IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Predictive Analysis of position specific nutritional requirements in youth players using machine learning and data science techniques

Aditya Nema

Excellent Bio Research Solutions Pvt. Ltd.

1042, Napier Town, 4th Bridge, Jabalpur 482001 India

Abstract

Aspiring young soccer players have a unique set of nutritional requirements owing to the sport being a high energy expenditure nature game as well as accelerated growth of players at this age. Soccer is still an emerging sport in India, and oftentimes those who belong to economically weaker sections of society do not have accessibility to professional guidance for appropriate nutrition management. However, with the improvement in technology, along with increased accessibility and affordability of internet in most parts of the country it is possible to make this nutritional knowledge easily accessible to young players through the internet. Recent advancement in machine learning and data science technology has great potential to bridge this knowledge gap. The purpose of this study is to assess the feasibility of using machine learning and data science techniques in predicting nutritional requirements of young players customized to their playing position and to equip them with information regarding their caloric requirements from each source. The predictive capabilities of machine learning can be further used to empower players by identifying their nutritional deficiencies and recommending supplements or diet plans for their recovery as per their playing positions and metabolic requirements.

Keywords: Soccer, Adolescent players, Machine Learning, Data Science Technology, Nutritional requirements, diet plans, playing position

The game of soccer has been played in India since a long time, however, it has been gaining popularity at an accelerated rate among youths since the past decade, predominantly in the eastern and southern regions of the country. Despite this, nutritional expertise in these regions remains concentrated in big cities, resulting in youth and amateur players in smaller towns and villages with limited knowledge regarding their nutritional requirements, adversely affecting their development opportunities. Most of the young players are either in their preteen and teenage years, during which they require a distinct set of nutrient requirements. Soccer also demands a high level of physical fitness specific to the playing positions of the players which makes high caliber nutritional knowledge one of the most important factors in shaping these youths' future careers. Therefore, incorporating data driven technology in the field of nutrition can immeasurably enhance players' fitness levels.

There has been exponential growth in the field of digital technology in India, which had been further fueled by the COVID-19 pandemic in last two years. This rapid proliferation of technology can be tapped into to deliver knowledge resources to people who are living in remote regions of the country with minimal expenditure. Presently, even secluded regions of the country have internet access - with approximately 700 million users - accounting for around 57 percent of the total Indian population. Moreover, Rural India had 44% more internet users compared to urban India (1). Similarly, the number of aspiring young footballers is following an upward trend in both rural and urban areas. This surge in interest has prompted the establishment of foreign football academies in India that aim to develop grassroots football. Additionally, numerous non-governmental organizations (NGOs) and academies have emerged in rural and remote regions of the country. There is a considerable increase in the participation of young and adolescent girls in this sport. For instance, in the Anantapur district (Andhra Pradesh, India) alone, approximately 3,000 boys and girls enroll in an NGO-operated sporting academy each year. Considering their specific nutritional requirements, the availability of accurate nutritional information becomes a crucial factor for assessing the growth and development of these aspiring players (2)

Digital technology proliferation can make relevant nutritional information accessible to rising number of young athletes. This information pertaining to their specific needs can be generated using machine learning technology. This technology has already been applied in other important problem areas in nutrition, such as obesity, metabolic health, and malnutrition **(3).**

In this study, appropriate data science tools are used to process and analyze dietary data of youth players obtained from various sources. By utilizing machine learning algorithms, predictive models to determine the optimal diet plan tailored to nutritional need of players specific to their playing positions has been generated. Machine learning algorithms have the capacity to learn and evolve dynamically to enhance the output, which makes it an excellent gadget to be associated with refining the fitness levels of soccer player. These systems also have capacity to learn from large unstructured data and provide accurate prediction and suggestion, this feature can pave the way for creation of mobile applications and wearable devices which can generate customized player position specific dietary plans or fitness and injury management devices that can monitor player heath and provide real-time advice in various scenarios.

Nutritional requirement in young players

Understanding nutrition is one of the most critical components in training youth players. Nutritional knowledge must not only include the required calories also include the components of healthy diet, timing of intake and continuous adjustment and balance in these factors with changing needs. A well-balanced diet containing appropriate amounts of macronutrients (protein, carbohydrates and fat) and micronutrients (vitamins and minerals) is essential to provide enough energy for growth and activity. Fluids are also essential for hydration to support growth and athletic performance. **(4)**

Energy requirements

The energy needs of young athletes are influenced by factors such as age, body size, body composition and training intensity. Sport-specific expenditure for young athlete has traditionally been evaluated based on physical activity level (PAL) and resting energy expenditure (REE). The estimate energy requirement for children and adolescent are based of energy expenditure, requirement for growth and level of physical activity (5). In the case of academy and youth players, Very Active PAL is considered as the physical activity level.

Table 3.1 Examples of EER (Calories) for Males and Females Ages 10 to 18

	Age	Reference weight (lb/kg)	Reference height (in./cm)	Sedentary PAL[1]	Low active PAL[2]	Active PAL[3]	Very active PAL[4]
Male	10	70/32	55/140	1,605	1,879	2,154	2,492
	12	90/41	58/147	1,793	2,108	2,423	2,810
	14	130/59	65/165	2,316	2,716	3,114	3,601
	16	150/68	69/175	2,527	2,969	3,412	3,956
	18	170/77	71/180	2,692	3,172	3,652	4,243
Female	10	70/32	55/140	1,475	1,734	1,978	2,384
	12	90/41	60/152	1,623	1,916	2,191	2,649
	14	105/48	62/157	1,677	1,987	2,281	2,768
	16	120/54	63/160	1,707	2,033	2,339	2,849
	18	130/59	66/168	1,763	2,108	2,431	2,970

[1]Rare in children.

[2]Less than 1 hour/day of activity.

[3]Approximately 1 hour/day of activity.

[4]More than 1 hour/day of activity.

Data based on calculations from J. Otten, J.P. Hellwig, and L.D. Meyers for Institute of Medicine of the National Academies, 2006, *Dietary Reference Intakes: The essential guide to nutrient requirements* (Washington, DC: National Academies Press), 82.

Source-(5)

Macro and micronutrient requirements

Soccer players need a continuous supply of carbohydrates to meet the high energy demands of the game as well as to maintain and fuel body growth at their age of development. While calculating carbohydrate percentage intake, the desired outcome, whether it is increasing muscle mass, reducing body fat, or improving performance, must be taken into consideration. The type of carbohydrate and timing of consumption is also equally important. For example, simple carbohydrates digest quickly, proving to be a great source when energy is needed quickly such as before an event, during an event and during competition. (5)Although protein is not a primary energy source it is still the most sought after and vital body building macronutrient. Similar to carbohydrates protein requirements are high for young soccer players as the duration and intensity of their exercise is very high. Under these conditions, protein plays an important role in maintaining blood sugar and muscular ability. However, consuming too much protein may results in hampering the performances of player contradictory to helping him. The formula given below can be used to determine the daily range of protein a young athlete would ideally require. (5)

------------weight in Kg*1.2 = ---------gm

------------weight in Kg*1.7 = ----------gm

Thus a 14-year-old athlete with a body weight of 50 kg would need 60gm-85gm of protein. Proteins and carbohydrates must be included in recovery food which must be provided twice -within 30 min of exercise and again within 1 h to 2 h of exercise - to help reload players' muscles with glycogen and provide for a proper recovery (3)

Fat is also important source of energy particularly in young athletes. Fats are required to provide absorbable fat-soluble vitamins (A, D, E, K), essential fatty acids and insulation, and to protect vital organs. (3)**.** 20% to 30% of total energy intake should be in the form of fat for young athletes.

Vitamin and minerals are an essential part of young athletes' diets. However, for young athletes, monitoring of Vitamin D, calcium and Iron intake is crucial. Children up to the age of eight years require 1000mg/day of calcium intake, while nine to eighteen-year-olds need to have around 1300mg/day of calcium intake. Vitamin D is necessary for bone health and is also involved in the absorption and regulation of calcium. In the age bracket of nine to eighteen years 600 IU intake of Vitamin D is recommended. In order to avoid anemia and other iron deficiency related health issues, children between nine to thirteen years of age should have an intake of 8 mg of Iron per day. This requirement increases in the age bracket of fourteen to eighteen years – 11 mg per day for boys and 15 mg per day of iron intake for girls. Monitoring of iron is especially important for female adolescent athletes, especially for vegetarian female athletes.

Diet planning and nutrition management

There are several contributing factors which influence the food choices of athletes such as prevalence of certain food types, social economical background and knowledge of nutritional information. Inaccessibility of information is the evil which significantly affects the diet plans of young athletes especially in countries where soccer is not prioritized and does not provide resources pertaining to the former. In India, a large number of sports training centers for adolescent athletes are not equipped with nutritionists. According to a study conducted in a sports school located in a southern Indian state, when young athletes turned to their coaches for nutrition information and dietary suggestions during training and competitions, coaches had the opinion that diets must be regulated based on "training intensity" or "energy expended". However, on probing, they were unable to elaborate on the specifications about number of calories to be provided through diet. (6)

Constant dearth of relevant and timely nutritional awareness often leads to huge discrepancies in dietary intake of players, especially those who belong to socially or economically backward backgrounds. As per a comparative study conducted on adolescent player under sports authority of India, the Dietary Nutrition Index (DNI) of athletes is significantly lower than the recommended level. They were also found to be deficient in micronutrients, especially girls, who were found to be iron deficient. There was a significant DNI difference in urban and tribal athletes too, with the latter being comparatively deficient in micro and macro nutrients. (7)

	SAI BOYS					SAI GIRLS				
Nutrient	Observed	Mean	Difference	SD	DNI	Observed	Mean	Difference	SD	DNI
	Mean	RDA	Mean			Mean	RDA	Mean		
Protein (gm)	102.6	130.5	-27.8	37.4	-0.7	63.5	102.3	-38.8	17.2	-2.2
Fat (gm)	83.9	49.7	34.8	31.8	1.09	60	40.1	19.9	22.2	0.9
Carbohydrate (gm)	449.8	522.2	-72.3	101.2	-0.7	364.5	409.4	-44.8	80.8	-0.5
Calorie (Kcal)	2940.9	3297.3	-356.4	706.8	-0.5	2259	2652.5	-393.5	514.8	-0.7
Calcium (mg)	1194.8	800	394.8	910.3	0.4	480.5	799.9	-319.4	156.1	-2
Iron (mg)	24.2	11	13.22	9.2	1.4	15.98	11.08	4.9	6.3	0.7
Vitamin A (µg)	1041.5	600	441.5	1147.2	0.3	756.7	600	156.7	667.8	0.2
Vitamin D retinol (µg)	359.1	399.9	-40.8	356.3	-0.11	110.3	399.9	-289.6	69.2	-4.1
Vitamin D Carotene(µg)	52.4	15.08	37.4	34.9	1.11	27.63	15.03	12.6	15.2	0.83
Vitamin B1 (mg)	2.11	1.4	0.63	0.7	0.9	1.29	1.1	0.1	0.4	0.4
Vitamin B2 (mg)	1.9	1.76	0.14	1.2	0.11	0.91	0.51	0.4	0.4	1
Vitamin C (mg)	47.6	40	7.6	26.6	0.2	45.3	40	5.3	19.4	0.2

Table 2: Difference Between Observed Mean Intake and RDA of Different Nutrients in SAI Boys and Girls.

Source- (7)

In such scenarios, proliferation of digital technological advancement can be used to deliver, manage and monitor dietary intake for young athletes. Machine learning is one such technology which can bridge the gap for athletes and provide them accessibility to timely dietary advises and monitoring. It can be applied on the given data provided by individual athletes or sports schools and be used to analyze and manage nutrition knowledge in a cost effective way.

Data Analysis and machine Learning tools

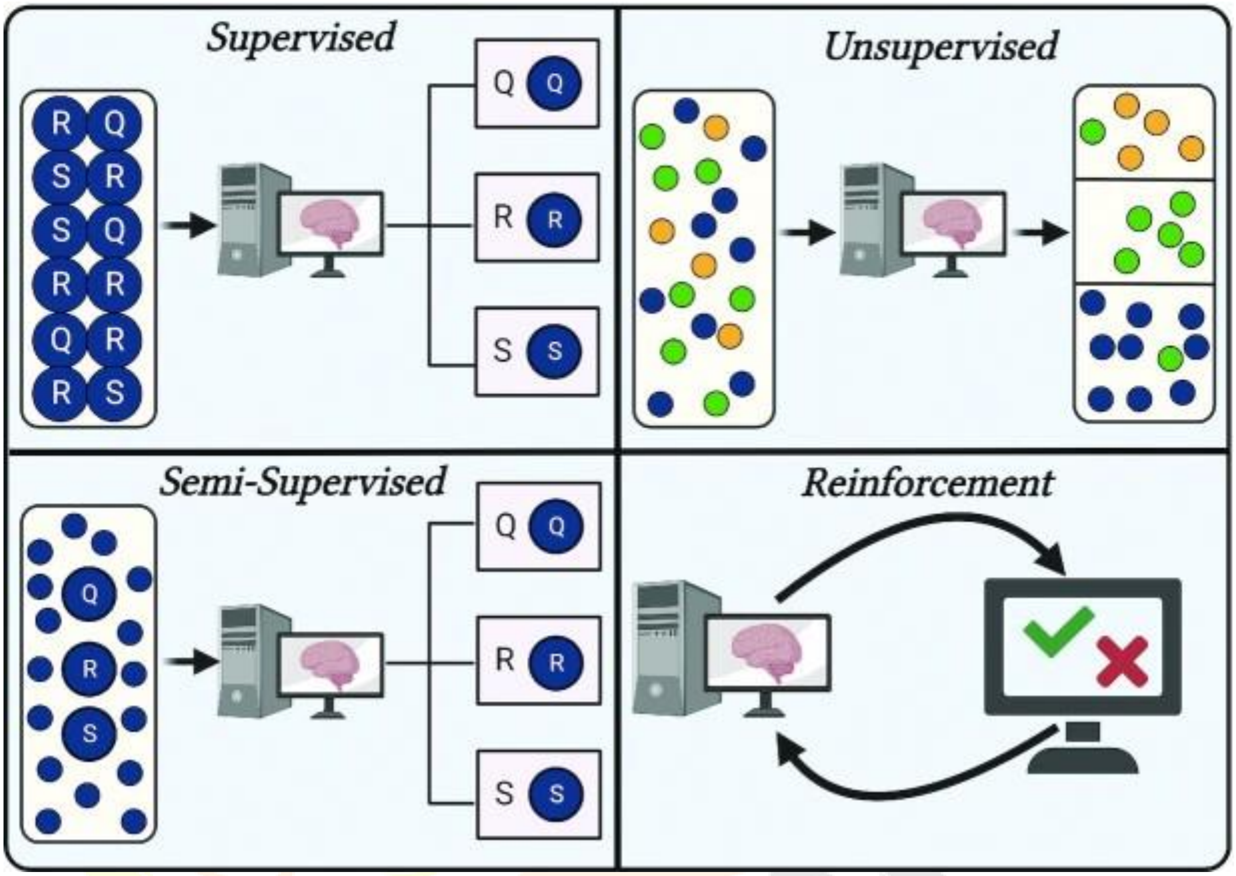
Machine learning is a realm of Artificial intelligence that makes use of large-scale data and various algorithms to provide predictive conclusions. These algorithms have advanced capabilities such as self-learning and improvising outputs without further reprograming. Data science uses statistical tools and models to transform large-scale unorganized data into meaningful and ready-to-use data for machine learning algorithms.

Machine learning and data science algorithms are presently being applied both in professional soccer and other sports for game analytics and predictions. They are also being used for monitoring player performance and injury prevention. Machine learning has already been applied in several aspects related to nutrition - such as childhood obesity prediction, malnutrition detection and epidemiology **(3).** The ability of machine learning algorithms pertaining to large-scale data can be utilized to empower young athletes with timely and accessible nutritional knowledge.

Machine learning algorithms

Machine learning algorithms are programs or patterns which can learn and generate outcomes on the basis of the provided data. According to the types of data provided, the algorithms have been classified as :

- Supervised machine learning
- Unsupervised machine learning
- Semi supervised learning
- Reinforcement



Source of image(3)

Supervised learning

Supervised learning uses labelled datasets. Labelled datasets are a type of data in which the characteristics of data are tagged and can be mapped with respect to the output. Algorithms such as linear regression, logical regression, support vector machines and decision tree are used in this type of training for generation of predictions, comparisons, classifications, assessments or forecasting. This kind of learning is applied in marketing, finance and healthcare sectors.

Unsupervised learning

Unsupervised learning uses unlabeled datasets which are not categorized with respect to a predefined output. The algorithms used in this type of learning work independently to discover patterns and insights and provide groups of clustered data**.** A few examples of unsupervised learning are - K means clustering which can be used to identify clustering patterns, Apriori which can discover relationships, association patterns and neural networks are examples of algorithms used in this type of learning.

Semi-supervised learning

Semi supervised learning is a hybrid of supervised and unsupervised learning. It uses a mix of labelled and unlabeled data - using smaller datasets that are labelled to classify and extract information from the larger unlabeled dataset. It uses algorithms such as generative adversarial network that can be used for audio-video manipulations and self - trained naïve Bayes classifier which is used for natural language processing. This type of learning is useful when there is not enough labelled data or acquiring labelled data is costly and time consuming.

Methodology

Based on the research data above, an extrapolated dataset was created containing ages, weights, calories from fats, proteins, carbohydrates, and vitamins/minerals of youth players in the age range of 10-18. This dataset was converted to a ".csv" format file and imported to a Python language using Jupyter Notebook Integrated Development Environment, following which the below steps were ensued.

Step 1- Import modules required and assign dataset to a data frame variable.

Step 2- Capture vital pieces of information through exploratory data analysis.

Step 3- Perform Data preprocessing (re-labelling, replacing and truncating).

Step 4- Plot graphs for better understanding of the data in hand.

Step 5- Select the type of Machine Learning model to be used (Random Forest Regressor) and define it appropriately.

Step 6- Redefine the model by the train-test-split method and cross examine the predictions with the test dataset.

Step 7- Accept user inputs and fit them in the model created

Step 8- Obtain predictions and print as output

The first step in creating the program was importing the required modules from python libraries i.e. NumPy, seaborn and pandas.

1) importing the required libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
```

Next, the dataset was read using the Pandas module and was assigned to a data frame.

2) reading the data and assigning it to a dataframe

```
[4]  df=pd.read_csv("project.csv")
```

Then Exploratory Data analysis was performed which is an essential component of any machine learning program. In this step, commands were entered to describe the dataset and to analyze each of its items

3) EDA- exploratory data analysis

describing the dataset

```
[5] df.describe()
```

	age	weight	height	calories/very active pal	protein	carbs	fat	vitamin minerals
count	50.000000	50.000000	50.000000	50.000000	50.000000	50.0	50.0	0.0
mean	14.000000	56.860000	163.400000	3444.880000	79.604000	0.0	0.0	NaN
std	2.857143	17.283955	16.089038	746.491768	24.197537	0.0	0.0	NaN
min	10.000000	28.000000	136.000000	2311.000000	39.200000	0.0	0.0	NaN
25%	12.000000	41.250000	147.250000	2791.750000	57.750000	0.0	0.0	NaN
50%	14.000000	59.500000	168.500000	3606.000000	83.300000	0.0	0.0	NaN
75%	16.000000	68.750000	175.750000	3943.750000	96.250000	0.0	0.0	NaN
max	18.000000	91.000000	191.000000	4986.000000	127.400000	0.0	0.0	NaN

analyzing the first five elements of the dataframe

```
[6] df.head()
```

	position	age	weight	height	calories/very active pal	protein	carbs	fat	vitamin minerals
0	gk	10	36	147	2325	50.4	0	0	NaN
1	def	10	35	145	2401	49.0	0	0	NaN
2	wb	10	30	139	2560	42.0	0	0	NaN
3	mid	10	32	140	2492	44.8	0	0	NaN
4	fwd	10	28	137	2571	39.2	0	0	NaN

analyzing the items of the dataframe individually

```
[7] df.items
```

```
<bound method DataFrame.items of    position  age  weight  height  calories/very active pal  protein  carbs  \
0        gk   10      36     147                      2325     50.4      0
1       def   10      35     145                      2401     49.0      0
2        wb   10      30     139                      2560     42.0      0
3       mid   10      32     140                      2492     44.8      0
4       fwd   10      28     137                      2571     39.2      0
5        gk   10      37     148                      2311     51.8      0
6       def   10      34     144                      2392     47.6      0
7        wb   10      29     136                      2498     40.6      0
8       mid   10      32     139                      2563     44.8      0
9       fwd   10      32     139                      2684     44.8      0
10       gk   12      53     161                      2601     74.2      0
11      def   12      52     160                      2788     72.8      0
12       wb   12      43     144                      3002     60.2      0
13      mid   12      41     147                      2810     57.4      0
14      fwd   12      43     149                      2981     60.2      0
15       gk   12      48     156                      2607     67.2      0
16      def   12      47     154                      2803     65.8      0
17       wb   12      39     143                      2945     54.6      0
18      mid   12      42     149                      2913     58.8      0
19      fwd   12      39     142                      3147     54.6      0
20       gk   14      67     178                      3194     93.8      0
```

.

The next step was data preprocessing. Firstly, the columns were renamed such that they had shorter new names so that they be could accessed much more easily. Secondly, the protein, carbohydrates and fat caloric null values were replaced using the nutritional values from the research data in this paper, and columns such as vitamins and minerals were truncated since their requirements are almost the same for sportsmen and non-athletes alike.

replacing null values of each row with the nutritional value from research data

```
[9]
for i in range (0,50):
  df["protein"][i]=df["protein"][i]*8
  df["fat"][i]=df['cals'][i]*0.244
  if(df["position"][i])=="gk":
    df["carbs"][i]=df["weight"][i]*(3.9*9)
  elif(df["position"][i])=="def":
    df["carbs"][i]=df["weight"][i]*(4.3*9)
  elif(df["position"][i])=="mid":
    df["carbs"][i]=df["weight"][i]*(5.65*9)
  elif(df["position"][i])=="wb":
    df["carbs"][i]=df["weight"][i]*(5.6*9)
  elif(df["position"][i])=="fwd":
    df["carbs"][i]=df["weight"][i]*(5.7*9)
```

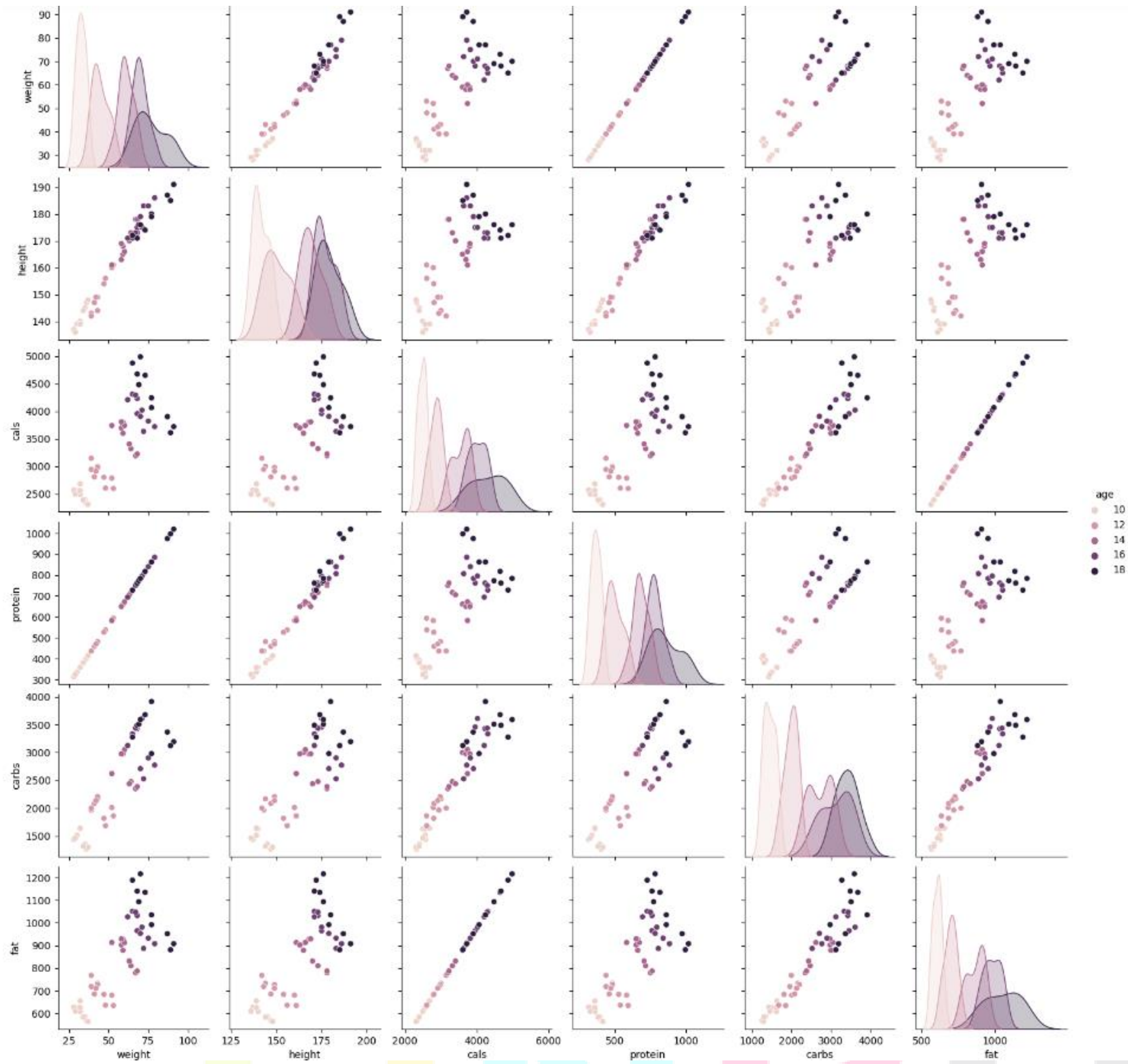
dropping redundant null values

```
[11] df.drop("vitamin minerals",axis=1,inplace=True)
```

```
[26] df.head()
```

	position	age	weight	height	cals	protein	carbs	fat
0	1	10	36	147	2325	403.2	1263.6	567.300
1	2	10	35	145	2401	392.0	1354.5	585.844
2	3	10	30	139	2560	336.0	1512.0	624.640
3	4	10	32	140	2492	358.4	1627.2	608.048
4	5	10	28	137	2571	313.6	1436.4	627.324

After this, a visual correlation consisting of 16 graphs was printed using the data in hand for a better understanding of the processed dataset on which the algorithms will be used.



The next step would be defining the model, however, prior to this the string values (player positions) had to be converted to integer values so that it would become easier for the machine learning model to process the entered data.

6) replacing string values with integer values to make the data easier to process for the machine learning model (data preprocessing part 2)

```
[14] df=df.replace(to_replace="gk",value=1)
     df=df.replace(to_replace="def",value=2)
     df=df.replace(to_replace="wb",value=3)
     df=df.replace(to_replace="mid",value=4)
     df=df.replace(to_replace="fwd",value=5)
```

```
[30] df.head(10)
```

	position	age	weight	height	cals	protein	carbs	fat
0	1	10	36	147	2325	403.2	1263.6	567.300
1	2	10	35	145	2401	392.0	1354.5	585.844
2	3	10	30	139	2560	336.0	1512.0	624.640
3	4	10	32	140	2492	358.4	1627.2	608.048
4	5	10	28	137	2571	313.6	1436.4	627.324
5	1	10	37	148	2311	414.4	1298.7	563.884
6	2	10	34	144	2392	380.8	1315.8	583.648
7	3	10	29	136	2498	324.8	1461.6	609.512
8	4	10	32	139	2563	358.4	1627.2	625.372
9	5	10	32	139	2684	358.4	1641.6	654.896

To define the model, a Random Forest Regressor model was selected as it can handle multiple inputs and give out multiple regression outputs and can provide important feature analysis.

The target values were separated and the model was redefined by the train test split method and the predictions were cross examined with the test dataset.

7) importing libraries for model training

```
[16] from sklearn.ensemble import RandomForestRegressor
     from sklearn.model_selection import train_test_split
```

8) seperating target variables and defining the model

```
[17] x=df[["position","age","weight","height"]]
     y=df[["carbs","protein","fat"]]
```

```
[18] x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=42,test_size=0.25)
     randomforestmodel=RandomForestRegressor(n_estimators=100,random_state=42)
     randomforestmodel.fit(x_train,y_train)
     y_pred=randomforestmodel.predict(x_test)
```

9) examining the predicted values and comparing them with the original dataset

```
[19] y_pred

     array([[1903.437  ,  448.672  ,  673.12036],
            [3381.1875 ,  749.168  , 1041.9776 ],
            [2847.096  ,  834.624  ,  935.70584],
            [3209.634  ,  947.968  ,  946.61264],
            [1642.482  ,  425.488  ,  635.22472],
            [3544.6635 ,  781.872  , 1084.02612],
            [2656.9035 ,  721.168  ,  858.74336],
            [2513.367  ,  766.864  ,  832.61828],
            [3114.9315 ,  752.304  ,  993.27276],
            [1770.408  ,  415.296  ,  658.27784],
            [1818.558  ,  469.28   ,  665.74912],
            [1591.443  ,  349.664  ,  638.68464],
            [2985.9435 ,  726.32   ,  968.78492]])
```

```
[32] y_test.head(3)
```

	carbs	protein	fat
13	2084.85	459.2	685.640
39	3437.10	750.4	1047.492
30	2527.20	806.4	886.940

Now, the final phase has reached where the players' inputs about their position, height, weight and age, are received and the data was processed again and lastly, the data was fitted into the trained model. Consequently, the predictions were obtained (calories required for the player from carbohydrates, proteins and fats) from the model and printed appropriately.

10) recieving the users input (player details)

```
[26] name=input("ENTER YOUR NAME \n")
     new_data=pd.DataFrame({"position":[input("ENTER YOUR POSITION \n('gk' for goalkeeper\n 'def' for defender\n 'mid' for midlfielder\n 'wb' for wing-back\n 'fwd' for forward)\n")],"age":[input("ENTER YOUR AGE (age range 10-18) \n")],"weight":input(
```

```
ENTER YOUR NAME
John Evans
ENTER YOUR POSITION
('gk' for goalkeeper
 'def' for defender
 'mid' for midlfielder
 'wb' for wing-back
 'fwd' for forward)
mid
ENTER YOUR AGE (age range 10-18)
17
ENTER YOUR WEIGHT
76
ENTER YOUR HEIGHT
178
```

11) replacing string values with integer values to make the final data easier to predict for the machine learning model (data processing)

```
[27] new_data=new_data.replace(to_replace="gk",value=1)
     new_data=new_data.replace(to_replace="def",value=2)
     new_data=new_data.replace(to_replace="wb",value=3)
     new_data=new_data.replace(to_replace="mid",value=4)
     new_data=new_data.replace(to_replace="fwd",value=5)
```

12) obtaining predicitons from the trained model and storing the caloric values in distinct variables

```
[28] new_pred=randomforestmodel.predict(new_data)
     prediction=new_pred.tolist()
     carbs=prediction[0][0]
     prot=prediction[0][1]
     fats=prediction[0][2]
```

13) creating a final reference dictionary and printing the predicted results

```
[29] final_prediction_dictionary = {'c':carbs,'p': prot,'f': fats}
```

```
[34] print("Hey",name,", this is how you should distribute your daily caloric intake. All the best!")
     print("Calories required from carbohydrates are : ",(int(final_prediction_dictionary['c'])))
     print("Calories required from proteins are : ",(int(final_prediction_dictionary['p'])))
     print("Calories required from fats are : ",(int(final_prediction_dictionary['f'])))
     print("Total calories required in a day are : ",int(carbs+prot+fats))
```

```
Hey John Evans , this is how you should distribute your daily caloric intake. All the best!
Calories required from carbohydrates are :  3500
Calories required from proteins are :  807
Calories required from fats are :  990
Total calories required in a day are :  5298
```

FINAL PROGRAM OUTPUTS-

Example: Player 1 (John Evans) is a midfielder and needs more calories from carbohydrates than defenders but less than forwards with respect to the other players' nutritional specifications. The variation in nutrition required (ex: more calories from proteins for defenders and goalkeepers, increased number of calories required from carbohydrates for forwards and wing-backs etc.) is accounted for by the program as shown in the position, age, weight and height specific outputs below

```
Hey John Evans , this is how you should distribute your daily caloric intake. All the best!
Calories required from carbohydrates are :  3500
Calories required from proteins are :  807
Calories required from fats are :  990
Total calories required in a day are :  5298
```

10) recieving the users input (player details)

```
[24] name=input("ENTER YOUR NAME \n")
     new_data=pd.DataFrame({"position":[input("ENTER YOUR POSITION \n('gk' for goalkeeper\n 'def' for defender\n 'mid' for mi

     ENTER YOUR NAME
     Andy Atkinson
     ENTER YOUR POSITION
     ('gk' for goalkeeper
      'def' for defender
      'mid' for midlfielder
      'wb' for wing-back
      'fwd' for forward)
     def
     ENTER YOUR AGE (age range 10-18)
     11
     ENTER YOUR WEIGHT
     51
     ENTER YOUR HEIGHT
     162
```

11) replacing string values with integer values to make the final data easier to predict for the machine learning model (data processing)

```
[25] new_data=new_data.replace(to_replace="gk",value=1)
     new_data=new_data.replace(to_replace="def",value=2)
     new_data=new_data.replace(to_replace="wb",value=3)
     new_data=new_data.replace(to_replace="mid",value=4)
     new_data=new_data.replace(to_replace="fwd",value=5)
```

12) obtaining predicitons from the trained model and storing the caloric values in distinct variables

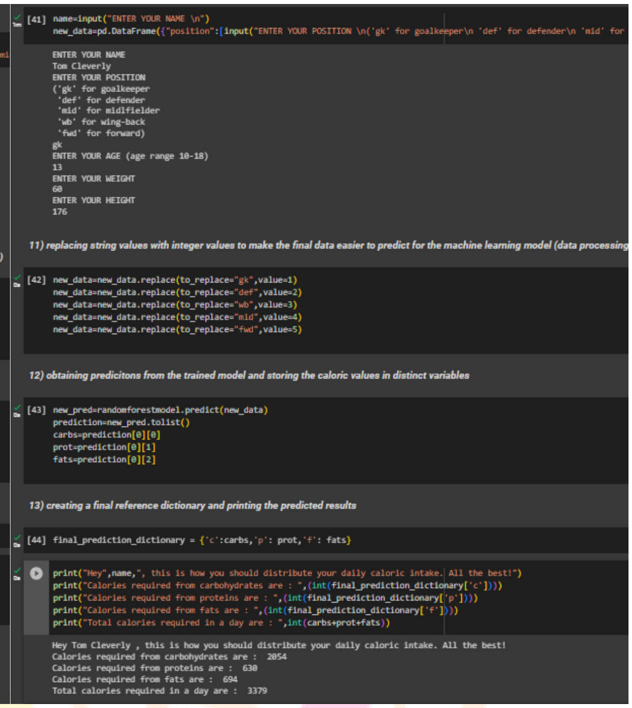
```
[26] new_pred=randomforestmodel.predict(new_data)
     prediction=new_pred.tolist()
     carbs=prediction[0][0]
     prot=prediction[0][1]
     fats=prediction[0][2]
```

13) creating a final reference dictionary and printing the predicted results

```
[27] final_prediction_dictionary = {'c':carbs,'p': prot,'f': fats}

[28] print("Hey",name,", this is how you should distribute your daily caloric intake. All the best!")
     print("Calories required from carbohydrates are : ",(int(final_prediction_dictionary['c'])))
     print("Calories required from proteins are : ",(int(final_prediction_dictionary['p'])))
     print("Calories required from fats are : ",(int(final_prediction_dictionary['f'])))
     print("Total calories required in a day are : ",int(carbs+prot+fats))

     Hey Andy Atkinson , this is how you should distribute your daily caloric intake. All the best!
     Calories required from carbohydrates are :  1781
     Calories required from proteins are :  522
     Calories required from fats are :  652
     Total calories required in a day are :  2955
```



```
[41] name=input("ENTER YOUR NAME \n")
     new_data=pd.DataFrame({"position":[input("ENTER YOUR POSITION \n('gk' for goalkeeper\n 'def' for defender\n 'mid' for

     ENTER YOUR NAME
     Tom Cleverly
     ENTER YOUR POSITION
     ('gk' for goalkeeper
      'def' for defender
      'mid' for midlfielder
      'wb' for wing-back
      'fwd' for forward)
     gk
     ENTER YOUR AGE (age range 10-18)
     13
     ENTER YOUR WEIGHT
     60
     ENTER YOUR HEIGHT
     176
```

11) replacing string values with integer values to make the final data easier to predict for the machine learning model (data processing

```
[42] new_data=new_data.replace(to_replace="gk",value=1)
     new_data=new_data.replace(to_replace="def",value=2)
     new_data=new_data.replace(to_replace="wb",value=3)
     new_data=new_data.replace(to_replace="mid",value=4)
     new_data=new_data.replace(to_replace="fwd",value=5)
```

12) obtaining predicitons from the trained model and storing the caloric values in distinct variables

```
[43] new_pred=randomforestmodel.predict(new_data)
     prediction=new_pred.tolist()
     carbs=prediction[0][0]
     prot=prediction[0][1]
     fats=prediction[0][2]
```

13) creating a final reference dictionary and printing the predicted results

```
[44] final_prediction_dictionary = {'c':carbs,'p': prot,'f': fats}

     print("Hey",name,", this is how you should distribute your daily caloric intake. All the best!")
     print("Calories required from carbohydrates are : ",(int(final_prediction_dictionary['c'])))
     print("Calories required from proteins are : ",(int(final_prediction_dictionary['p'])))
     print("Calories required from fats are : ",(int(final_prediction_dictionary['f'])))
     print("Total calories required in a day are : ",int(carbs+prot+fats))

     Hey Tom Cleverly , this is how you should distribute your daily caloric intake. All the best!
     Calories required from carbohydrates are :  2054
     Calories required from proteins are :  630
     Calories required from fats are :  694
     Total calories required in a day are :  3379
```

10) recieving the users input (player details)

```
     name=input("ENTER YOUR NAME \n")
     new_data=pd.DataFrame({"position":[input("ENTER YOUR POSITION \n('gk' for goalkeeper\n 'def' for defender\n 'mid' for mi

     ENTER YOUR NAME
     Tony Hunter
     ENTER YOUR POSITION
     ('gk' for goalkeeper
      'def' for defender
      'mid' for midlfielder
      'wb' for wing-back
      'fwd' for forward)
     fwd
     ENTER YOUR AGE (age range 10-18)
     18
     ENTER YOUR WEIGHT
     64
     ENTER YOUR HEIGHT
     173
```

11) replacing string values with integer values to make the final data easier to predict for the machine learning model (data processing)

```
[30] new_data=new_data.replace(to_replace="gk",value=1)
     new_data=new_data.replace(to_replace="def",value=2)
     new_data=new_data.replace(to_replace="wb",value=3)
     new_data=new_data.replace(to_replace="mid",value=4)
     new_data=new_data.replace(to_replace="fwd",value=5)
```

12) obtaining predicitons from the trained model and storing the caloric values in distinct variables

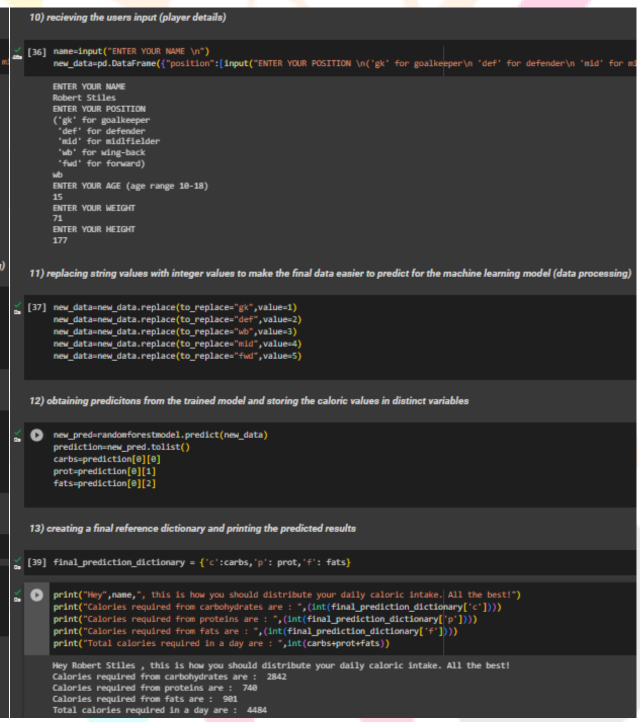
```
[31] new_pred=randomforestmodel.predict(new_data)
     prediction=new_pred.tolist()
     carbs=prediction[0][0]
     prot=prediction[0][1]
     fats=prediction[0][2]
```

13) creating a final reference dictionary and printing the predicted results

```
[32] final_prediction_dictionary = {'c':carbs,'p': prot,'f': fats}

[33] print("Hey",name,", this is how you should distribute your daily caloric intake. All the best!")
     print("Calories required from carbohydrates are : ",(int(final_prediction_dictionary['c'])))
     print("Calories required from proteins are : ",(int(final_prediction_dictionary['p'])))
     print("Calories required from fats are : ",(int(final_prediction_dictionary['f'])))
     print("Total calories required in a day are : ",int(carbs+prot+fats))

     Hey Tony Hunter , this is how you should distribute your daily caloric intake. All the best!
     Calories required from carbohydrates are :  3325
     Calories required from proteins are :  738
     Calories required from fats are :  1107
     Total calories required in a day are :  5170
```



10) recieving the users input (player details)

```
[36] name=input("ENTER YOUR NAME \n")
     new_data=pd.DataFrame({"position":[input("ENTER YOUR POSITION \n('gk' for goalkeeper\n 'def' for defender\n 'mid' for mi

     ENTER YOUR NAME
     Robert Stiles
     ENTER YOUR POSITION
     ('gk' for goalkeeper
      'def' for defender
      'mid' for midlfielder
      'wb' for wing-back
      'fwd' for forward)
     wb
     ENTER YOUR AGE (age range 10-18)
     15
     ENTER YOUR WEIGHT
     71
     ENTER YOUR HEIGHT
     177
```

11) replacing string values with integer values to make the final data easier to predict for the machine learning model (data processing)

```
[37] new_data=new_data.replace(to_replace="gk",value=1)
     new_data=new_data.replace(to_replace="def",value=2)
     new_data=new_data.replace(to_replace="wb",value=3)
     new_data=new_data.replace(to_replace="mid",value=4)
     new_data=new_data.replace(to_replace="fwd",value=5)
```

12) obtaining predicitons from the trained model and storing the caloric values in distinct variables

```
     new_pred=randomforestmodel.predict(new_data)
     prediction=new_pred.tolist()
     carbs=prediction[0][0]
     prot=prediction[0][1]
     fats=prediction[0][2]
```

13) creating a final reference dictionary and printing the predicted results

```
[39] final_prediction_dictionary = {'c':carbs,'p': prot,'f': fats}

     print("Hey",name,", this is how you should distribute your daily caloric intake. All the best!")
     print("Calories required from carbohydrates are : ",(int(final_prediction_dictionary['c'])))
     print("Calories required from proteins are : ",(int(final_prediction_dictionary['p'])))
     print("Calories required from fats are : ",(int(final_prediction_dictionary['f'])))
     print("Total calories required in a day are : ",int(carbs+prot+fats))

     Hey Robert Stiles , this is how you should distribute your daily caloric intake. All the best!
     Calories required from carbohydrates are :  2842
     Calories required from proteins are :  740
     Calories required from fats are :  901
     Total calories required in a day are :  4484
```


Discussion

As in the above outputs, the program takes the player's position, age, weight and height in as inputs and analyzes them to ultimately present the required calories from carbohydrates, proteins and fats. These predictions fluctuate according to position specific parameters most conclusively, then go on to take in account the players BMI and modify themselves. These effective predictions can rectify many soccer players' diets who are in need of guidance and boost their performance efficaciously. This program can also be used by personal coaches to regulate their pupils' diets as well as academies to maintain their youth players' fitness levels. Additionally, as it is free of cost, the program can be used by anyone and everyone, providing fair opportunities to those who are economically or technologically backwards.

Conclusion.

The study provided us evidence that predictive capabilities of machine learning can be extensively applied in the field of soccer players' nutrition requirements and can provide extra insights on nutrition for young soccer players with respect to their specifications. The information thus generated can be accessed by players whenever they want and can be cost effectively delivered through digital applications which can be accessed by WIFI or Bluetooth file sharing.

The algorithms used in the study can further be trained to learn and provide customized dietary plans as per food preferences, injury recovery and other gameplay specific needs. This useful gadget can further be upgraded to be available for access as a mobile application or wearable device to accurately receive real time inputs and predict dietary requirements with pinpoint precision.

References

1.Mar 16, 2023,. Rural India had 44% more internet users compared to urban market:Nielsen report. The economic times.

https://economictimes.indiatimes.com/tech

2.Faizal khan, November 12, 2022 FIFA World Cup 2022: How India's football acdemies are raising the game. *moneycontrol.com*

https://www./news/trends/sports-trends/india-football-academies-raising-the-game-9499581.html

3.Kirk D, Kok E, Tufano M, Tekinerdogan B, Feskens EJM, Camps G. Machine Learning in Nutrition Research. Adv Nutr. 2022 Dec 22;13(6):2573-2589. doi: 10.1093/advances/nmac103. Erratum in: Adv Nutr. 2023 May;14(3):584. Erratum in: Adv Nutr. 2023 Apr 1;: PMID: 36166846; PMCID: PMC9776646.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9776646/#sup1

4. Purcell LK; Canadian Paediatric Society, Paediatric Sports and Exercise Medicine Section. Sport nutrition for young athletes. Paediatr Child Health. 2013 Apr;18(4):200-5. doi: 10.1093/pch/18.4.200. PMID: 24421690; PMCID: PMC3805623.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3805623/

5.Heather Mangieri,*Fueling young athelets (kindle edition)*, Human Kinetics; 1st edition (20 July 2016)

6.Cherian KS, Gavaravarapu SM, Sainoji A, Yagnambhatt VR. Coaches' perceptions about food, appetite, and nutrition of adolescent Indian athletes - A qualitative study. Heliyon. 2020 Feb 7;6(2):e03354. doi: 10.1016/j.heliyon.2020.e03354. PMID: 32072052; PMCID: PMC7013178.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7013178/

7.Singh R, Verma V, Varoda A, Venugopal R. Comparative Study of Dietary Nutrition Index of Players of Sports Authority of India (SAI) and Tribal Schools of Kanker Chhattisgarh India. Curr Res Nutr Food Sci 2021; 9(1). doi : http://dx.doi.org/10.12944/CRNFSJ.9.1.28

Singh R, Verma V, Varoda A, Venugopal R. Comparative Study of Dietary Nutrition Index of Players of Sports Authority of India (SAI) and Tribal Schools of Kanker Chhattisgarh India. Curr Res Nutr Food Sci 2021; 9(1). Available From: https://bit.ly/2LGcVJt