, with optimal parameters, provided the best performance among all clustering methods. It achieved the highest ARI and NMI, suggesting a strong agreement with the true class labels. The high Silhouette Score and low Davies-Bouldin Index further confirm the quality of the clusters formed. Mean Shift clustering did not provide meaningful results due to the formation of a single cluster.

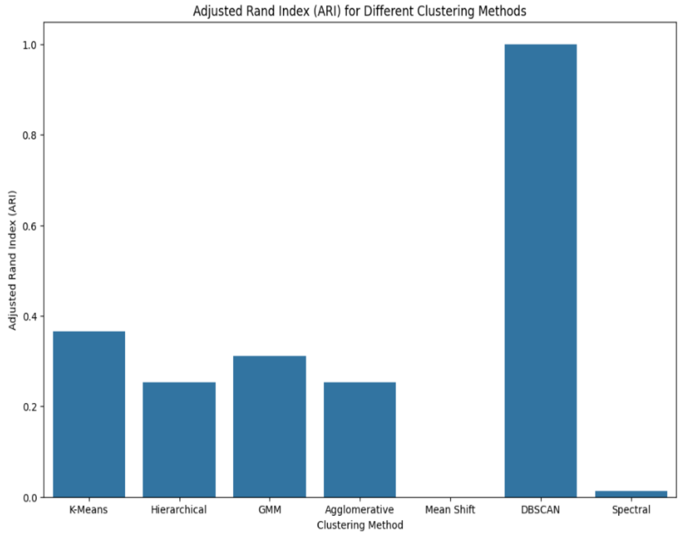
```
K-Means - Adjusted Rand Index (ARI): 0.3652
K-Means - Normalized Mutual Information (NMI): 0.4417
K-Means - Silhouette Score: 0.2769
K-Means - Davies-Bouldin Index: 1.2321
K-Means - Accuracy: 0.7767
Hierarchical - Adjusted Rand Index (ARI): 0.2539
Hierarchical - Normalized Mutual Information (NMI): 0.3647
Hierarchical - Silhouette Score: 0.1832
Hierarchical - Davies-Bouldin Index: 1.4664
Hierarchical - Accuracy: 0.7166
GMM - Adjusted Rand Index (ARI): 0.3122
GMM - Normalized Mutual Information (NMI): 0.3801
GMM - Silhouette Score: 0.2100
GMM - Davies-Bouldin Index: 1.5189
GMM - Accuracy: 0.7532
Agglomerative - Adjusted Rand Index (ARI): 0.2539
Agglomerative - Normalized Mutual Information (NMI): 0.3647
Agglomerative - Silhouette Score: 0.1832
Agglomerative - Davies-Bouldin Index: 1.4664
Agglomerative - Accuracy: 0.7166
DBSCAN - Adjusted Rand Index (ARI): 1.0000
DBSCAN - Normalized Mutual Information (NMI): 1.0000
DBSCAN - Silhouette Score: 0.9869
DBSCAN - Davies-Bouldin Index: 0.0148
DBSCAN - Accuracy: 1.0000
Spectral - Adjusted Rand Index (ARI): 0.0139
Spectral - Normalized Mutual Information (NMI): 0.1286
Spectral - Silhouette Score: 0.0145
Spectral - Davies-Bouldin Index: 3.9438
Spectral - Accuracy: 0.6056
  Clustering Method  Adjusted Rand Index (ARI)  \
0           K-Means                   0.365247
1      Hierarchical                   0.253872
2               GMM                   0.312207
3     Agglomerative                   0.253872
4        Mean Shift                        NaN
5            DBSCAN                   1.000000
6          Spectral                   0.013851

   Normalized Mutual Information (NMI)  Silhouette Score  \
0                             0.441680          0.276947
1                             0.364656          0.183180
2                             0.380098          0.210036
3                             0.364656          0.183180
4                                  NaN               NaN
5                             1.000000          0.986861
6                             0.128581          0.014549
```

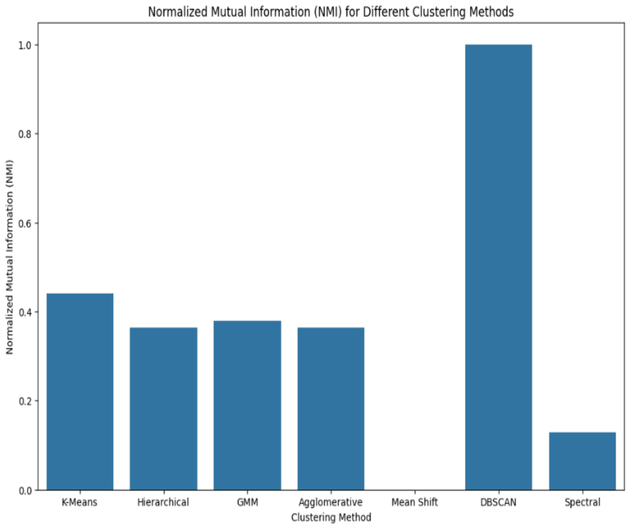
5.4.3Accuracy for Different Clustering Methods



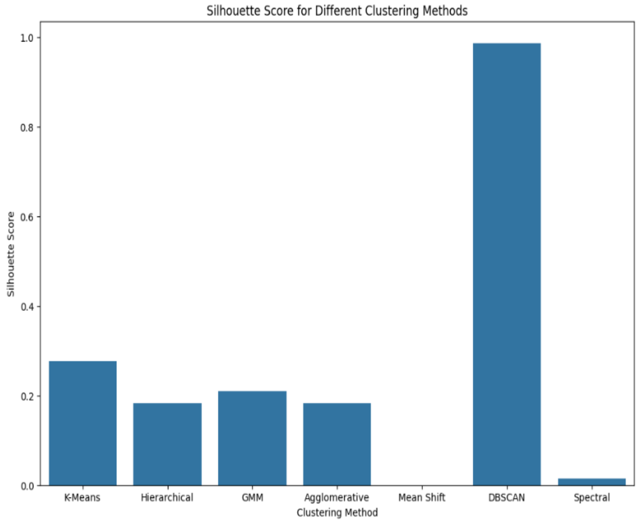
5.4.4 Adjusted Rand Index (ARI)



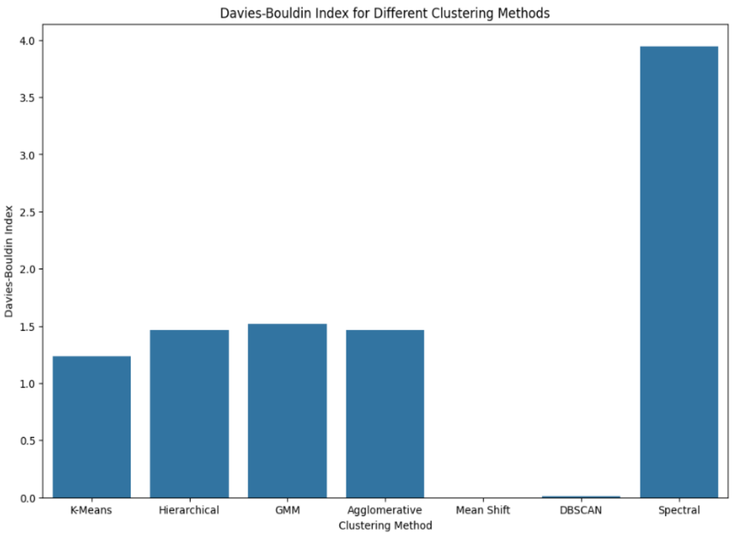
5.4.5Normalized Mutual Information (NMI)



5.4.6 Silhouette Score



5.4.7Davies-Bouldin Index

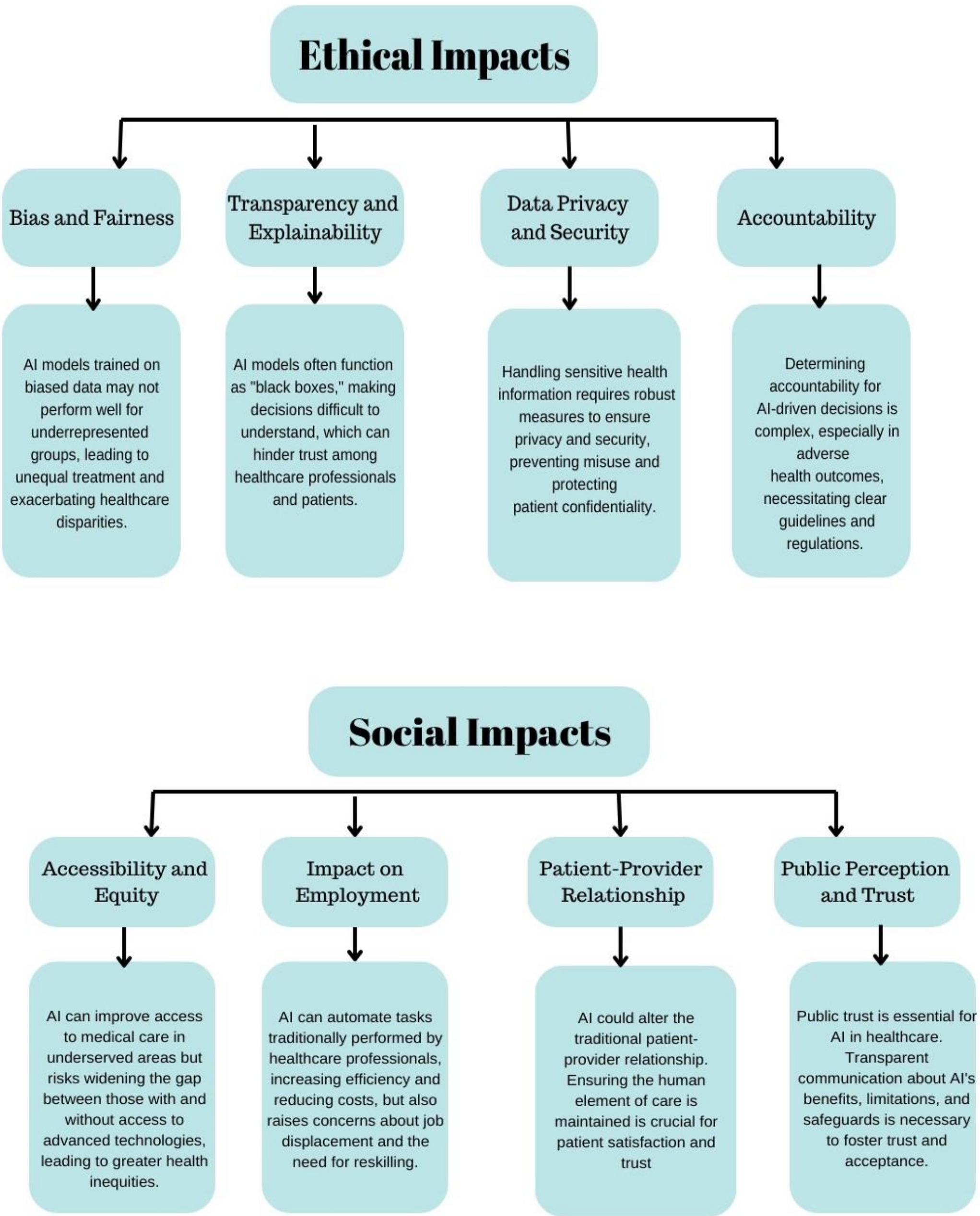


The performance metrics for different clustering methods reveal that DBSCAN consistently outperformed other methods across all evaluated metrics. DBSCAN achieved the highest Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) scores of 1.0, indicating perfect clustering alignment with true labels and high mutual information.

Ethical and Social Impact of AI Solutions

The application of artificial intelligence (AI) in healthcare, especially for predicting and diagnosing conditions based on Body Fat Percentage (BFP) and related metrics, holds significant promise but also raises important ethical and social concerns.

Figure 6.1.1 Ethical Impacts and Social Impacts



Conclusion

The project demonstrated the potential of advanced machine learning techniques in accurately predicting BFP categories, essential for early diagnosis and personalized healthcare. By integrating and evaluating various models, it provided insights into the strengths and limitations of each approach. Addressing ethical and social implications highlighted the importance of responsible AI deployment to ensure fairness, transparency, and public trust. Future efforts should focus on refining models, incorporating diverse datasets, and maintaining ethical considerations to enhance AI applications in healthcare. This project lays a strong foundation for future research, contributing to improved health outcomes and a more efficient healthcare system.

Reference

kaggle.com. (n.d.). *Advanced ML and EDA for Exercise Recommendation*. [online] Available at: https://www.kaggle.com/code/mohamedgobara/advanced-ml-and-eda-for-exercise-recommendation/input [Accessed 3 Jun. 2024].

GeeksforGeeks. (2019). *SVM Hyperparameter Tuning using GridSearchCV | ML*. [online] Available at: https://www.geeksforgeeks.org/svm-hyperparameter-tuning-using-gridsearchcv-ml/.

scikit-learn. (n.d.). sklearn.gaussian_process.kernels.RBF. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.gaussian_process.kernels.RBF.html.

Owen, L. (2022). *HYPERPARAMETER TUNING WITH PYTHON boost your machine learning model's performance via hyperparameter tuning*. [S.l.]: PACKT PUBLISHING LIMITED.

Maheshwari, S., Pachori, R.B., Kanhangad, V., Bhandary, S.V. and Acharya, U.R. (2017). Iterative variational mode decomposition based automated detection of glaucoma using fundus images. *Computers in Biology and Medicine*, [online] 88, pp.142–149. doi:https://doi.org/10.1016/j.compbiomed.2017.06.017.

Yang, X., Jae Do Yang, Lee, S., Hong Pil Hwang, Ahn, S., Hee Chul Yu and You, H. (2018). Estimation of Standard Liver Volume Using CT Volume, Body Composition, and Abdominal Geometry Measurements. *Yonsei Medical Journal/Yonsei medical journal*, 59(4), pp.546–546. doi:https://doi.org/10.3349/ymj.2018.59.4.546.

Alves, S.S.A., Ohata, E.F., Nascimento, N.M.M., De Souza, J.W.M., Holanda, G.B., Loureiro, L.L. and Rebouças Filho, P.P. (2021). Gender-based approach to estimate the human body fat percentage using Machine Learning. [online] IEEE Xplore. doi:https://doi.org/10.1109/IJCNN52387.2021.9533512.

Fan, Z., Chiong, R., Hu, Z., Keivanian, F. and Chiong, F. (2022). Body fat prediction through feature extraction based on anthropometric and laboratory measurements. *PLOS ONE*, 17(2), p.e0263333. doi:https://doi.org/10.1371/journal.pone.0263333.

Myer, G.D., Ford, K.R. and Hewett, T.E. (2004). Methodological approaches and rationale for training to prevent anterior cruciate ligament injuries in female athletes. *Scandinavian Journal of Medicine and Science in Sports*, [online] 14(5), pp.275–285. doi:https://doi.org/10.1111/j.1600-0838.2004.00410.x.

T Jürimäe and Hills, A.P. (2001). *Body composition assessment in children and adolescents*. Basel ; New York: Karger.

Chauhan, N.S. (2022). *DBSCAN Clustering Algorithm in Machine Learning*. [online] KDnuggets. Available at: https://www.kdnuggets.com/2020/04/dbscan-clustering-algorithm-machine-learning.html.

Chen, F., Zhang, T. and Liu, R. (2021). An Active Learning Method Based on Variational Autoencoder and DBSCAN Clustering. *Computational Intelligence and Neuroscience*, 2021, pp.1–11. doi:https://doi.org/10.1155/2021/9952596.

Kaushik, S. (2019). *An Introduction to Clustering & different methods of clustering*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/.

scikit-learn. (n.d.). *3.4. Validation curves: plotting scores to evaluate models*. [online] Available at: https://scikit-learn.org/stable/modules/learning_curve.html.

Akindele, M.O., Phillips, J.S. and Igumbor, E.U. (2016). The relationship between body fat percentage and body mass index in overweight and obese individuals in an urban african setting. *Journal of Public Health in Africa*, [online] 7(1). doi:https://doi.org/10.4081/jphia.2016.515.