**Physical Performance Classification Report**

1. **Abstract**

This report outlines an analysis of a dataset aimed at predicting individual performance on an obstacle course. Using multi-class classification algorithms, their effectiveness is explored in predicting performance classes based on physical attributes and other characteristics. Key results include model performance comparisons, feature importance rankings, and predictions for an unlabelled dataset. The relationship between different features for the individuals is also explored. The findings highlight the most suitable model and display their predictions on an unlabelled dataset.

**Key Findings:**

* The Random Forest Classifier was the most effective out of the different models used for prediction.
* The accuracy of the Random Forest classifier is 0.71, the next best being the Multilayer Perceptron with the same accuracy score of 0.71.
* The different classifiers can predict individuals in class ‘D’ to a significantly higher accuracy than other classes.
* The F1-score average for all models for the classes ‘A’, ‘B’, ‘C’, and ‘D’ is 0.524, 0.590, and 0.616, and 0.818, respectively.
* The ‘sit and bend forward’ feature was seen to have the most importance for classification
* PCA and t-SNE visualizations showed no clusters within the data, indicating no pattern within the data.

1. **Introduction:**

**Overview of the Data**

The overall dataset contains two types of data, a labelled and unlabelled set. The labelled set contains data about physical attributes from 10,000 individuals, which the models are trained on. The unlabelled set contains data about the physical attributes from 20 individuals, which will be input into the model for prediction.

The features within the dataset included:

* Age (20-64 years)
* Gender (M, F)
* Physical characteristics: height (cm), weight (kg), body fat percentage.
* Physiological metrics: diastolic and systolic blood pressure.
* Performance metrics: grip force, sit-and-bend-forward (cm), sit-up counts, broad jump (cm).

The classes were labelled A (elite) to D (below average) based on the individual’s performance on the obstacle course. These classes are determined by the course completion time taken by the individuals. This ranges from <3 minutes (Class A), to >6 minutes (Class D).

**Aims and Objectives**

**Primary Objective**:

* Predict the performance class (A to D) of individuals accurately based on the provided features.

**Secondary Objectives**:

* + Compare the performance of various classification algorithms.
  + Identify the most influential features for prediction.
  + Apply the selected model to classify unlabelled data.
  + Perform additional analyses to gain insight into the data.

**Analytical Summary of the Data**

Out of the 10,000 individuals that data was collected on, Figures 1 and 2 show the number and percentage of individuals belonging to each class.

A graph of different colored bars

Description automatically generated

*Figure 1. Number of Individuals in Each Class*

A graph of a number of individuals

Description automatically generated

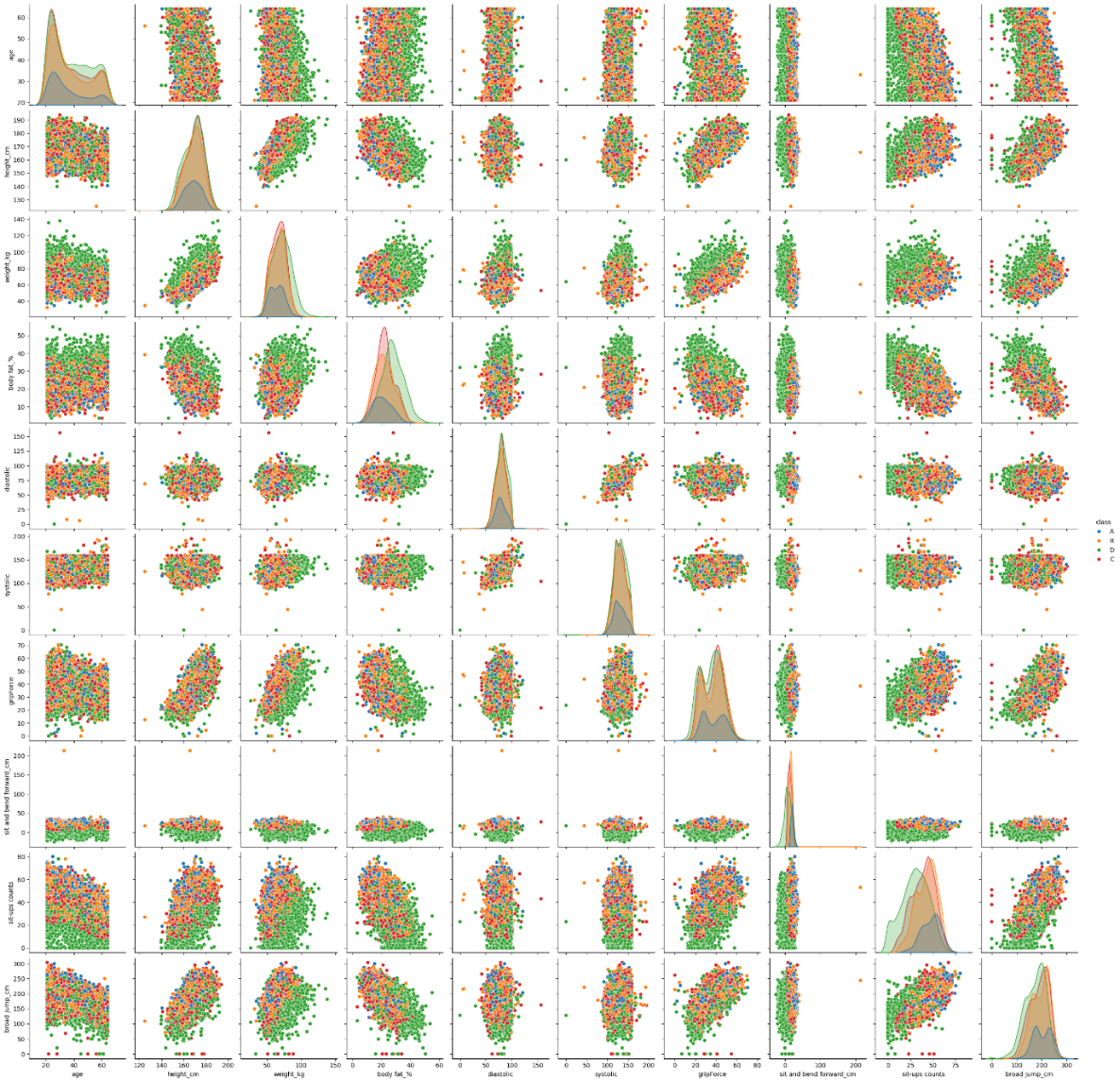
*Figure 2. Percentage of Individuals in Each Class*

Figures 1 and 2 show that the classes C and D make up majority of the data, around 63% of the data.

Additionally, Figure 3 shows the correlation between the features present in the data and classes assigned to individuals.

As seen in Figure 3, a clear boundary can be seen between the D class (in green), and the rest of the classes. This boundary is seen most clearly in the ‘sit and bend forward\_cm’ feature (8th column). This could suggest how important the feature is for determining class.

*(Figure 3 on the next page)*



*Figure 3. Pair plot of the different features in the dataset.*

1. **Methods:**

**Classifiers:**

While developing the model, 5 different classifiers were considered:

1. **Logistic Regression**:

This classifier explores the relationship between features and the target class through a linear decision boundary. It is suitable for both binary and multiclass classification. Important parameters include the regularization penalties.

1. **Random Forest:**

This classifier is an ensemble learning method based on decision trees. It is able to handle non-linear relationships and can rank feature importance. Important parameters include number of estimators, the maximum number of features, and the maximum tree depth.

1. **Gradient Boosting**:

This classifier is another ensemble method that combines multiple decision trees to create a more powerful model. It works by building trees in a serial manner, where each tree tries to correct the mistakes of the previous one. Important parameters are similar to the Random Forest model, with the addition of the learning rate parameter.

1. **Support Vector Machines**:

This classifier uses various kernel functions (e.g. linear, RBF) to handle complex relationships. It is effective for multi-dimensional data. Important parameters include the kernel function and degree of the polynomial kernel.

1. **Multilayer Perceptron (MLP)**:

This classifier is a neural network-based model with multiple hidden layers. It can capture complex, non-linear patterns in the data. Important parameters include the number of hidden layers, number of nodes in each layer, and the learning rate of the model.

**Model Optimization:**

Model optimization was performed via cross-validation and a Grid search of the hyperparameters. A module from the scikit.learn package called GridSearchCV was used. This module allows cross validation and hyperparameter optimization within one set of code. Parameters were optimized for all models used. For each model, the following parameters were optimized:

* **Logistic Regression:** C (regularization function)
* **Random Forest:** n\_estimators, max\_depth, min\_samples\_split, min\_samples\_leaf, max\_features
* **Gradient Boosting**: learning\_rate, n\_estimators, max\_depth, min\_samples\_split, min\_samples\_leaf, max\_features
* **SVM:** C, kernel, gamma, degree
* **Multilayer Perceptron:** hidden\_layer\_sizes, alpha, learning \_rate\_init.

1. **Results**

**Choice of Classifier:**

The classifier chosen to take forward to use as a predictor was the Random Forest classifier. While overall, the test scores were all below 0.72, Random Forest provided the best test score of 0.71. The Multilayer Perceptron model also provided a score of 0.71, and training score of 0.86. Similarly, the test scores for Logistic regression, Support Vector machines, and Gradient boosting can be seen in Table 1 below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | logistic Regression | Random Forest | Gradient Boosting | SVM | multilayer perceptron |
| Train Score | 0.61 | 0.95 | 0.96 | 0.95 | 0.86 |
| test Score | 0.61 | 0.71 | 0.67 | 0.64 | 0.71 |

*Table 1. Train and Test scores of different models.*

Additionally, the f-scores seen with the Random Forest model are superior to the rest of the models. These scores can be summarized in the tables below. As seen in Tables 2-6, the score for the Random Forest model is significantly higher, with only the Multilayer Perceptron classifier providing similar scores.

|  |  |  |  |
| --- | --- | --- | --- |
| Logistic Regression | Precision | Recall | f1-score |
| A | 0.60 | 0.39 | 0.47 |
| B | 0.51 | 0.58 | 0.54 |
| C | 0.52 | 0.51 | 0.51 |
| D | 0.77 | 0.78 | 0.77 |
| Accuracy |  |  | 0.60 |

|  |  |  |  |
| --- | --- | --- | --- |
| Random Forest | Precision | Recall | f1-score |
| A | 0.58 | 0.51 | 0.54 |
| B | 0.59 | 0.71 | 0.64 |
| C | 0.71 | 0.66 | 0.68 |
| D | 0.89 | 0.82 | 0.85 |
| Accuracy |  |  | 0.71 |

|  |  |  |  |
| --- | --- | --- | --- |
| Gradient Boosting | Precision | Recall | f1-score |
| A | 0.49 | 0.49 | 0.48 |
| B | 0.58 | 0.61 | 0.59 |
| C | 0.65 | 0.63 | 0.64 |
| D | 0.83 | 0.82 | 0.82 |
| Accuracy |  |  | 0.67 |

|  |  |  |  |
| --- | --- | --- | --- |
| SVM | Precision | Recall | f1-score |
| A | 0.49 | 0.57 | 0.53 |
| B | 0.55 | 0.61 | 0.58 |
| C | 0.61 | 0.55 | 0.58 |
| D | 0.82 | 0.78 | 0.80 |
| Accuracy |  |  | 0.64 |

|  |  |  |  |
| --- | --- | --- | --- |
| Multilayer Percpeptron | Precision | Recall | f1-score |
| A | 0.60 | 0.61 | 0.60 |
| B | 0.62 | 0.65 | 0.64 |
| C | 0.67 | 0.67 | 0.67 |
| D | 0.87 | 0.84 | 0.85 |
| Accuracy |  |  | 0.71 |

*Tables 2-6. F1-scores for each model and class*

The scores highlight that the class ‘D’ has a significantly higher predictive accuracy (compared to other classes) in all models. The average F1-score for class ‘D’ is 0.818, while the average F1-score score for the classes ‘A’, ‘B’, and ‘C’ is 0.524, 0.590, and 0.616 respectively.

Taking these scores into account, the best model to take forward as a predictor is the Random Forest classifier. This model was chosen over the Multilayer Perceptron model as allows the ability to clearly record feature importance and provides a clearer insight into the different features.

*Continued on the next page*

**Feature Importance and Selection:**

Figure 4 shows the importances of different features as seen in the Random Forest model.

A bar graph with text

Description automatically generated

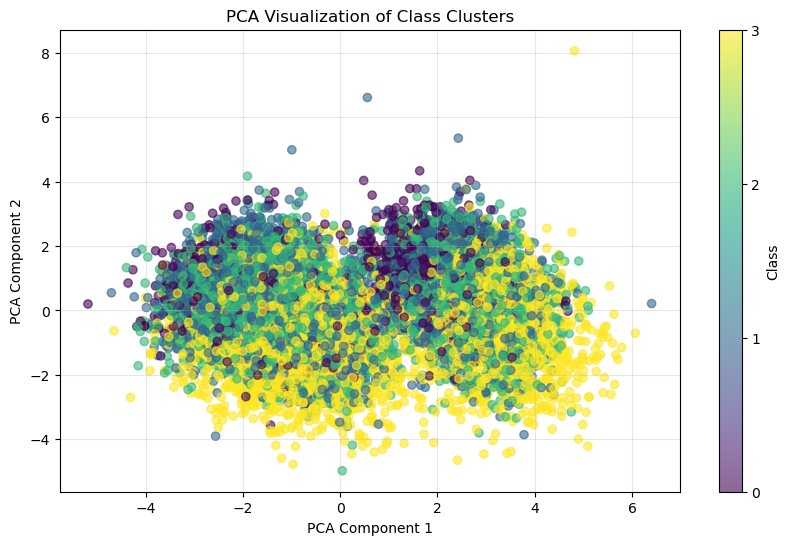
*Figure 4. Feature Importances seen in the Random Forest model.*

Figure 4 shows that the ‘sit and bend forward’ feature provides the most weight at 0.296. This substantiates the results seen in the pair plot in Figure 3.

*Continued on the next page*

**Additional Analysis of the Data:**

Due to the less than optimal performances in the test set from all models, Principal Component Analysis (PCA) was performed on the initial data to see if any clear clusters or divisions can be visualized in the data. PCA allows for multi-dimensional data to be compressed into a two-dimensional space. The results of the PCA are seen in Figure 5. Classes A-D are represented as Classes 0-3 respectively.



*Figure 5. PCA Visualization of Class Clusters*

The clusters for different classes significantly overlap, suggesting poor class separation in the PCA-reduced space. This could indicate:

* The features may not be very informative for distinguishing between classes.
* Higher-dimensional interactions may better represent the data.

Finally, k-Means clustering was performed and a t-SNE was created to further explore the data and see if distinct clusters can be seen. This is shown in Figure 6.

A colorful dots on a white background

Description automatically generated

*Figure 6. t-SNE Visualization of k-Means Clusters*

Subsequently, the true labels or true classes were assigned to the clustering pattern to determine whether the data is actually grouped into such clusters. The results can be seen in Figure 7.

A colorful dots on a white background

Description automatically generated

*Figure 7. t-SNE Visualization of True Labels*

Figure 7 shows no real pattern within the data, substantiating the results seen in the PCA that was done. This poor class separation seen in both the analyses further affirms the low test scores seen in the various models.

**Results in the unlabelled dataset:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Age | Gender | Height (cm) | Weight  (kg) | Body  Fat (%) | Dia | Sys | Grip Force | Sit and  bend forward  (cm) | Sit-ups  counts | Broad jump (cm) | Predicted Class |
| 36 | M | 177.5 | 73.1 | 13.8 | 82 | 135 | 50.8 | 19.1 | 64 | 252 | A |
| 35 | M | 176.4 | 77.7 | 21.2 | 74 | 129 | 34 | 11.7 | 41 | 213 | C |
| 34 | F | 162.6 | 52.1 | 20.4 | 72 | 115 | 31.3 | 28.2 | 65 | 208 | A |
| 32 | F | 156.3 | 60.8 | 38.9 | 68 | 120 | 25.2 | 5.3 | 13 | 90 | D |
| 27 | M | 165.4 | 65 | 24.5 | 82 | 123 | 36.5 | 10.2 | 46 | 179 | C |
| 24 | F | 162.6 | 58.9 | 26 | 74 | 131 | 26.4 | 14.7 | 31 | 172 | C |
| 64 | M | 168 | 73 | 20.5 | 84 | 127 | 41.8 | 12.6 | 27 | 190 | B |
| 25 | F | 158.1 | 56.3 | 29.6 | 60 | 105 | 21.9 | 6.1 | 19 | 141 | D |
| 22 | M | 179.1 | 74.7 | 13.2 | 71 | 121 | 40.9 | 21.2 | 62 | 259 | B |
| 37 | M | 175 | 63.7 | 17.1 | 76 | 113 | 40.7 | 9.1 | 41 | 220 | C |
| 32 | M | 180.8 | 88.9 | 25.1 | 89 | 137 | 46.3 | 6.5 | 47 | 210 | D |
| 23 | M | 176 | 83.9 | 22 | 77 | 133 | 37.4 | 22.7 | 61 | 230 | C |
| 55 | M | 171.1 | 76.4 | 24.2 | 78 | 135 | 41.2 | -0.2 | 35 | 183 | D |
| 57 | M | 175.4 | 79.9 | 24.3 | 85 | 137 | 42.1 | 18.8 | 40 | 223 | C |
| 27 | M | 189.2 | 78 | 11.6 | 79 | 136 | 49.1 | 13.6 | 49 | 248 | C |
| 47 | M | 176.1 | 80.2 | 22.8 | 73 | 123 | 43 | 2.3 | 20 | 196 | D |
| 37 | M | 174 | 75.2 | 18.5 | 82 | 142 | 39.7 | 20.5 | 50 | 224 | B |
| 23 | F | 172.3 | 64.3 | 29.5 | 76 | 120 | 29.6 | 15.2 | 46 | 177 | B |
| 28 | F | 160.9 | 62.4 | 34.9 | 73 | 121 | 23.9 | 10.1 | 43 | 163 | D |
| 30 | M | 180 | 83.6 | 13.4 | 99 | 145 | 49 | 17 | 51 | 264 | B |

Table 7 show the final predictions on the unlabelled dataset.

*Table 7. Predictions for unlabelled data*

A graph of individual individuality

Description automatically generated with medium confidence

To get a deeper insight into the data, Figure 8 highlights the number of individuals predicted into each class.

As seen in Figure 8, class ‘C’ contains the most individuals, followed by class ‘D’. Both these classes make up 65% of individuals.

*Figure 8. Number of Individuals in Each class*

**Discussion:**

5 different models were used for classification of the 20 individuals. Based on the performance of these models, the Random Forest classifier appeared the most efficient and accurate.

Looking at the labelled data, Figure 1 shows that the class ‘D’ has the most individuals. This led to more data being available for the model to be trained on, leading to a higher accuracy within the class. This is reflected in the class-by-class analysis in Tables 2-6, where the test scores seen in class ‘D’ are significantly higher for all models.

The pair plot seen in Figure 3 also shows clear clustering of the class ‘D’, while this clustering is not present for any other class. This could also suggest that based on the features, class ‘D’ is able to be categorized easily compared to the other classes.

Figure 3 also highlights the importance of the ‘sit and bend forward’ feature. This is confirmed by the plot of feature importance seen in Figure 4 as well. This could suggest that this feature is a strong determinant for the class assigned to individuals.

Apart from class ‘D’, there is no distinct pattern seen in any other classes. PCA and t-SNE were performed, which confirmed this idea. There were no clusters seen within the visualizations.

Based on the Random Forest model’s accuracy (0.71), it can be inferred that around 14 out of 20 individuals were labelled correctly.

In conclusion, while the Random Forest model performed better than the others, the only accurate predictions (test score > 0.75) that it is able to make are for individuals belonging to the ‘D’ class.

*(End of report)*