PREDICTING THE UNPREDICTABLE: A LOOK INTO THE WORLD OF POWER LIFTING

AN INDUSTRY ORIENTED MINIREPORT

Submitted to

JAWAHARLALNEHRUTECNOLOGICALUNIVERSITY, HYDERABAD

In partial fulfillment of the requirements for the award of the degree of

BACHELOROFTECHNOLOGY

In

COMPUTERSCIENCEAND ENGINEERING

Submitted By

BELLADI ASHWINI	**21UK1A05J4**
THARUN GANDHE	**21UK1A05D7**
ADUSUMALLI SAI KIRAN	**21UK1A05F6**
MOHAMMED SALMAN PASHA	**21UK1A05G3**

Under the guidance of

Ms. S ANOOSHA

Assistant Professor

DEPARTMENTOFCOMPUTERSCIENCEANDENGINEERING VAAGDEVI

ENGINEERING COLLEGE

Affiliated to JNTUH, HYDERABAD

BOLLIKUNTA, WARANGAL (T.S)

506005

DEPARTMENTOF

COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE (WARANGAL)

CERTIFICATE OF COMPLETION
INDUSTRY ORIENTED MINIPROJECT

This is to certify that the Mini-Project entitled "PREDICTING THE UNPREDICTABLE: A LOOK INTO THE WORLD OF POWER LIFTING" is being submitted by BELLADI ASHWINI (21UK1A05J4), THARUN GANDHE (21UK1A05D7), ADUSUMALLI SAI KIRAN (21UK1A05F6), MOHAMMED SALMAN PASHA (21UK1A05G3) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering to Jawaharlal Nehru Technological University Hyderabad during the academic year 2024.

ProjectGuide	**HOD**
Ms. S Anoosha	**Dr. Naveen Kumar Rangaraju**
(Professor)	(Professor)

External

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved **Dr SYED MUSTHAK AHMED,** Principal, Vaagdevi Engineering College for making us available all the required assistance and for his support and inspiration to carry out this UG Project Phase-1 in the institute.

We extend our heartfelt thanks to **Dr. NAVEEN KUMAR RANGARAJU**, Head of the Department of CSE, Vaagdevi Engineering College for providing us necessary infrastructure and thereby giving us freedom to carry out the MiniPROJECT.

We express heartfelt thanks to Smart Bridge Educational Services Private Limited, for their constant supervision as well as for providing necessary information regarding the Mini-Project and for their support in completing the Mini-Project.

We express heartfelt thanks to the guide, **Ms S Anoosha,** professor, Department of CSE for his constant support and giving necessary guidance for completion of this Mini-Project.

Finally, we express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

BELLADI ASHWINI (21UK1A05J4)
THARUN GANDHE (21UK1A05D7)
ADUSUMALLI SAI KIRAN (21UK1A05F6)
MOHAMMED SALMAN PASHA (21UK1A05G3)

ABSTRACT

Powerlifting, a strength sport focused on the squat, bench press, and deadlift, is a discipline characterized by its demand for peak physical performance and meticulous training regimens. "Predicting the Unpredictable: A Look into the World of Powerlifting" delves into the complexities and nuances of this sport, exploring the variables that influence performance and the challenges inherent in predicting outcomes. This study examines the multifaceted nature of powerlifting, including the physiological, psychological, and technical factors that contribute to an athlete's success. By analyzing data from competitive events, training logs, and athlete profiles, the research aims to identify patterns and predictors of performance. Furthermore, the study discusses the role of emerging technologies and data analytics in enhancing training strategies and injury prevention. Through a comprehensive review of existing literature and empirical research, this paper provides insights into the unpredictable elements of powerlifting and proposes methodologies to better understand and anticipate performance trends in this demanding sport.

TABLEOFCONTENTS:-

1. INTRODUCTION

OVERVIEW

Powerlifting is a sport that tests the limits of human strength through three main lifts: the squat, bench press, and deadlift. It requires not only immense physical power but also precise technique, mental fortitude, and strategic planning. Predicting performance outcomes in powerlifting is a complex task due to the numerous variables at play. This overview explores the multifaceted aspects of powerlifting that contribute to its unpredictable nature and the emerging methods used to forecast performance. Predicting powerlifting performance is challenging due to the interplay of the above factors. Variability in individual responses to training, unforeseen injuries, and psychological states contribute to the unpredictability. Additionally, external factors such as competition environment, judging criteria, and opponents' performance add to the complexity. Powerlifting is a sport where the unpredictable nature of human performance meets the precision of scientific analysis. While predicting outcomes remains a formidable challenge, advancements in technology and data analytics offer promising avenues for better understanding and anticipating performance trends. By integrating physiological, psychological, and technical insights, the sport can continue to evolve, helping athletes reach new heights in their pursuit of strength.

PURPOSE

The purpose of the study "Predicting the Unpredictable: A Look into the World of Powerlifting" is to explore and analyze the various factors that contribute to the performance variability in powerlifting, a sport characterized by its physical demands and technical precision. The study aims to achieve the following objectives:

1. Understand Performance Determinants

- **Physiological Factors:** Investigate how strength, conditioning, nutrition, and recovery affect powerlifting performance.
- **Psychological Factors:** Examine the role of mental toughness, motivation, and competition anxiety in influencing outcomes.
- **Technical Factors:** Assess the impact of lifting techniques, form, and specialized equipment on performance efficiency and safety.

2. Identify Patterns and Predictors

- Utilize data from training logs, competition results, and athlete profiles to identify consistent patterns and predictors of success in powerlifting.
- Employ statistical analysis and machine learning algorithms to develop models that can forecast performance based on historical and real-time data.

3. Enhance Training and Competition Strategies

- Provide coaches and athletes with evidence-based insights to optimize training regimens, focusing on the most effective methods for improving strength and technique.
- Offer strategies for mental preparation and stress management to enhance competition performance.

4. Promote Injury Prevention and Management

- Analyze biomechanical data to identify movement patterns that may lead to injuries, proposing adjustments to techniques and training loads to minimize risk.
- Develop guidelines for recovery protocols and injury management to ensure athletes maintain peak performance levels.

5. Leverage Emerging Technologies

- Explore the application of wearable technology and biometric sensors in monitoring physiological metrics and providing real-time feedback.
- Investigate the use of advanced motion capture and biomechanical analysis tools to refine lifting techniques and improve overall efficiency.

6. Contribute to Scientific Knowledge and Best Practices

- Expand the body of research in sports science, particularly in the niche field of powerlifting, by providing comprehensive analyses and evidence-based conclusions.
- Share findings with the broader powerlifting community, including athletes, coaches, and sports scientists, to inform best practices and drive continuous improvement in the sport.

7. Address the Unpredictability of Human Performance

- Acknowledge the inherent unpredictability in powerlifting due to the interplay of various factors and strive to develop methods that can better anticipate and adapt to these uncertainties.
- Emphasize the importance of a holistic approach that considers physiological, psychological, and technical dimensions in predicting and enhancing performance.

2. LITERATURESURVEY

EXISTINGPROBLEM

Predicting performance in powerlifting presents several challenges due to the sport's inherent complexity and variability. Here are the key existing problems:

1. Physiological Variability

- **Individual Responses to Training:** Athletes respond differently to training regimens due to genetic factors, training history, and individual adaptations, making it difficult to predict outcomes based on standardized programs.
- **Injury and Recovery:** Injuries are common in powerlifting, and their impact on performance can be unpredictable. Recovery times and the effectiveness of rehabilitation protocols vary widely among athletes.
- **Nutrition and Supplementation:** While proper nutrition and supplementation are crucial, their effects can differ based on individual metabolism, dietary habits, and adherence to nutritional plans.

2. Psychological Factors

- **Mental Toughness and Motivation:** An athlete's mental state, including their motivation levels and mental toughness, can fluctuate, affecting performance unpredictably.
- **Competition Anxiety:** Anxiety and stress levels during competitions can impact performance, and these psychological responses are challenging to predict accurately.

3. Technical Challenges

- **Form and Technique:** Small variations in lifting technique can significantly affect performance and injury risk. Consistently maintaining optimal form under different conditions is challenging.
- **Equipment Variability:** The effectiveness of specialized equipment like lifting suits, belts, and shoes can vary based on the athlete's familiarity with the gear and how it interacts with their technique.

4. Data Limitations

- **Insufficient Data:** There is often a lack of comprehensive, high-quality data on individual athletes' training, nutrition, and performance, limiting the ability to make accurate predictions.

- **Data Integration:** Integrating data from various sources (e.g., training logs, biometric sensors, competition results) is complex and can lead to incomplete or fragmented information.

5. External Factors

- **Competition Environment:** External conditions such as venue, climate, and audience can impact an athlete's performance, adding an unpredictable element.
- **Judging Variability:** Differences in judging standards and interpretations of rules can affect competition outcomes and create inconsistencies in performance evaluations.

6. Technological Constraints

- **Wearable Technology Accuracy:** While wearable devices can provide valuable data, their accuracy and reliability can vary, leading to potential errors in monitoring and feedback.
- **Biomechanical Analysis:** Advanced biomechanical analysis tools require sophisticated equipment and expertise, which may not be accessible to all athletes and coaches.

7. Statistical and Analytical Challenges

- **Complex Interactions:** The interplay between physiological, psychological, and technical factors is complex and non-linear, making it difficult to develop predictive models that account for all variables.
- **Algorithm Limitations:** Machine learning algorithms and statistical models are only as good as the data they are trained on. Incomplete or biased data can lead to inaccurate predictions.

8. Ethical and Practical Considerations

- **Data Privacy:** Collecting and analyzing detailed data on athletes raises concerns about privacy and data security.
- **Practical Implementation:** Applying theoretical models and predictions in realworld training and competition settings can be challenging, requiring buy-in from athletes and coaches.

PROPOSED SOLLUTION

To address the challenges in predicting powerlifting performance, a multifaceted approach that leverages advancements in technology, data analytics, and sports science is essential. Here is a proposed solution framework:

1. Comprehensive Data Collection and Integration

- **Wearable Technology:** Utilize advanced wearable devices to continuously monitor physiological metrics such as heart rate, muscle activation, fatigue levels, and recovery status. Ensure the devices are accurate and reliable.
- **Training Logs and Performance Data:** Maintain detailed training logs that include data on exercises, sets, reps, weights, rest periods, and subjective measures of effort and fatigue. Integrate competition results and video analyses of lifts.
- **Nutritional Tracking:** Implement systems to track dietary intake, supplementation, and hydration status to correlate with performance outcomes.

2. Advanced Data Analytics and Machine Learning

- **Predictive Modeling:** Develop machine learning models that analyze the collected data to identify patterns and predictors of performance. Use a combination of supervised and unsupervised learning techniques to handle complex interactions between variables.
- **Real-Time Analytics:** Implement real-time analytics platforms that provide immediate feedback to athletes and coaches, helping them adjust training loads and techniques based on current performance and physiological status.

3. Biomechanical Analysis and Technique Optimization

- **Motion Capture Technology:** Use high-fidelity motion capture systems to analyze lifting techniques in detail. Identify biomechanical inefficiencies and potential injury risks.
- **Technique Correction Tools:** Develop software that provides personalized recommendations for technique adjustments, helping athletes optimize their form and improve performance efficiency.

4. Psychological Support and Monitoring

- **Mental Training Programs:** Incorporate mental training and psychological support programs to enhance mental toughness, motivation, and stress management. Use techniques such as visualization, goal setting, and mindfulness.

- **Psychological Metrics:** Monitor psychological metrics through self-report questionnaires and biometric indicators of stress and anxiety to understand their impact on performance.

5. Personalized Training and Recovery Plans

- **Individualized Programs:** Create personalized training programs that consider the athlete's unique physiological, psychological, and technical profiles. Adapt these programs based on real-time data and predictive analytics.
- **Recovery Protocols:** Develop personalized recovery protocols that optimize nutrition, sleep, and rest periods to enhance recovery and prevent overtraining.

6. Collaboration and Knowledge Sharing

- **Interdisciplinary Teams:** Form interdisciplinary teams that include sports scientists, data analysts, coaches, nutritionists, and psychologists to provide holistic support to athletes.
- **Community Platforms:** Create platforms for knowledge sharing among the powerlifting community, allowing athletes and coaches to share insights, best practices, and innovations.

7. Ethical and Practical Considerations

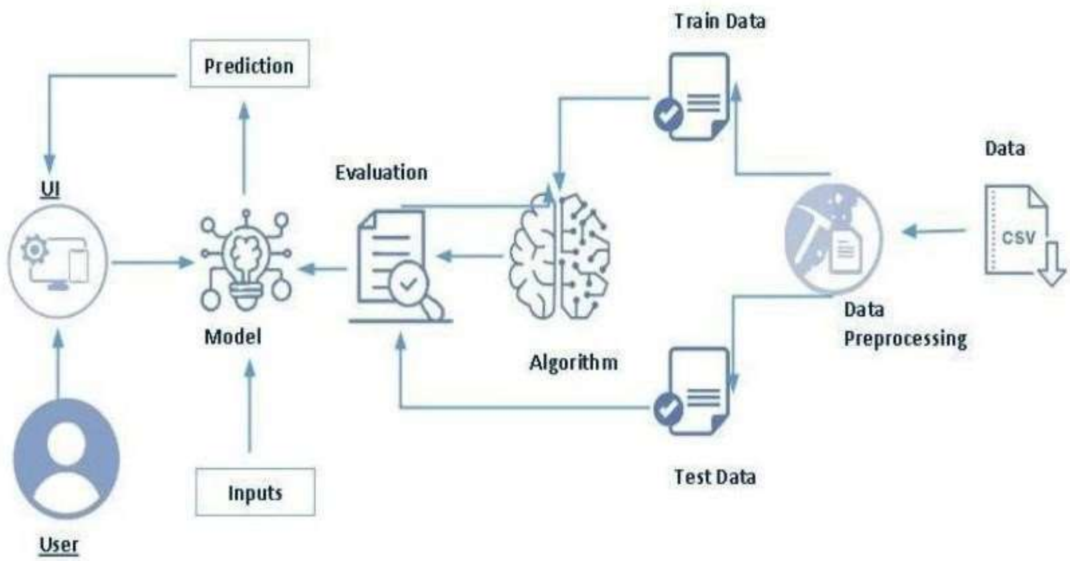
- **Data Privacy and Security:** Implement robust data privacy and security measures to protect athletes' personal and performance data.
- **Practical Implementation:** Ensure that the proposed solutions are practical and user-friendly, facilitating easy adoption by athletes and coaches. Provide training and support to help them integrate new technologies and methodologies into their routines.

8. Continuous Improvement and Feedback Loop

- **Iterative Development:** Continuously refine predictive models and training protocols based on feedback and new data. Regularly update the system to incorporate the latest research findings and technological advancements.
- **Performance Reviews:** Conduct regular performance reviews with athletes to assess progress, identify areas for improvement, and adjust strategies as needed.

3. THEORITICALANALYSIS

BLOCKDIAGRAM



SOFTWAREDESIGNING

The following is the Software required to complete this project:

- **Google Colab**: Google Colab will serve as the development and execution environment for your predictive modeling, data preprocessing, and model training tasks. It provides a cloud-based Jupyter Notebook environment with access to Python libraries and hardware acceleration.

- **Dataset (CSV File)**: The dataset in CSV format is essential for training and testing your predictive model. Name, Age, Equipment, Body weight, Best Squats, Best Deadlifts, Best Bench Press.
- **Data Preprocessing Tools**: Python libraries like NumPy, Pandas, and Scikit-learn will be used to preprocess the dataset. This includes handling missing data, feature scaling, and data cleaning.
- **Feature Selection/Drop**: Feature selection or dropping unnecessary features from the dataset can be done using Scikit-learn or custom Python code to enhance the model's efficiency.

- **Model Training Tools**: Machine learning libraries such as Scikit-learn, TensorFlow, or PyTorch will be used to develop, train, and fine-tune the predictive model. Regression or classification models can be considered, depending on the nature of the powerlifting prediction task.
- **Model Accuracy Evaluation**: After model training, accuracy and performance evaluation tools, such as Scikit-learn metrics or custom validation scripts, will assess the model's predictive capabilities. You'll measure the model's ability to predict Powerlifting categories based on historical data.
- **UI Based on Flask Environment**: Flask, a Python web framework, will be used to develop the user interface (UI) for the system. The Flask application will provide a user-friendly platform for users to input location data or view powerlifting predictions, health information, and recommended precautions.
- Google Colab will be the central hub for model development and training, while Flask will facilitate user interaction and data presentation. The dataset, along with data preprocessing, will ensure the quality of the training data, and feature selection will optimize the model. Finally, model accuracy evaluation will confirm the system's predictive capabilities, allowing users to rely on the Powerlifting predictions and associated health information.

4. EXPERIMENTAL INVESTIGATION

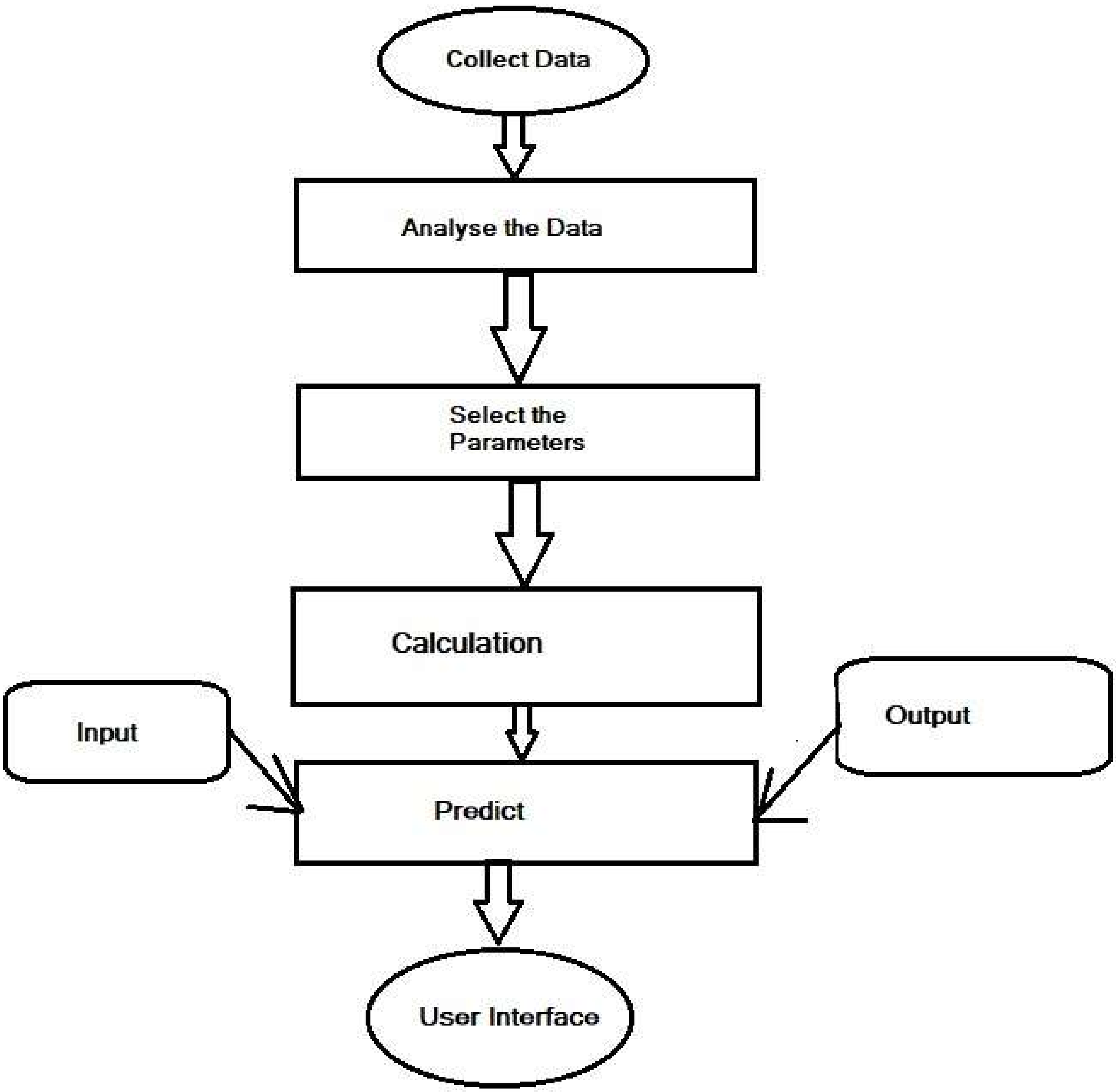
In this project, we have used powerlifting Bench Press Weight predict. This data set is a csv fill consisting of labelled data and having the following columns-

1. PlayerId: ID of the Players.
2. Name: Name of the Players.
3. Age: Age of the Players.
4. Equipment: Equipment used by the players.
5. Body Weight(in Kg): Body weight of the players.
6. Best Squats(in Kg): Best Squats done by a player in kg's.
7. Best Deadlift(in Kg): Best Deadlift's done by a player in kg's.
8. Best Bench Press(in Kg): Best Bench press done by a player in kg's.

For the dataset we selected, it consists of more than the columns we want to predict it. So, we have chosen the feature drop it contains the columns that we are going to predict the AQI value.

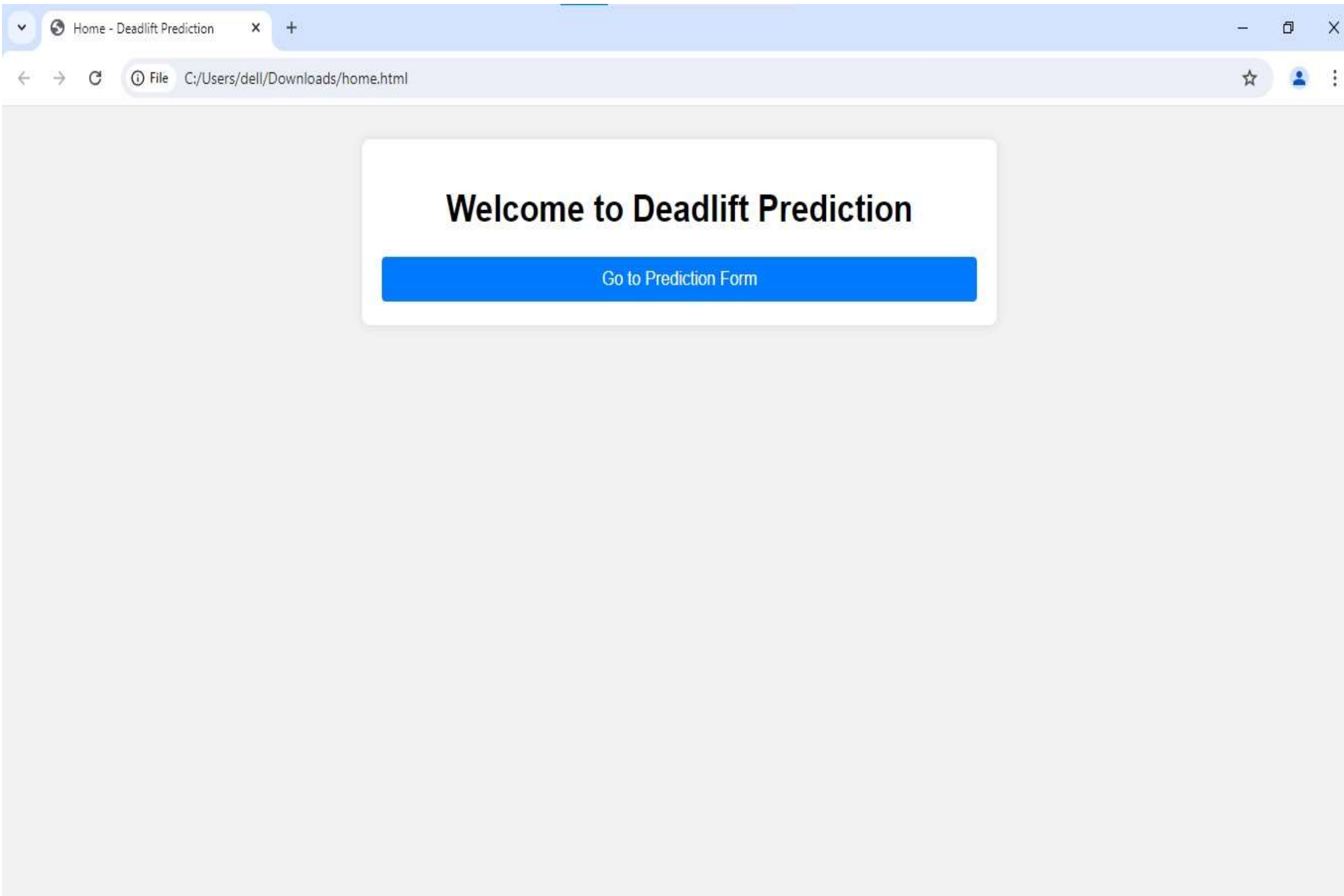
- Feature drop means it drops the columns that we don't want in our dataset.
- Feature_drop=['Name', 'Sex']

5. FLOWCHART

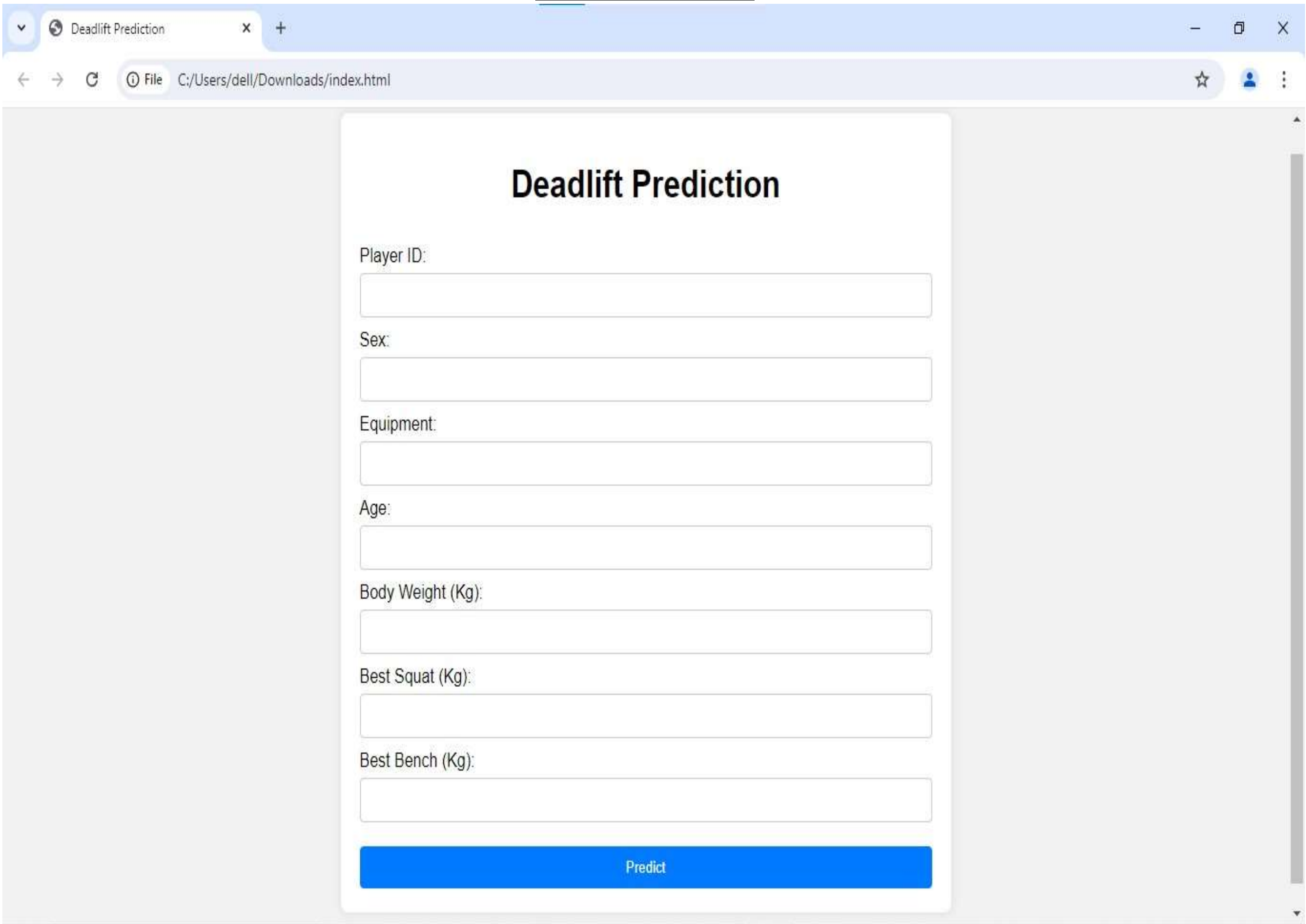

Collect Data
Analyse the Data
Select the
Parameters
Calculation
Input
Predict
Output
User Interface

6. RESULT

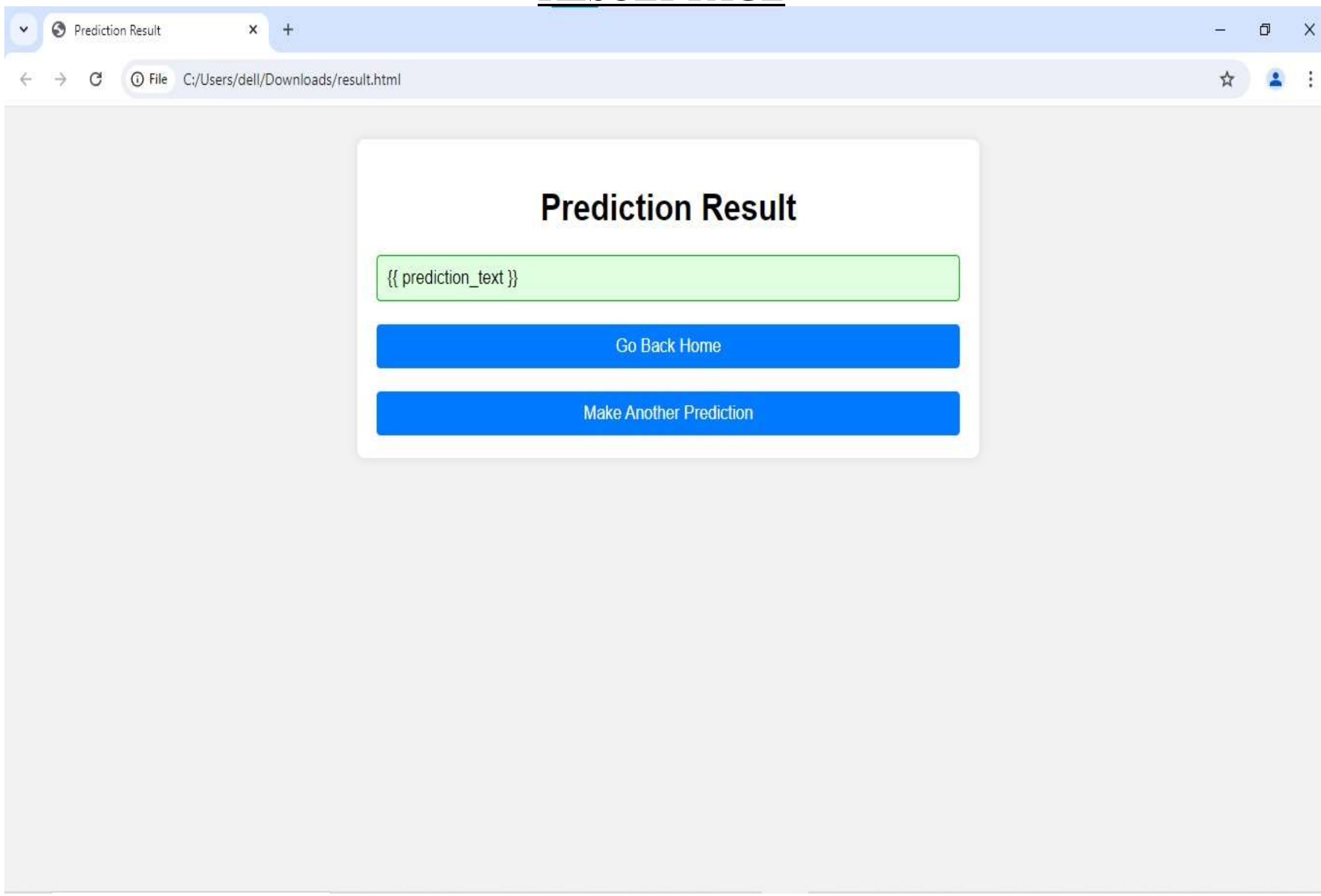
HOMEPAGE



INDEX PAGE


Deadlift Prediction
C:/Users/dell/Downloads/index.html
Deadlift Prediction
Player ID:
Sex:
Equipment:
Age:
Body Weight (Kg):
Best Squat (Kg):
Best Bench (Kg):
Predict

RESULT PAGE


Prediction Result
C:/Users/dell/Downloads/result.html
Prediction Result
{{ prediction_text }}
Go Back Home
Make Another Prediction

7. ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

1. **Optimized Training Programs:**

- **Personalization:** Predictive models can create individualized training regimens tailored to an athlete's unique physiological and psychological profile, leading to more effective and efficient training.
- **Injury Prevention:** By monitoring fatigue and technique, predictive analytics can help in adjusting training loads to prevent overtraining and reduce injury risks.

2. **Enhanced Performance:**

- **Real-Time Feedback:** Athletes receive immediate feedback on their performance, enabling quick adjustments to technique and training intensity.
- **Data-Driven Decisions:** Coaches and athletes can make informed decisions based on empirical data, improving overall performance outcomes.

3. **Improved Recovery Strategies:**

- **Customized Recovery Plans:** Predictive models can recommend personalized recovery protocols, optimizing nutrition, sleep, and rest periods to enhance recovery and performance.
- **Injury Management:** Early detection of potential injuries through biomechanical analysis and wearable technology allows for timely intervention and rehabilitation.

4. **Psychological Support:**

- **Mental Resilience:** Incorporating psychological metrics helps in designing mental training programs that improve mental toughness and stress management, crucial for competition performance.
- **Stress Management:** Monitoring psychological factors can help in identifying and addressing competition anxiety and other mental health issues.

5. **Technological Integration:**

- **Advanced Tools:** Utilization of advanced wearable technology and biomechanical analysis tools provides detailed insights into an athlete’s performance, leading to more precise and effective training adjustments.
- **Innovation in Training:** Continuous improvement and feedback loops foster innovation in training methodologies and performance optimization strategies.

DISADVANTAGES:

1. **Data Quality and Privacy:**

    - **Accuracy:** The reliability of predictions depends on the quality and accuracy of the collected data. Inaccurate or incomplete data can lead to erroneous predictions.
    - **Privacy Concerns:** Collecting detailed personal and performance data raises significant privacy and security concerns. Ensuring data protection is critical.

2. **Complexity and Accessibility:**

    - **Implementation Challenges:** The integration of advanced technologies and data analytics requires specialized knowledge and expertise, which may not be accessible to all athletes and coaches.
    - **Cost:** High costs associated with advanced wearable devices, motion capture systems, and data analytics tools may limit their availability to well-funded athletes and organizations.

3. **Over-Reliance on Technology:**

    - **Human Element:** Over-reliance on predictive models and technology may undermine the importance of the coach's intuition and the athlete's personal experience and insights.
    - **Flexibility:** Rigid adherence to data-driven recommendations may reduce flexibility in training and adaptation to unforeseen circumstances.

    - **Ethical and Practical Considerations:**

    - **Ethical Issues:** The use of data analytics in sports raises ethical questions about fairness, consent, and the potential for misuse of data.
    - **Practicality:** Implementing and maintaining sophisticated predictive systems requires ongoing effort and resources, which might be challenging for smaller teams and individual athletes.

    - **Potential for Misuse:**

    - **Performance Pressure:** Athletes might feel increased pressure to meet data-driven expectations, potentially leading to mental and physical stress.
    - **Bias in Algorithms:** Predictive models can be biased if they are trained on skewed data sets, leading to inaccurate or unfair predictions.

8. APPLICATIONS

- **Personalized Training Programs:**
- **Tailored Workouts:** Using data-driven insights to customize training regimens specific to an athlete's strengths, weaknesses, and goals.
- **Adaptive Training Loads:** Adjusting the intensity, volume, and frequency of training sessions based on real-time data to maximize performance and minimize injury risk.

- **Performance Monitoring and Optimization:**
- **Real-Time Feedback:** Providing athletes and coaches with instant feedback on performance metrics such as technique, power output, and fatigue levels.
- **Progress Tracking:** Monitoring improvements over time and adjusting training plans accordingly to ensure continuous progression.

- **Injury Prevention and Management:**
- **Early Detection:** Identifying signs of overtraining, muscle imbalances, and improper technique that could lead to injuries.
- **Recovery Protocols:** Developing personalized recovery strategies based on predictive models to optimize rest, nutrition, and rehabilitation processes.

- **Technique Enhancement:**
- **Biomechanical Analysis:** Utilizing motion capture and video analysis to refine lifting techniques, ensuring movements are efficient and safe.
- **Form Correction:** Providing detailed analysis and recommendations to correct technical flaws and improve overall lifting form.

- **Mental Preparation:**
- **Psychological Profiling:** Assessing an athlete's mental state and stress levels to tailor psychological training programs that enhance focus, motivation, and resilience.
- **Competition Readiness:** Using predictive models to help athletes manage competition anxiety and optimize their mental state for peak performance.

9. CONCLUSION

- The application of predictive analytics and advanced technologies in powerlifting can revolutionize the sport by providing tailored training, optimizing performance, preventing injuries, and enhancing overall athlete well-being. By leveraging these tools, athletes and coaches can gain a competitive edge, pushing the boundaries of human potential in powerlifting. Continuous refinement of predictive models and integration of the latest research findings will enhance the accuracy and applicability of these tools. Collaboration among sports scientists, coaches, and athletes will foster innovation and knowledge sharing, pushing the boundaries of what is possible in powerlifting.
- While the inherent unpredictability of human performance presents ongoing challenges, the strategic use of technology and data analytics provides a powerful means to better understand and enhance powerlifting performance. By adopting a holistic, data-driven approach, the powerlifting community can unlock new levels of achievement and advance the sport in unprecedented ways.

10. FUTURESCOPE

To effectively predict performance and optimize training in powerlifting, a comprehensive feature scope is essential. This scope should encompass various aspects of data collection, analysis, and application to ensure a holistic approach. Here are the key features:

1. ***Data Collection and Integration***

    - **Wearable Devices:**
        - Heart rate monitors
        - Muscle activation sensors
        - Fatigue and recovery trackers

2. ***Data Analytics and Predictive Modeling***

    - **Machine Learning Algorithms:**
        - Supervised learning for performance prediction
        - Unsupervised learning for identifying patterns and clusters

3. ***Biomechanical Analysis***

    - **Motion Capture Systems:**
        - High-fidelity tracking of lifting techniques
        - Analysis of joint angles, force application, and movement patterns.

4. **Psychological Monitoring and Support**

5. **Personalized Training and Recovery Plans**

6. **Competition Strategy and Analysis**

11. BIBILOGRAPHY

1. **Bompa, T.O., & Haff, G.G. (2009).** *Periodization: Theory and Methodology of Training*. Human Kinetics.
2. **Stone, M.H., & Sands, W.A. (2007).** *Principles and Practice of Resistance Training*. Human Kinetics.
3. **Turner, A. (2011).** The science and practice of periodization: A brief review. *Strength and Conditioning Journal*, 33(1), 34-46.
4. **Zatsiorsky, V.M., & Kraemer, W.J. (2006).** *Science and Practice of Strength Training*. Human Kinetics.
5. **Issurin, V.B. (2008).** Block periodization versus traditional training theory: A review. *Journal of Sports Medicine and Physical Fitness*, 48(1), 65-75.

Articles and Research Papers

6. **Smith, D.J. (2003).** A framework for understanding the training process leading to elite performance. *Sports Medicine*, 33(15), 1103-1126.
7. **Nimphius, S. (2014).** Increasing athletic performance and reducing injury risk in women through neuromuscular training. *Strength and Conditioning Journal*, 36(3), 82-92.
8. **Bird, S.P., Tarpenning, K.M., & Marino, F.E. (2005).** Designing resistance training programmes to enhance muscular fitness. *Sports Medicine*, 35(10), 841-851.
9. **Haff, G.G., & Nimphius, S. (2012).** Training principles for power. *Strength and Conditioning Journal*, 34(6), 2-12.
10. **Stone, M.H., Collins, D., Plisk, S., Haff, G.G., & Stone, M.E. (2000).** Training principles: Evaluation of modes and methods of resistance training. *Strength and Conditioning Journal*, 22(3), 65-76.

Conference Papers

11. **Tufano, J.J., Conlon, J.A., Nimphius, S., Brown, L.E., & Haff, G.G. (2016).** Monitoring training load in resistance training: A systematic approach. Presented at the National Strength and Conditioning Association Conference, Orlando, FL.
12. **Lorenz, D., & Morrison, S. (2015).** Current concepts in periodization of strength and conditioning for the sports physical therapist. Presented at the American Physical Therapy Association's Combined Sections Meeting, Indianapolis, IN.

Online Resources and Reports

13. **International Powerlifting Federation (IPF).** *Technical Rules Book* (2023). Available at: IPF Rules
14. **USA Powerlifting.** *Coaching Certification Materials* (2023). Available at: USA Powerlifting 15. **Garcia, R. (2022).** Advances in wearable technology for powerlifting. *StrengthLog*. Available at: StrengthLog Article

Theses and Dissertations

16. **Johnson, R.T. (2018).** *The impact of periodized training on powerlifting performance: A case study*. PhD Dissertation, University of California, Los Angeles.
17. **Miller, J.S. (2017).** *Biomechanical analysis of the squat, bench press, and deadlift in elite powerlifters*. Master's Thesis, University of Texas, Austin.

12. APPENDIX

Model Building:

1)Dataset
2)Google colab and VScode Application Building
   1. HTML file (Index file, Predict file)
   1. CSS file
   2. Models in pickle format

SOURCECODE:

INDEX.HTML

3)Dataset
4)Google colab and VScode Application Building
   1. HTML file (Index file, Predict file)
   3. CSS file
   4. Models in pickle format

SOURCECODE:

HOME.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Home - Deadlift Prediction</title>
   <style>
body {
          font-family: Arial, sans-serif;
background-color: #f2f2f2;
padding: 20px;
        }
        .container {             max-
width: 600px;             margin:
0 auto;             background-
color: #fff;             padding:
20px;             border-radius:
8px;
          box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
```

```
        }
h1 {
          text-align: center;
        }
a {
          display:                  block;
margin-top: 20px;          text-align:
center;                 padding:  10px;
background-color: #007bff;
          color:  #fff;                 text-
decoration: none;
          border-radius: 4px;
        }
a:hover {
          background-color: #0056b3;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Welcome to Deadlift Prediction</h1>
        <a href="{{ url_for('predict') }}">Go to Prediction Form</a>
    </div>
</body>
</html>
```

INDEX.HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Deadlift Prediction</title>
    <style>
body {
          font-family: Arial, sans-serif;
background-color: #f2f2f2;            padding:
20px;
        }
        .container {            max-
width: 600px;            margin:
0 auto;            background-
color: #fff;            padding:
20px;            border-radius:
8px;
          box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        }
h1 {
          text-align: center;
```

```
        }
form {
display:
flex;
flex-
direction:
column;
        }
label {
            margin-top: 10px;
        }
        input[type="text"]         {
padding: 10px;           margin-
top: 5px;              border: 1px
solid #ccc;
            border-radius: 4px;
        }
        input[type="submit"]           {
margin-top: 20px;            padding:
10px;                background-color:
#007bff;
            color:             #fff;
border: none;         border-
radius: 4px;           cursor:
pointer;
        }
        input[type="submit"]:hover {
            background-color: #0056b3;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Deadlift Prediction</h1>
        <form action="{{ url_for('predict') }}" method="post">
            <label for="playerId">Player ID:</label>
            <input type="text" id="playerId" name="playerId" required>

            <label for="Sex">Sex:</label>
            <input type="text" id="Sex" name="Sex" required>

            <label for="Equipment">Equipment:</label>
            <input type="text" id="Equipment" name="Equipment" required>

            <label for="Age">Age:</label>
            <input type="text" id="Age" name="Age" required>

            <label for="BodyweightKg">Body Weight (Kg):</label>
```

```
        <input type="text" id="BodyweightKg" name="BodyweightKg" required>

        <label for="BestSquatKg">Best Squat (Kg):</label>
        <input type="text" id="BestSquatKg" name="BestSquatKg" required>

        <label for="BestBenchKg">Best Bench (Kg):</label>
        <input type="text" id="BestBenchKg" name="BestBenchKg" required>            <input
type="submit" value="Predict">
      </form>
    </div>
</body>
</html>
```

PREDICT.HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Prediction Result</title>
    <style>
body {
        font-family: Arial, sans-serif;
background-color: #f2f2f2;          padding:
20px;
      }
      .container {          max-
width: 600px;          margin:
0 auto;          background-
color: #fff;          padding:
20px;          border-radius:
8px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
      }
h1 {
        text-align: center;
      }
      .prediction-text          {
margin-top: 20px;          padding:
10px;          background-color:
#e0ffe0;          border: 1px solid
#00b300;
        border-radius: 4px;
      }
a {
        display:          block;
margin-top: 20px;          text-align:
center;          padding: 10px;
background-color: #007bff;
```

```
            color: #fff;                   text-
 decoration: none;
            border-radius: 4px;
        }
 a:hover {
            background-color: #0056b3;
        }
    </style>
 </head>
 <body>
    <div class="container">
        <h1>Prediction Result</h1>
        <div class="prediction-text">
            {{ prediction_text }}
        </div>
        <a href="{{ url_for('home') }}">Go Back Home</a>
        <a   href="{{   url_for('predict')   }}">Make   Another   Prediction</a>
 </div>
 </body>
 </html>
```

APP.PY

```
import pandas as pd
import numpy as np
import xgboost
import pickle
import os
from flask import Flask, render_template, url_for, request

app = Flask(__name__)
model = pickle.load(open(r'model.pkl', 'rb'))

@app.route('/')  def  home():       return
render_template('home.html')

@app.route('/predict', methods=["POST", "GET"])
def predict():     if request.method == "POST":
       input_feature     =     [float(x)     for     x     in     request.form.values()]
features_values = [np.array(input_feature)]
       names  =  ['playerId',  'Sex',  'Equipment',  'Age',  'BodyweightKg',  'BestSquatKg',  'BestBenchKg']
data = pd.DataFrame(features_values, columns=names)
       prediction = model.predict(data)          text =
"Estimated Deadlift for the builder is: "
       return    render_template("result.html",    prediction_text=text    +    str(prediction))
return render_template("index.html")

if __name__ == '__main__':
   app.run(debug=True)
```

CODE SNIPPETS

MODEL BUILDING

```
import pandas as pd
import numpy as np
```

```
data1=pd.read_csv("/content/X_train.csv",header='infer')
```

```
data2=pd.read_csv("/content/y_train.csv",header='infer')
```

```
data1.head()
```

	playerId	Name	Sex	Equipment	Age	BodyweightKg	BestSquatKg	BestDeadliftKg
0	19391.0	Carlos Ceron	M	Raw	23.0	87.30	205.0	235.0
1	15978.0	Tito Herrera	M	Wraps	23.0	73.48	220.0	260.0
2	27209.0	Levi Lehman	M	Raw	26.0	112.40	142.5	220.0
3	27496.0	Stacy Hayford	F	Raw	35.0	59.42	95.0	102.5
4	20293.0	Brittany Hirt	F	Raw	26.5	61.40	105.0	127.5

```
data1.drop(columns=['Sex'])
```

	playerId	Name	Equipment	Age	BodyweightKg	BestSquatKg	BestDeadliftKg
0	19391.0	Carlos Ceron	Raw	23.0	87.30	205.0	235.00
1	15978.0	Tito Herrera	Wraps	23.0	73.48	220.0	260.00
2	27209.0	Levi Lehman	Raw	26.0	112.40	142.5	220.00
3	27496.0	Stacy Hayford	Raw	35.0	59.42	95.0	102.50
4	20293.0	Brittany Hirt	Raw	26.5	61.40	105.0	127.50
...	...	...	...	...	...	...	...
18895	29802.0	Rachael Podolak	Raw	20.0	65.70	92.5	135.00
18896	5390.0	Daniel Andersen	Raw	23.5	88.85	190.0	230.00
18897	860.0	Mark Brewer	Multi-ply	43.0	106.78	206.38	229.06
18898	15795.0	Michael Perkins	Raw	22.5	82.00	235.0	272.50
18899	23654.0	Art Reyna	Single-ply	15.0	59.42	130.0	140.00

18900 rows × 7 columns

```
[ ] data2.head()
```

	playerId	BestBenchKg
0	19391.0	125.0
1	15978.0	157.5
2	27209.0	145.0
3	27496.0	60.0
4	20293.0	60.0

```
[ ] data=data1.merge(data2,on=['playerId'],how='inner')
```

```
[ ] data.head()
```

	playerId	Name	Sex	Equipment	Age	BodyweightKg	BestSquatKg	BestDeadliftKg	BestBenchKg
0	19391.0	Carlos Ceron	M	Raw	23.0	87.30	205.0	235.0	125.0
1	15978.0	Tito Herrera	M	Wraps	23.0	73.48	220.0	260.0	157.5
2	27209.0	Levi Lehman	M	Raw	26.0	112.40	142.5	220.0	145.0
3	27496.0	Stacy Hayford	F	Raw	35.0	59.42	95.0	102.5	60.0
4	20293.0	Brittany Hirt	F	Raw	26.5	61.40	105.0	127.5	60.0

```
[ ] data.describe()
```

	playerId	Age	BodyweightKg	BestDeadliftKg	BestBenchKg
count	18900.00000	18725.00000	18900.000000	18900.00000	18900.000000
mean	15039.49963	29.66470	85.425557	201.12277	116.963389
std	8674.67268	11.55708	22.959720	62.17163	51.231651
min	0.00000	7.00000	26.130000	18.10000	9.100000
25%	7462.75000	21.50000	67.700000	149.85750	72.500000
50%	15122.50000	26.50000	82.100000	204.12000	115.000000
75%	22540.25000	35.00000	98.970000	247.50000	150.000000
max	29998.00000	83.00000	201.000000	408.23000	425.000000

```
[ ] from sklearn.preprocessing import LabelEncoder
    data['Equipment']= LabelEncoder().fit_transform(data['Equipment'])
```

```
[ ] # Check the column names in your DataFrame
    print(data.columns)
```

```
Index(['playerId', 'Name', 'Sex', 'Equipment', 'Age', 'BodyweightKg',
       'BestSquatKg', 'BestDeadliftKg', 'BestBenchKg'],
      dtype='object')
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18900 entries, 0 to 18899
Data columns (total 9 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   playerId        18900 non-null  float64
 1   Name            18900 non-null  object
 2   Sex             18900 non-null  object
 3   Equipment       18900 non-null  int64
 4   Age             18725 non-null  float64
 5   BodyweightKg    18900 non-null  float64
 6   BestSquatKg     18900 non-null  object
 7   BestDeadliftKg  18900 non-null  float64
 8   BestBenchKg     18900 non-null  float64
dtypes: float64(5), int64(1), object(3)
memory usage: 1.3+ MB
```

+ Code + Text

```
[ ] data1['Name']= LabelEncoder().fit_transform(data1['Name'])
```

```
[ ] data['Age'].fillna(data['Age'].mean(),inplace=True)
```

```
[ ] data.isnull().sum()
```

```
playerId          0
Name              0
Sex               0
Equipment         0
Age               0
BodyweightKg      0
BestSquatKg       0
BestDeadliftKg    0
BestBenchKg       0
dtype: int64
```

```
[ ] data['Sex']= data['Sex'].map({'M':1,'F':0})
    from sklearn.preprocessing import LabelEncoder
    data['Equipment']= LabelEncoder().fit_transform(data['Equipment'])
```

```
data['BestSquatKg']=LabelEncoder().fit_transform(data['BestSquatKg'])
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18900 entries, 0 to 18899
Data columns (total 9 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   playerId        18900 non-null  float64
 1   Name            18900 non-null  object
 2   Sex             18900 non-null  int64
 3   Equipment       18900 non-null  int64
 4   Age             18900 non-null  float64
 5   BodyweightKg    18900 non-null  float64
 6   BestSquatKg     18900 non-null  int64
 7   BestDeadliftKg  18900 non-null  float64
 8   BestBenchKg     18900 non-null  float64
dtypes: float64(5), int64(3), object(1)
memory usage: 1.3+ MB
```

```
[ ] data.shape
```

```
(18900, 9)
```

```
numerical_data = data.select_dtypes(include=['number'])
```

```
Cor = numerical_data.corr()
Cor
```

	playerId	Sex	Equipment	Age	BodyweightKg	BestSquatKg	BestDeadliftKg	BestBenchKg
playerId	1.000000	0.001251	0.010193	0.006190	0.005322	0.002332	0.007222	0.002759
Sex	0.001251	1.000000	0.060221	-0.038825	0.487996	0.001777	0.711668	0.685652
Equipment	0.010193	0.060221	1.000000	0.042759	0.109411	0.045799	0.126675	0.134533
Age	0.006190	-0.038825	0.042759	1.000000	0.110192	0.063723	-0.030556	0.036950
BodyweightKg	0.005322	0.487996	0.109411	0.110192	1.000000	0.122680	0.636692	0.658753
BestSquatKg	0.002332	0.001777	0.045799	0.063723	0.122680	1.000000	0.110069	0.174180
BestDeadliftKg	0.007222	0.711668	0.126675	-0.030556	0.636692	0.110069	1.000000	0.874053
BestBenchKg	0.002759	0.685652	0.134533	0.036950	0.658753	0.174180	0.874053	1.000000

```
data.drop(columns=['Name'],axis=1,inplace=True)
```

```
y = data['BestDeadliftKg']
x = data.drop(columns=['BestDeadliftKg'],axis=1)
```

```
x
```

	playerId	Sex	Equipment	Age	BodyweightKg	BestSquatKg	BestBenchKg
0	19391.0	1	1	23.0	87.30	268	125.00
1	15978.0	1	3	23.0	73.48	297	157.50
2	27209.0	1	1	26.0	112.40	140	145.00
3	27496.0	0	1	35.0	59.42	616	60.00
4	20293.0	0	1	26.5	61.40	68	60.00
...	...	...	...	...	...	...	...
18895	29802.0	0	1	20.0	65.70	613	55.00
18896	5390.0	1	1	23.5	88.85	235	125.00
18897	860.0	1	0	43.0	106.78	270	151.95
18898	15795.0	1	1	22.5	82.00	324	135.00
18899	23654.0	1	2	15.0	59.42	115	80.00

18900 rows × 7 columns

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
```

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3,random_state=0)
```

```
print(x_train.shape)
print(x_test.shape)
```

```
(13230, 7)
(5670, 7)
```

```
print(y_train.shape)
print(y_test.shape)
```

```
(13230,)
(5670,)
```

```
print(y_train.shape)
print(y_test.shape)
```

```
(13230,)
(5670,)
```

```
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
import xgboost as xgb
```

```
x_train
```

	playerId	Sex	Equipment	Age	BodyweightKg	BestSquatKg	BestBenchKg
6335	7623.0	0	1	37.0	107.37	168	72.5
844	25912.0	1	3	26.0	130.00	518	200.0
2421	23278.0	1	1	28.0	127.20	240	155.0
17006	29880.0	1	1	22.5	82.43	310	150.0
1875	13172.0	1	1	20.5	117.77	438	202.5
...	...	...	...	...	...	...	...
9225	20516.0	1	1	30.0	109.72	351	202.5
13123	23596.0	1	1	21.5	92.30	298	130.0
9845	18812.0	0	1	28.0	84.91	168	77.5
10799	16195.0	1	1	20.5	81.60	203	102.5
2732	28654.0	0	0	37.0	74.75	168	102.5

13230 rows × 7 columns

```
[ ] y_train
```

```
6335     177.5
844      352.5
2421     210.0
17006    262.5
1875     310.0
         ...
9225     227.5
13123    257.5
9845     165.5
10799    217.5
2732     145.0
Name: BestDeadliftKg, Length: 13230, dtype: float64
```

```
y_train.info()
```

```
<class 'pandas.core.series.Series'>
Index: 13230 entries, 6335 to 2732
Series name: BestDeadliftKg
Non-Null Count  Dtype
--------------  -----
13230 non-null  float64
dtypes: float64(1)
memory usage: 206.7 KB
```

```
[ ] lr = LinearRegression()
    lr.fit(x_train,y_train)
    y_pred1 = lr.predict(x_test)
```

```
[ ] mse = mean_squared_error(y_test,y_pred1)
    print("Mean Squared Error:", mse)
    rmse = np.sqrt(mse)
```

```
Mean Squared Error: 776.9743522396445
```

```
[ ] from sklearn.metrics import r2_score
    r2_score(y_test,y_pred1)
```

```
0.7995753541725656
```

```
[ ] from sklearn.ensemble import RandomForestRegressor
    rf = RandomForestRegressor()
    rf.fit(x_train,y_train)
    y_pred2 = rf.predict(x_test)
```

```
[ ] mse= mean_squared_error(y_test,y_pred2)
    print("Mean Squared Error:", mse)
    rmse = np.sqrt(mse)
```

```
Mean Squared Error: 474.87946163577783
```

```
[ ] print("RMSE value:(:.2f)",format(rmse))
```

```
RMSE value:(:.2f) 21.791729202515754
```

```
[ ] print("Training Accuracy for RandomForest: {:.2f}",format(rf.score(x_train,y_train)*100),'%')
    print("Testing Accuracy for RandomForest: {:.2f}",format(rf.score(x_test,y_test)*100),'%')
```

```
Training Accuracy for RandomForest: {:.2f} 98.32775209880883 %
Testing Accuracy for RandomForest: {:.2f} 87.75023298584796 %
```

```
from sklearn.tree import DecisionTreeRegressor
dt = DecisionTreeRegressor()
dt.fit(x_train,y_train)
y_pred3 = dt.predict(x_test)
```

```
mse= mean_squared_error(y_test,y_pred3)
print("Mean Squared Error:", mse)
rmse = np.sqrt(mse)
```

```
Mean Squared Error: 890.4273617636685
```

```
print("RMSE value:(:.2f)",format(rmse))
```

```
RMSE value:(:.2f) 29.840029520154108
```

+ Code + Text

```
print("Training Accuracy for DecisionTree: {:.2f}",format(dt.score(x_train,y_train)*100),'%')
print("Testing Accuracy for DecisionTree: {:.2f}",format(dt.score(x_test,y_test)*100),'%')
```

```
Training Accuracy for DecisionTree: {:.2f} 100.0 %
Testing Accuracy for DecisionTree: {:.2f} 77.03095499843528 %
```

```
import xgboost as xgb
xgb_model = xgb.XGBRegressor()
xgb_model.fit(x_train,y_train)
y_pred4 = xgb_model.predict(x_test)
```

```
mse= mean_squared_error(y_test,y_pred4)
print("Mean Squared Error:", mse)
rmse = np.sqrt(mse)
```

```
Mean Squared Error: 475.85359822192135
```

```
print("RMSE value:(:.2f)",format(rmse))
```

```
RMSE value:(:.2f) 21.814068813999864
```

```
print("Training Accuracy for XGBoost: {:.2f}",format(xgb_model.score(x_train,y_train)*100),'%')
```

```
Training Accuracy for XGBoost: {:.2f} 94.51461911481135 %
```

```
print("Testing Accuracy for XGBoost: {:.2f}",format(xgb_model.score(x_test,y_test)*100),'%')
```

```
Testing Accuracy for XGBoost: {:.2f} 87.72510461710547 %
```

```
from prettytable import PrettyTable
tb=PrettyTable()
tb.field_names=["Model","RMSE","Training Accuracy","Testing Accuracy"]
tb.add_row(["Linear Regression",rmse,lr.score(x_train,y_train)*100,lr.score(x_test,y_test)*100])
tb.add_row(["Random Forest",rmse,rf.score(x_train,y_train)*100,rf.score(x_test,y_test)*100])
tb.add_row(["Decision Tree",rmse,dt.score(x_train,y_train)*100,dt.score(x_test,y_test)*100])
tb.add_row(["XGBoost",rmse,xgb_model.score(x_train,y_train)*100,xgb_model.score(x_test,y_test)*100])
```

```
print(tb)
```

```
+-------------------+-------------------+-------------------+-------------------+
|       Model       |        RMSE       | Training Accuracy |  Testing Accuracy |
+-------------------+-------------------+-------------------+-------------------+
| Linear Regression | 21.814068813999864 | 79.55925360084694 | 79.95753541725657 |
|   Random Forest   | 21.814068813999864 | 98.32775209880883 | 87.75023298584796 |
|   Decision Tree   | 21.814068813999864 |       100.0       | 77.03095499843528 |
|      XGBoost      | 21.814068813999864 | 94.51461911481135 | 87.72510461710547 |
+-------------------+-------------------+-------------------+-------------------+
```

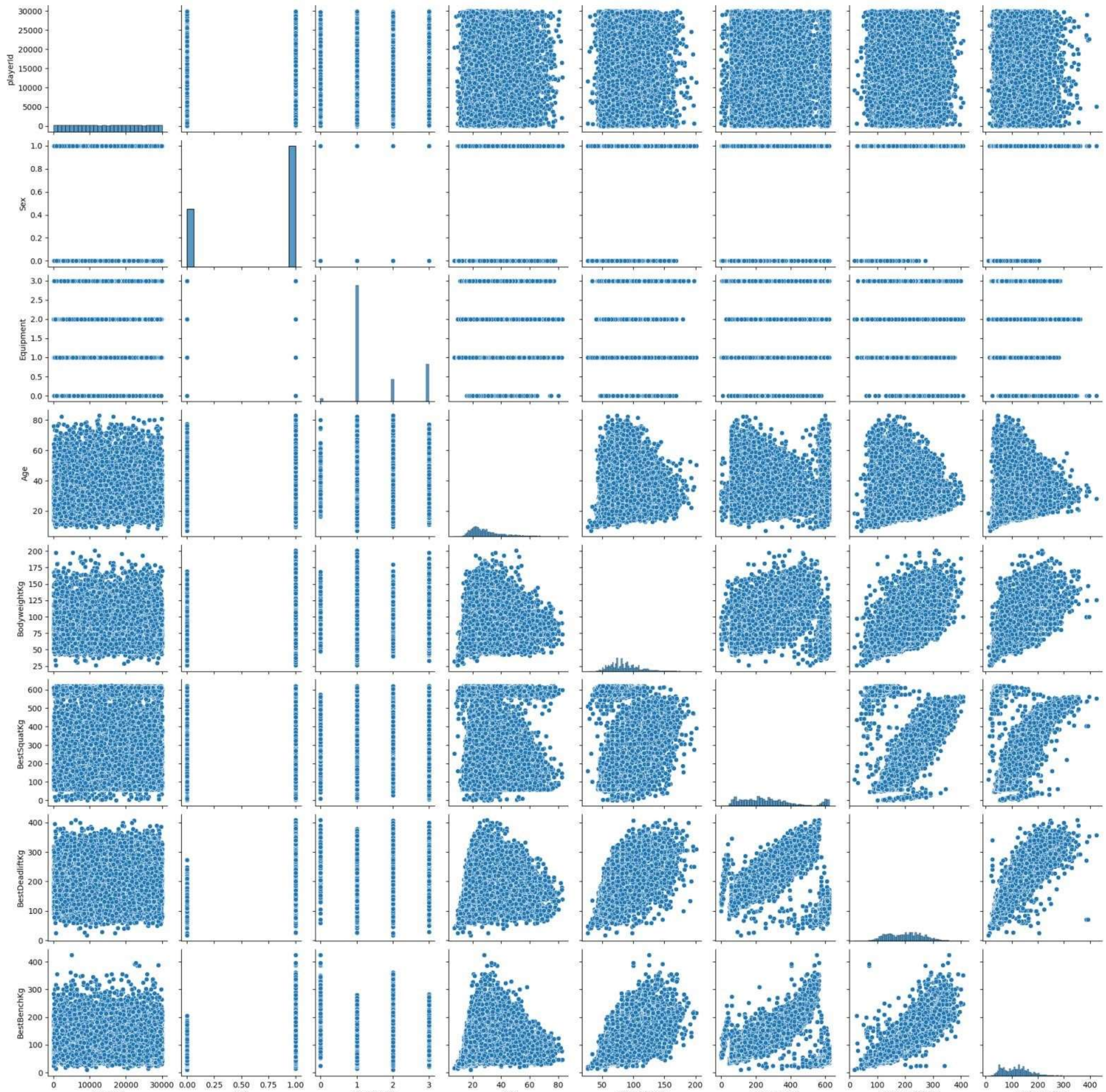
```
from sklearn.model_selection import cross_val_score
cv=cross_val_score(rf,x,y,cv=5)
np.mean(cv)
```

```
0.8801222377302913
```

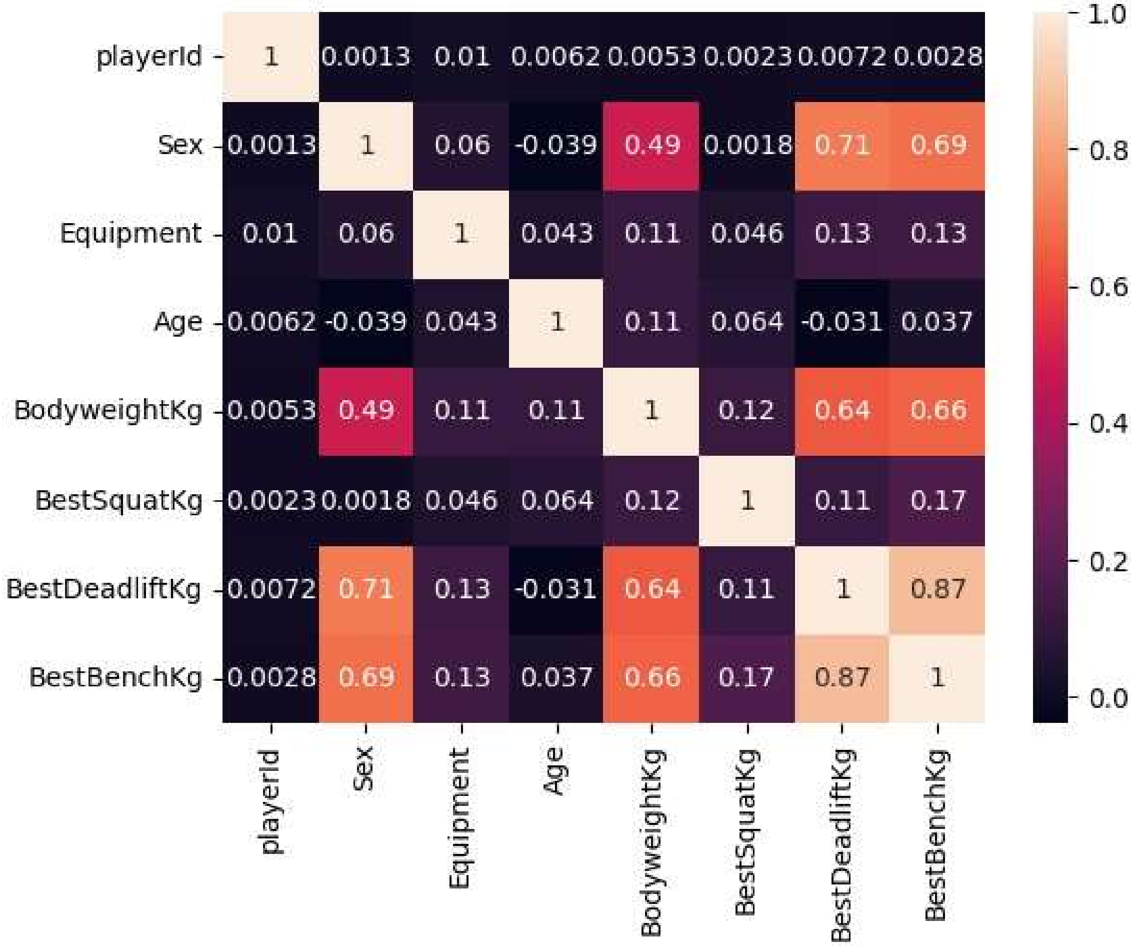
```
import pickle
pickle.dump(rf,open('model.pkl','wb'))
```

```
Start coding or generate with AI.
```

PAIRPLOT:


playerId
Sex
Equipment
Age
BodyweightKg
BestSquatKg
BestDeadliftKg
BestBenchKg

HEATMAP:



playerId
Sex
Equipment
Age
BodyweightKg
BestSquatKg
BestDeadliftKg
BestBenchKg
1 0.0013 0.01 0.0062 0.0053 0.0023 0.0072 0.0028
0.0013 1 0.06 -0.039 0.49 0.0018 0.71 0.69
0.01 0.06 1 0.043 0.11 0.046 0.13 0.13
0.0062 -0.039 0.043 1 0.11 0.064 -0.031 0.037
0.0053 0.49 0.11 0.11 1 0.12 0.64 0.66
0.0023 0.0018 0.046 0.064 0.12 1 0.11 0.17
0.0072 0.71 0.13 -0.031 0.64 0.11 1 0.87
0.0028 0.69 0.13 0.037 0.66 0.17 0.87 1
1.0
0.8
0.6
0.4
0.2
0.0