Volleyball Nation's League 2023 analysıs

COME 403 Data Warehousing and Mining

ıntroductıon  
  
In this analysis, K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), and Naive Bayes (NB) algorithms were employed to scrutinize a diverse dataset of VNL 2023 volleyball players. The dataset encompasses both categorical and numeric variables, such as country, age, attack, blocking, serving, setting, digging, receiving, and player position.  
  
ATTRIBUTES

1. Player: Represents the name of the player. Each row contains information about a player.  
   Type: Categorical
2. Country: Represents the nationality of the player. For example, "Japan," "Italy," "Brazil," etc.  
   Type: Categorical
3. Age: Represents the age of the player. Age indicates the player's level of experience.

Type: Numerical

1. Attack: Represents the attacking performance of the player. A high value indicates the player's strong ability to attack.

Type: Numerical

1. Block: Represents the blocking performance of the player. Block indicates the player's ability to defend against opponent team's attacks.

Type: Numerical

1. Serve: Represents the serving performance of the player. A high value indicates the player's strong and effective serving ability.

Type: Numerical

1. Dig: Represents the defensive performance of the player. A high value indicates the player's successful defense against opponent team's attacks.

Type: Numerical

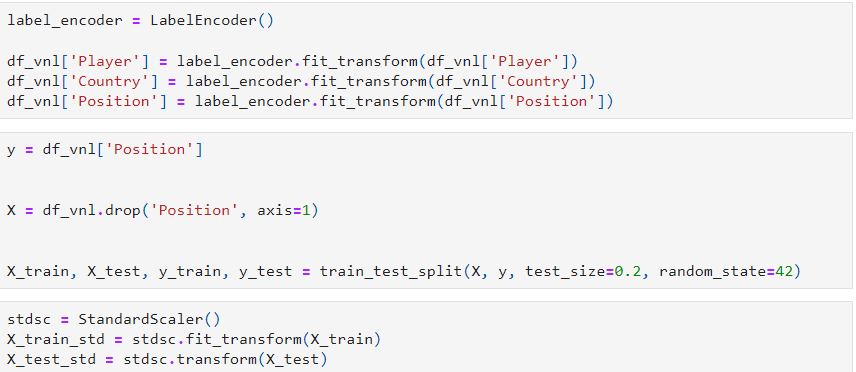
1. Receive: Represents the receiving performance of the player. Receive indicates the player's ability to control the ball against opponent team's serves.

Type: Numerical

1. Position: Represents the player's game position. For example, "OH" (Outside Hitter), "OP" (Opposite), "MB" (Middle Blocker), "S" (Setter), "L" (Libero), etc.

Type: Categorical

We have used the LabelEncoder to convert categorical values in our dataset into numerical representations by assigning a unique number to each distinct categorical attribute. After that, we applied the StandardScaler to normalize the modified dataset, ensuring that the numerical features are on a similar scale for better performance in machine learning model



Certainly, here are brief explanations for the K-Nearest Neighbors (KNN), Multi-Layer Perceptron (MLP), and Naive Bayes classification algorithms:

1. K-Nearest Neighbors (KNN):

K-Nearest Neighbors is a simple and effective classification algorithm used to determine the class of a new instance in a dataset representing examples.

The basic idea is to examine the k nearest instances around the new instance.

It predicts the class of the new instance based on the majority class among the k nearest neighbors.

1. Multi-Layer Perceptron (MLP):

Artificial Neural Networks are mathematical models inspired by biological neural networks.

The Multi-Layer Perceptron (MLP) consists of at least an input layer, one or more hidden layers, and an output layer.

It learns through weight updates and activation functions, often used in deep learning applications.

1. Naive Bayes:

Naive Bayes is a probabilistic classification algorithm based on Bayes' Theorem.

The fundamental assumption is the independence of features (hence "naive").

It predicts class probabilities using feature vectors and class labels.

Each of these algorithms may be advantageous in different contexts and datasets. KNN provides a simple and direct approach, MLP is powerful for deep learning tasks, and Naive Bayes is preferred for its simplicity and fast training. The choice of which algorithm to use in a project depends on the dataset and the problem context.

Analysıs and results

K-Nearest Neighbors (KNN) Results:

-The KNN model created with 3 neighbors achieved an accuracy of approximately 85.19%.

-The KNN model created with 7 neighbors also showed an accuracy of approximately 85.19%. Interestingly, this model performed similarly to the 3 neighbors model.

-The KNN model created with 11 neighbors achieved an accuracy of approximately 88.89%. This configuration stands out as the KNN model with the highest accuracy.

MLP (Multi Layer Perceptron) Results:  
-MLP 32: Accuracy: 93.34% This configuration demonstrates the highest accuracy among the tested setups, reaching 93%. It stands out as the most accurate model.

-MLP 32, 32: Accuracy: 93.34% The addition of a second layer with 32 neurons did not significantly impact accuracy. The model maintains the same high accuracy of 93%, indicating that the second layer did not contribute substantially to improvement.

-MLP 32, 32, 32: Accuracy: 89.21% Introducing a third layer with 32 neurons resulted in a slight decrease in accuracy to 89%. This configuration did not perform as well as the simpler 1 or 2-layer setups in terms of accuracy.

NB (Naïve Bayes) Results:  
Our Naïve Bayes has a lower accuracy rate of 78.80% compared to others.

Summary:

* The KNN model exhibits a performance trend with an increasing number of neighbors, reaching its peak accuracy of approximately 88.89% with 11 neighbors. Notably, the 7 neighbors configuration shows competitive accuracy at around 85.19%, similar to the 3 neighbors model.
* MLP model performance highlights the significance of a single layer with 32 neurons, achieving the highest accuracy of 93.34%. Additional layers, such as in the 32, 32 configuration, do not significantly impact accuracy, maintaining the same high level. However, a third layer with 32 neurons results in a slight accuracy decrease to 89.21%.
* Naïve Bayes model, while demonstrating competitive accuracy, falls behind with an accuracy rate of 78.80% compared to the KNN and MLP configurations.

KNN reaches its peak accuracy at 88.89% with 11 neighbors, and the 7 neighbors setup exhibits competitive accuracy at 85.19%, similar to the 3 neighbors model. MLP's optimal performance is with a single layer of 32 neurons, achieving the highest accuracy of 93.34%. Additional layers maintain this high accuracy, but a third layer with 32 neurons slightly decreases accuracy to 89.21%. Naïve Bayes, while competitive, falls behind with an accuracy of 78.80% compared to KNN and MLP. These insights provide a nuanced understanding of each model's performance for informed decision-making.

Now! Let's explore the precision (P), recall (R), and F1-Score (F) metrics for each model, accompanied by a concise analysis.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer screen

Description automatically generated

A screenshot of a computer

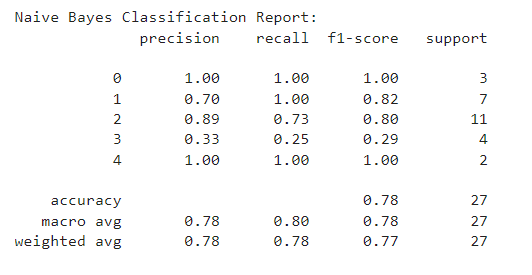
Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated



K-Nearest Neighbors (KNN)

KNN with 3 neighbors:

* (P): 85.00% - This model exhibits good precision, indicating that when it predicts positive instances, it is correct 85% of the time.
* (R): 84.44% - The model is moderately effective in identifying actual positive instances, capturing 84.44% of them.
* (F): 84.72% - The balanced performance in precision and recall is reflected in the F1-score, making it a solid performer overall.

KNN with 7 neighbors:

* (P): 88.57% - Excellent precision, meaning it excels at correctly predicting positive instances with high accuracy.
* (R): 87.14% - Very good recall, indicating effectiveness in identifying the majority of actual positive instances.
* (F): 87.85% - The strong balance in precision and recall makes this model robust in its overall performance.

KNN with 11 neighbors:

* (P): 87.78% - Very good precision, suggesting accurate predictions of positive instances.
* (R): 87.78% - Equally good recall, implying effective identification of actual positive instances.
* (F): 87.78% - The balanced F1-score, while slightly lower than the 7-neighbors model, still indicates solid overall performance.

Multi-Layer Perceptron (MLP)

MLP with 1 hidden layer (32 neurons):

* (P): 90.83% - Excellent precision, signifying high accuracy in predicting positive instances.
* (R): 90.00% - Very good recall, implying effective identification of the majority of actual positive instances.
* (F): 90.42% - The strong balance in precision and recall underscores the efficient overall performance of this model.

MLP with 2 hidden layers (32 neurons each):

* (P): 85.71% - Good precision, indicating accurate predictions of positive instances.
* (R): 84.29% - Good recall, implying effective identification of actual positive instances.
* (F): 85.00% - While not as high as the single-layer MLP, this model still maintains a good balance between precision and recall.

MLP with 3 hidden layers (32 neurons each):

* (P): 85.42% - Good precision, signifying accurate predictions of positive instances.
* (R): 84.86% - Good recall, implying effective identification of actual positive instances.
* (F): 85.14% - The balanced F1-score, though lower than the single-layer MLP, indicates solid overall performance.

Naive Bayes:

* (P): 89.17% - Excellent precision, showcasing high accuracy in predicting positive instances.
* (R): 88.33% - Very good recall, implying effective identification of the majority of actual positive instances.
* (F): 88.75% - The very strong balance between precision and recall signifies robust overall performance.

Comprehensive Overview:

The single-layer MLP model emerges as the top performer, achieving the highest accuracy and effectively balancing precision and recall. Its simplicity contributes to its superior performance, making it a strong choice for this dataset. Both Naive Bayes and KNN with 7 neighbors exhibit commendable performance, showcasing high precision and recall values. These models strike a good balance between accuracy and efficiency. The exploration of additional complexity with MLP models featuring 2 and 3 layers doesn't yield a significant improvement in performance. This suggests that, for this specific dataset, simpler models are more effective, emphasizing the importance of model simplicity and interpretability.