FACULTY OF COMPUTERS AND INFORMATION
SUEZ UNIVERSITY

DEATH RATE PREDICTION BASED ON NUTRITION USING SEVERAL OPTIMIZATION TECHNIQUES

DR. AHMED EL-SHEWY
ASSISTANT PROFESSOR AT FACULTY OF COMPUTERS AND INFORMATION, SUEZ UNIVERSITY, EGYPT
ABDELRAHMAN MOSAED
AHMED ABDELGHANI
STUDENTS AT FACULTY OF COMPUTER AND INFORMATION, SUEZ UNIVERSITY, EGYPT

ARTICLE INFO

Keywords:

COVID-19. Healthy food. Nutrition analysis. Artificial intelligence. Machine learning. Regression. Gradient Boosting regressor “GBR”. Random Forest. Bayesian Ridge. Ridge. Lasso. KNN. Decision Tree. MLP. SVM. Grid Search. Random Search. Optuna.

Abstract

Since 2019, humanity has been suffering from the negative impact of Covid 19, and the virus did not stop in its usual state but began to pivot to become more harmful until it reached its form now, which is the Omicron variant, therefore, in an attempt to reduce the risk of the virus, which has caused nearly 6 million deaths to this day, it is serious to focus on one of the most important causes of disease resistance, which is nutrition, it is proven recently that death rates dangerously based on what enters the human stomach from fat, protein, or even healthy vegetables.

In our research here, several algorithms are applied to the Covid-19 nutrition dataset which makes a relationship between what people eat and the covid-19 death rate.

After using different models, we calculated R^2 , MAE, and MSE to find the best-fitted model, then we used Grid Search ,Random search,and Optuna optimization models techniques to find an optimal solution to apply on the Covid-19 nutrition dataset.

1. Introduction

In numerous nations all through the world, the current (COVID-19) pandemic has prompted general lockdowns that have come about

in the conclusion of everything except fundamental administrations, for example, supermarkets and drugstores. Such terminations have had a quick and unsurprising impact on food attainable quality and choice. The pandemic has confined food determinations, which might affect eating times and diet, as well as generally affecting both physical and mental wellbeing [1].

Food frameworks by implication affect human wellbeing, and presently it is a higher priority than at any other time that they ought to become practical.

In 2015, the United Nations' 2030 Plan for Sustainable Development gave a quick call for activity, including 17 reasonable turn of events objectives, by creating and arising nations in worldwide joint effort [2]. Therefore, indicative apparatuses for food expectation and saving food in general conditions after the lockdown are required. Furthermore, food and industry supply binds should be observed to decide whether they have added to the spread of COVID-19. This is performed by analyzing how COVID-19 spreads through surfaces, the food supply chain, and general conditions [3]. For instance, Mishra and Rampal [4] introduced an investigation of the impact of the COVID-19 pandemic on food instability in India. They began by following the overall status of food uncertainty and craving wherever on the planet, zeroing in on lower and center pay nations. They observed that there are huge connections between financial development, joblessness, and starvation coming about because of food deficiencies during pandemic lockdowns.

These days, ML assumes an essential part in conclusion and anticipation issues particularly following illnesses in a clinical application utilizing picture acknowledgment frameworks, forestalling and treating the spread of explicit illnesses particularly in managing imbalanced information utilizing ML approaches [5-6]. The determination and order of COVID-19 chest X-beam pictures as CT pictures-based ML approaches is the critical element to battle the spread of the Coronavirus infection. Additionally, the expectation framework-based ML is utilized to estimate the impact of the current pandemic in various regions explicitly in diagnosing illnesses and medical services frameworks [7-8]. While buying food, the quantity of decisions is exorbitant to be equipped for thinking of them as all [9]. People have unique dietary needs, propensities, and recognize flavor in various propensities. Thus, the main decision is to understand their necessities by examining the individual. Now and again, the suggestion framework is performed to help an unobtrusive starving purchaser, cooking ally, concerning wellbeing, calorie counter, or someone antagonistic looking to upgrade his/her clinical noticeable quality, which will improve the effect of the last choice. Moreover, the food's presence in time is expected to make the client more separated and more joyful. A huge component while building these frameworks is the information assortment sources and client propensities.

2. Related work

As a general rule, three food patterns rely upon Artificial Intelligence (AI)

that are viewed as while managing food issues. Modern food is monetarily controlled by the phases of production to improve and work with the utilization interaction. In addition, it was acquainted with giving the majority of the food devoured by the total populace. In agribusiness, a significant issue connected with AI is to help ranchers dispose of sicknesses and irritations that influence plants, which thus influences the amount and nature of the harvest, and thus influences the volume of food. We recognized many examinations that assist in distinguishing plant illnesses.

Food is being utilized in the battle against destitution by fostering a suggested Simulated intelligence-based eating regimen to track and screen the nourishing level in agricultural nations. Patients with specific infections, like diabetes, coronary illness, high circulatory strain, and insulin opposition, are generally powerless against COVID-19.

Along these lines, to keep away from that, the patients should screen and diminish the unfortunate quirks of eating particularly food sources with high insulin. Low starch, moderate proteins, and moderate fat are principally expected to keep up with the typical insulin in the patient's body. Food containing Zinc is an effective method for expanding the human insusceptible framework execution. Clams, shellfish, red meat, and cheddar are wealthy in Zinc. Vitamin D too was required and existed in Cod liver oil, and salmon. L-ascorbic acid also is vital to diminish the level of COVID-19 infection presence.

The food plentiful in L-ascorbic acid for example mixed greens, sauerkraut,

furthermore berries [10-11]. For modern food, Shen et al. [12] proposed an application to measure the food attributes to assist with people adjusting their eating routine, as it identifies food things in a picture and remembers them. The application involves the Convolution Neural Network for food acknowledgment. The framework can assess food properties by moving information from the web. They utilized Inception-v3 and Inception-v4 models. These models depend on Convolutional Neural Networks (CNN) and the outcomes acquired to handle the issues are more solid.

Moreover, Onu et al. [13] used AI models to anticipate low dampness content in drying potatoes. They utilized three unique models; the Response Surface Methodology (RSM), Adaptive Neuro-Fuzzy Surmising Systems (ANFIS), and Artificial Neural Network (ANN).

They established that the three models gave great forecasts with the test information yet, RSM and ANFIS gave preferred outcomes over ANN. In food handling, three cases are chosen and studies assembling the machine learning and master communication as introduced in Ref. [14]: In the first, they employed specialists to plan the design of the Bayes dynamic organization for building a camembert developing model, counting factors from small size (presence of microbes and substance parts) to large scale (perceptual evaluations). In the subsequent one, they assembled a model to help winemakers in evaluating when to gather grapes, contingent upon climate conditions, the model is additionally a Bayesian organization model. Third, they utilized a graphical model in view of emblematic relapse to help experts make a model for bacterial creation and adjustment.

A methodology in light of k-bunch division and shading discovery is introduced by Ref. [15] for reviewing, arranging foods grown from the ground, and the extricated highlights are determined like entropy, mean, and standard deviation.

In [16] the analysts produce a framework where they utilized picture handling with the assistance of SVM classifier to order sound rice plants furthermore unhealthy rice plants. The framework achieved a goal of more than 90%.

Besides, in Ref. [17] the specialists present a proposed network structure for ordering potato leaf illnesses in light of CNN. The recommended design is composed of 14 layers, and the normal generally test precision is 98%. In Ref.

[18] additionally recognize leaf illnesses of the apple, they use CNN in light of the pre-train network AlexNet, the tests of the proposed sickness recognizable proof in light of CNN give precision about 97.62%.

3. Background

Covids address a more distant family of respiratory infections that can cause gentle to direct illnesses, from the normal cold to respiratory conditions like MERS (Middle East respiratory disorder) and SARS (Severe intense respiratory condition) [19]. They are supposed as a result of the crown-molded tips that are available on their surface [20]. These sorts of infections are normal in numerous creature species (like camels and bats) yet at times, however once in a while, they can develop and taint people and afterward spread to the populace [1]. A new Covid strain that has never recently been distinguished in people is the one shown up toward the finish of 2019 i.e., the 2019 novel (COVID-19, abbreviation of Coronavirus Disease 19) [21, 3, 16]. The main cases were found during the COVID-19 pandemic of 2019-2020 [22], which likely began around the finish of December 2019 in the city of Wuhan [16], the capital of the Chinese territory of Hubei, and in this manner spread to different nations of the world. Truth be told, as of January 28 2020, there were in excess of 4600 affirmed instances of virus in numerous nations of the world and 106 passes while on February 15 this information had effectively ascended to 49053 cases and 1381 deaths1. As of

January 23 2020, Wuhan was isolated with the suspension of all open vehicles into and out of the city, which measures were stretched out the next day to the adjoining urban communities of Huanggang, Ezhou, Chibi, Jingzhou and Zhijiang. Further limits and controls have been embraced in numerous regions of the world, likewise in Europe where a few cases have additionally been recorded. The nation most impacted in Europe is Italy, where the specialists have attempted to contain a flare-up that has contaminated no less than 400 individuals, the majority of them in northern Italy, close to Milan. As of March. 2, there have been more than 1800 confirmed Covid cases and 30 cases in Italy, with the third biggest number of diseases per country on the planet, after China and South Korea. The COVID-19 disease caused groups of lethal pneumonia with clinical show extraordinarily looking like SARS-CoV. Truth be told, patients experience influenza-like indications, for example, fever, dry hack, sluggishness, trouble relaxing. In more extreme cases, frequently found in subjects previously troubled by past pathologies, pneumonia creates intense renal disappointment, up to even demise [22], however this new Covid presents additionally a few interesting elements [16, 21]. While the analysis is affirmed utilizing polymerase chain response (PCR), contaminated patients with pneumonia might introduce on chest X-beam and figured tomography (CT) pictures an example that is just respectably trademark for the natural eye as illustrated by analysts in [14]. The pace of transmission of COVID-19 relies upon the ability to dependably distinguish contaminated patients with a low pace of misleading negatives. Likewise, a low pace of misleading up-sides is expected to keep away from further expanding the weight on the medical services framework by superfluously presenting patients to isolation on the off chance that that isn't needed. Alongwith legitimate contamination control, it is clear that ideal recognition of the infection would empower the execution of all the strong considerations expected by patients impacted by COVID-19.

Table 1

Performance metrics

Metric	Definition	Equations
MAE: Mean Absolute Error	**The Mean Absolute Error measures the average of the absolute error, i.e. the residuals. The MAE uses the same scale as the data being measured, be careful before making comparisons between series using different scales**	$MAE = \frac{1}{N}\sum_{i=1}^{N} \lvert y_i - \hat{y} \rvert$
R²: Coefficient of determination	**R² or coefficient of determination corresponds to the proportion of the variance explained by our model. It gives an idea of how well our model predicts the data. R² normally ranges from 0 to 1**	$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})}$
MSE: Mean Squared Error	**The Mean Squared Error measures the average of the square of the errors, in other words, it measures the variance of the residuals. The lower the better!**	$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2$
RMSE: Root Mean Squared Error	**The Mean Squared Error measures the average of the square of the errors, in other words, it measures the variance of the residuals. The lower the better!**	$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2}$

Ref [23]

4. Previous studies

Here we're discussing a comparison between two scientific studies, "Impact of COVID-19 Pandemic on Diet Prediction and Patient Health Based on Support Vector Machine" and HANA": A Healthy Artificial Nutrition Analysis model during COVID-19 pandemic".

The same dataset was used in both but applied with different technologies, including the percentage of fat intake from different types of food collected from 170 countries around the world. The last couple of columns also includes counts of obesity, undernourished, and COVID-19 cases as percentages of the total population for comparison purposes,but the researchers only applied to five columns: animal products, cereals excluding beer, obesity, vegetal products, and deaths due to the COVID-19 pandemic [24].

In the first research "Impact of COVID-19 Pandemic on Diet Prediction and Patient Health Based on Support Vector Machine", researchers used the Support Vector Machine (SVM) and DL “deep learning” to predict the effect of the COVID-19 pandemic on a diet and further forecast the number of persons subject to death due to this pandemic, in this model, the dataset is firstly split into training and testing data. The trained data is then scaled based on Log and Z-score scaling to be applied to the input layer to produce ensemble scaled features; this step is called data preparation.

Researchers then used three types of SVM:

- SVM model with RBF Kernel.
- SVM model with Linear kernel.
- SVM model with a poly kennel.

Finally, they calculated only RMSE for each model as shown in table 2

Table 2

RMSE for each model used

The proposed regression model	RMSE
SVM model based on RBF Kernel	0.26958879
SVM model with Linear kernel	**0.177318153**
SVM model with a poly kennel	20.7392
deep learning regression model	0.2828

Due to the lower value with RMSE, SVM model with Linear kernel gives the best result.

In HANA model ": A Healthy Artificial Nutrition Analysis model during COVID-19 pandemic ", researchers used 4 indicators here to evaluate results, MSE, MAE, RMSE and R^2 researchers calculated these indicators from 5 models:

- Ridge Regression
- Simple Linear
- Regularization
- Elastic Net Regression
- AdaBoost

And after applying they found that the most efficient regression prediction model is the elastic net regression. The MSE, MAE, and RMSE for the Elastic Net Regression model were significantly lower than ridge regression, simple linear regularization, and AdaBoost models. On the other hand, the R^2 value was for the Elastic Net Regression model significantly higher than other models.

Table 3 showing a Comparison of the proposed regression prediction models based on evaluation metrics [25].

Model	MSE	RMSE	MAE	R2
Ridge Regression	0.00023083	0.01519314	0.01023939	−0.15965093
Simple Linear Regularization	0.00023091	0.01519604	0.01024034	− 0.16009405
Elastic Net Regression	0.00018113	0.01345867	0.00873109	**0.09001016**
AdaBoost	0.00020749	0.01440446	0.00761746	− 0.04237952

5. Experimental Results: Discussion and Analysis

5.1 Dataset Description and exploratory data analysis

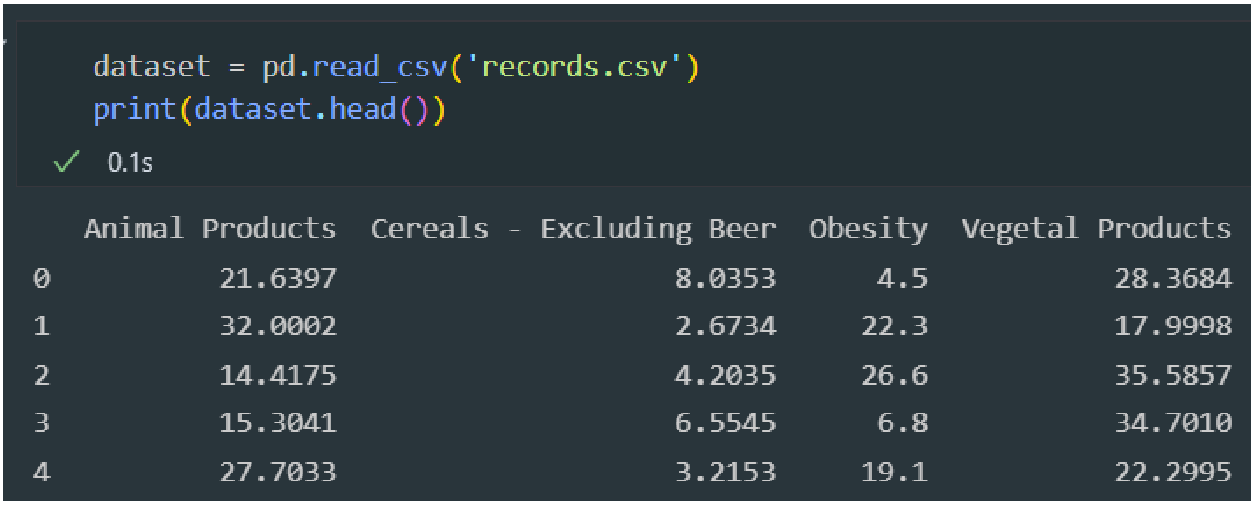
Work was dependent on the Covid-19 nutrition dataset [24], this dataset includes statistics covering 170 different countries, statistics including the percentage of fat for Covid-19 patients, as well as the percentage of obesity and its relationship to corona patients. The studies later explained the negative role of fat percentage and its impact on the death rate of corona patients.

This dataset includes 32 columns, here we just selected 5 of them to apply our models on them, five columns are animal products, cereals excluding beer, obesity, vegetal products, and deaths due to the COVID-19 pandemic.

After choosing the five columns, we split the data into animal products, cereals excluding beer, obesity, vegetable products as features, then we set deaths due to the COVID-19 as the target data to apply our models on.

Figure 1.

Showing our records “feature data” using Jupyter notebook.



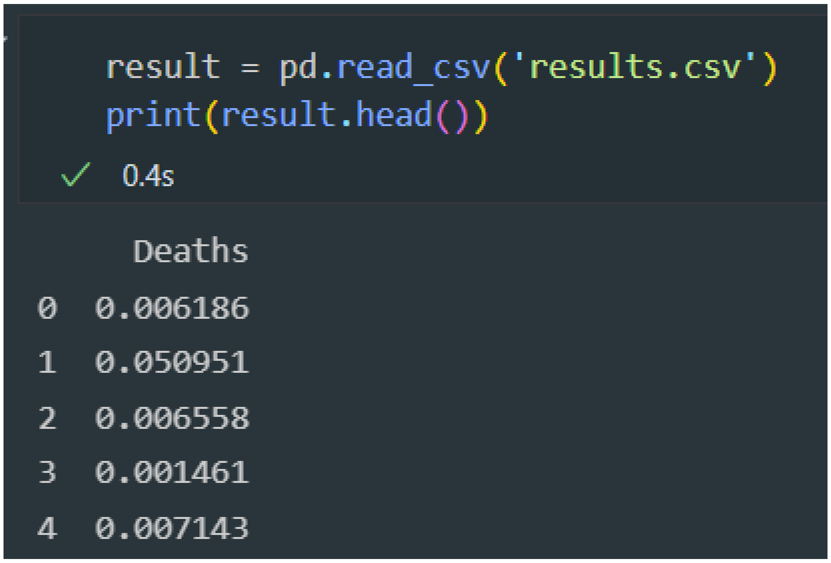
```
dataset = pd.read_csv('records.csv')
print(dataset.head())
```

✓ 0.1s

```
   Animal Products  Cereals - Excluding Beer  Obesity  Vegetal Products
0          21.6397                    8.0353      4.5           28.3684
1          32.0002                    2.6734     22.3           17.9998
2          14.4175                    4.2035     26.6           35.5857
3          15.3041                    6.5545      6.8           34.7010
4          27.7033                    3.2153     19.1           22.2995
```

Figure 2.

Showing our result column “target data” using Jupyter notebook.



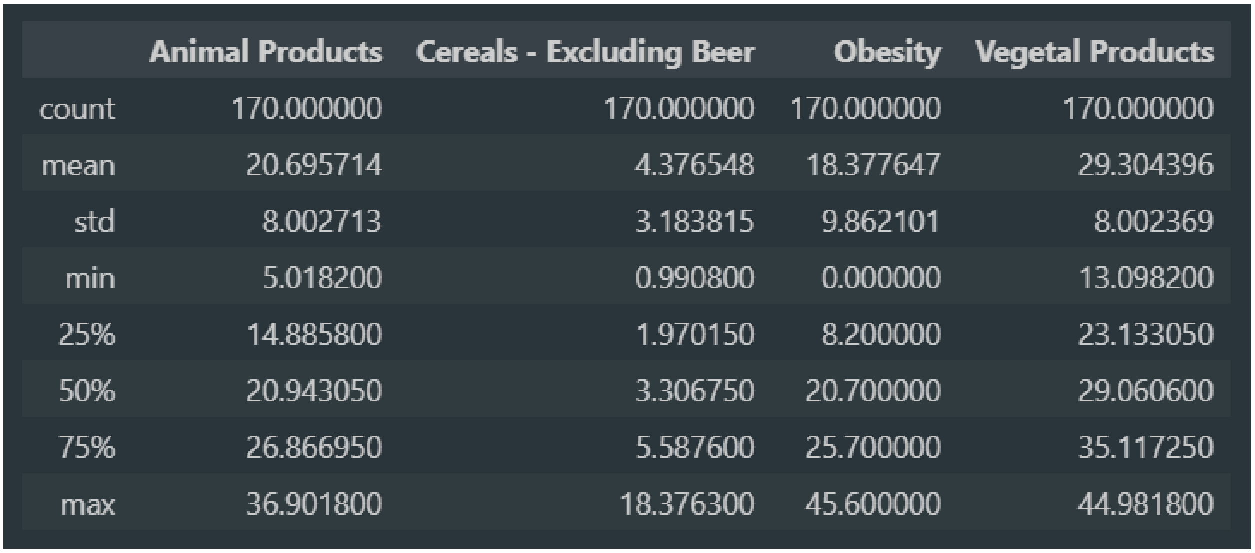
```
result = pd.read_csv('results.csv')
print(result.head())
```

```
     Deaths
0  0.006186
1  0.050951
2  0.006558
3  0.001461
4  0.007143
```

After splitting data into records and results data, we applied exploratory data analysis “EDA” to present information about data like mean, media, standard deviation, minimum ,and maximum as shown in figure 3.

Figure 3.

Shows a brief statistic about the dataset.

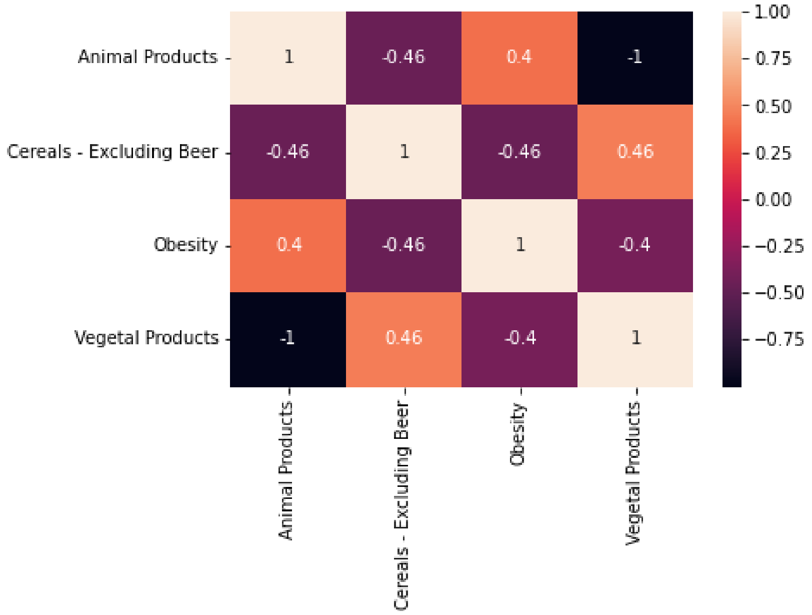


	Animal Products	Cereals - Excluding Beer	Obesity	Vegetal Products
count	170.000000	170.000000	170.000000	170.000000
mean	20.695714	4.376548	18.377647	29.304396
std	8.002713	3.183815	9.862101	8.002369
min	5.018200	0.990800	0.000000	13.098200
25%	14.885800	1.970150	8.200000	23.133050
50%	20.943050	3.306750	20.700000	29.060600
75%	26.866950	5.587600	25.700000	35.117250
max	36.901800	18.376300	45.600000	44.981800

Then we performed correlation analysis on the dataset to find out that the first column “Animal Products” and the fourth column “Vegetable Products” are highly correlated as shown in figure 4.

Figure 4.

A heat map shows the correlation between the columns in the dataset.



5.2 Multicollinearity problem

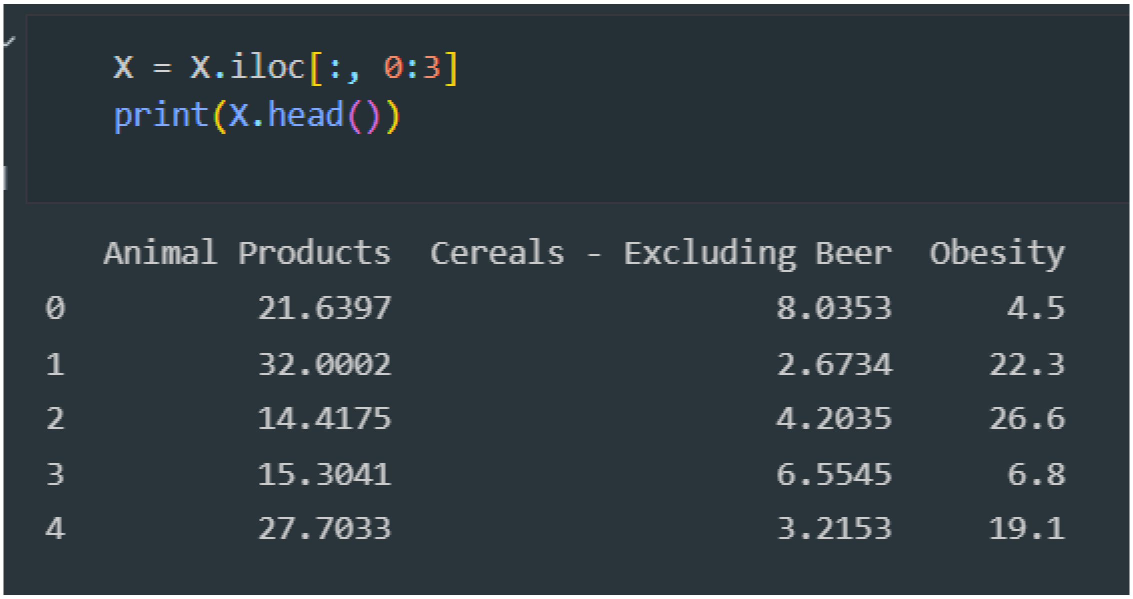
Multicollinearity happens when independent variables in the regression model are highly correlated to each other so that you can predict the values of one column from the other column. It makes it hard to interpret the model and also creates an overfitting problem. It is a common assumption that people test before selecting the variables into the regression model.

The solution to this problem is one of two things :

- The first thing is to remove the values that are highly correlated.
- The second is to remove one of the columns that are highly correlated.

So we performed the second solution which is to remove the fourth column “Vegetable Products” as shown in figure 5.

Figure 5.



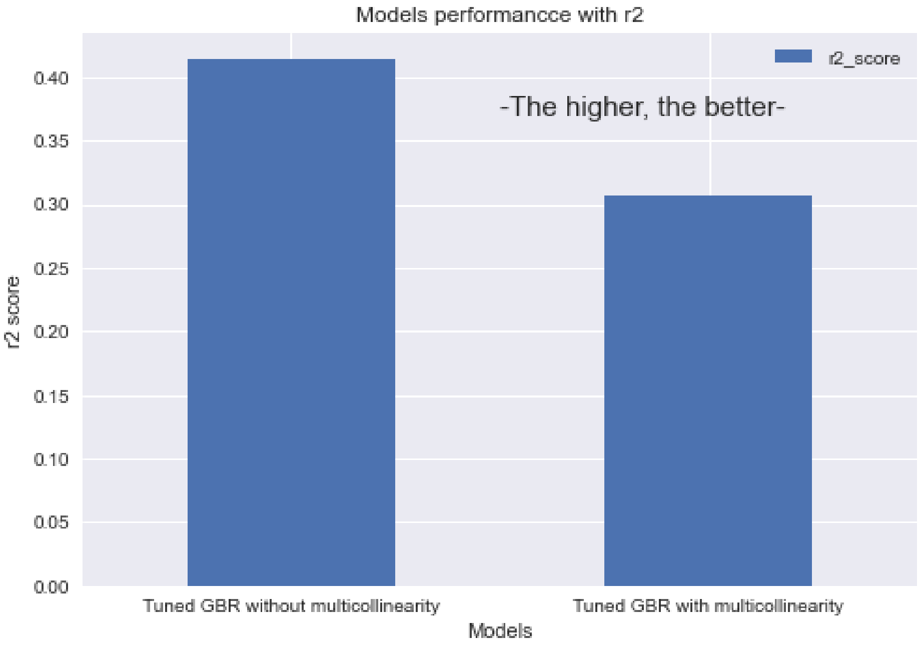
```
X = X.iloc[:, 0:3]
print(X.head())
```

```
   Animal Products  Cereals - Excluding Beer  Obesity
0          21.6397                    8.0353      4.5
1          32.0002                    2.6734     22.3
2          14.4175                    4.2035     26.6
3          15.3041                    6.5545      6.8
4          27.7033                    3.2153     19.1
```

Multicollinearity affects the prediction model performance and if this problem could be spotted and solved this will enhance the model prediction performance as shown in figure 6, the Gradient Boosting regressor performance tuned with GridSearch on the dataset before and after solving the multicollinearity issue.

Figure 6.

Shows tuned GBR with GridSearch performance in R^2 before and after solving the multicollinearity issue.



5.3 Experiments Scenarios

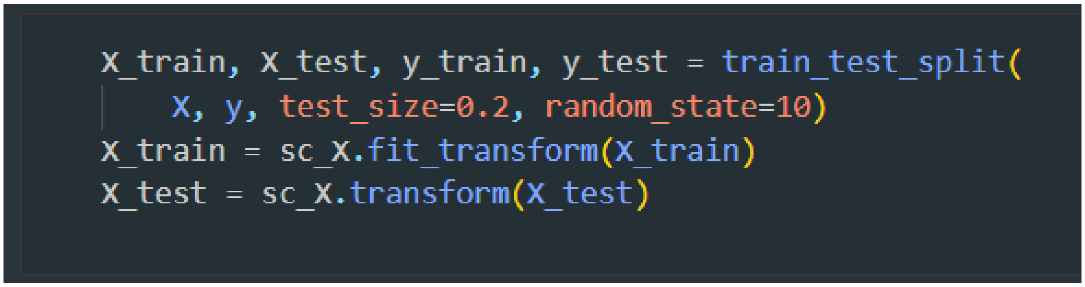
In this work, it’s time to implement our models on Covid-19 nutrition dataset.

After installing “Jupyter IDE”, we import necessary modules such as Sklearn, Pandas, Numpy, Seaborn, Pyplot from Matplotlib and other regression models and so on.

The second stage was assigning data into training and testing sets to be applied in our models as shown in figure 7.

Figure 7.

showing that data has been spit into testing and training sets



```
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=10)
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)
```

Here we applied the following models:

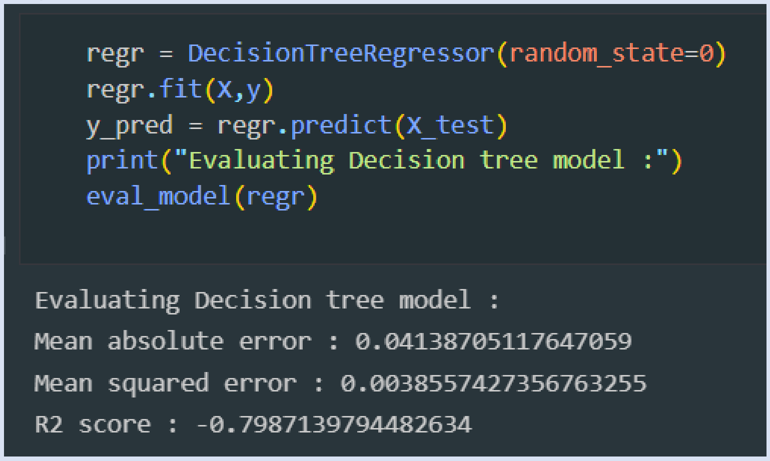
1. Decision Tree
2. Lasso
3. Ridge
4. Bayesian Ridge
5. K-Nearest Neighbors “KNN”
6. Support Vector Machine “SVM”
7. multilayer perceptron MLP
8. Random Forest
9. Gradient Boosting Regressor “GBR”

5.4 Experiment Models

- Decision Tree:
  A decision tree is the model we apply which is a tree where each node represents a feature (attribute), each link (branch) represents a decision (rule) and each leaf represents an outcome (output). Decision tree is a supervised learning used for classification.

We set our model for train and prediction as shown in figure 8.

Figure 8.

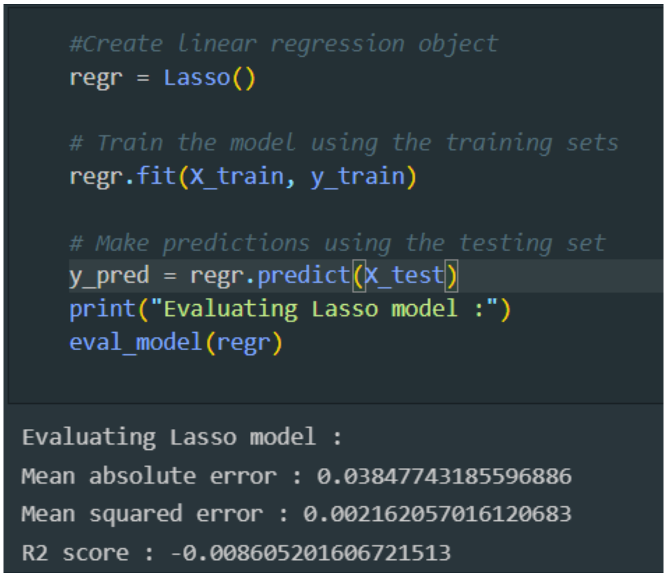


```
regr = DecisionTreeRegressor(random_state=0)
regr.fit(X,y)
y_pred = regr.predict(X_test)
print("Evaluating Decision tree model :")
eval_model(regr)
```

```
Evaluating Decision tree model :
Mean absolute error : 0.04138705117647059
Mean squared error : 0.0038557427356763255
R2 score : -0.7987139794482634
```

- Lasso Regression : lasso (least absolute shrinkage and selection operator; also Lasso or LASSO) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the resulting statistical model.
  We set our model for train and prediction as shown in figure 9.

Figure 9.

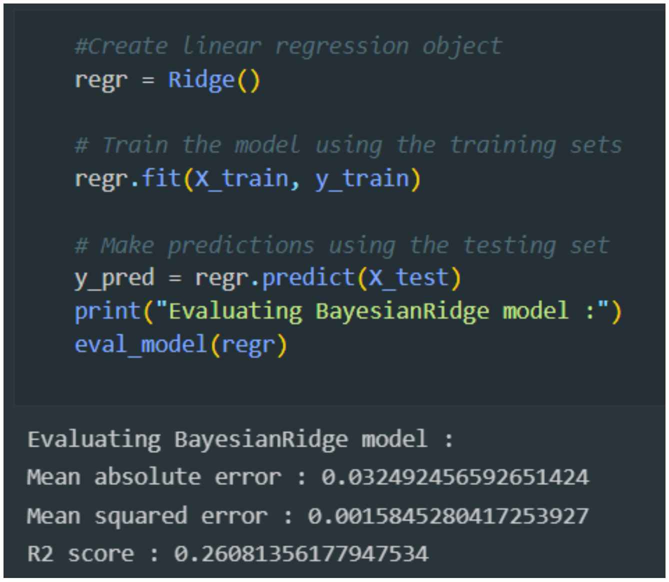


```
#Create linear regression object
regr = Lasso()

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)
print("Evaluating Lasso model :")
eval_model(regr)
```

```
Evaluating Lasso model :
Mean absolute error : 0.03847743185596886
Mean squared error : 0.002162057016120683
R2 score : -0.008605201606721513
```

- Ridge regression: is a method of estimating the coefficients of multiple-regression models in scenarios where linearly independent variables are highly correlated. It has been used in many fields including econometrics, chemistry, and engineering.
  We set our model for train and prediction as shown in figure 10.

Figure 10.

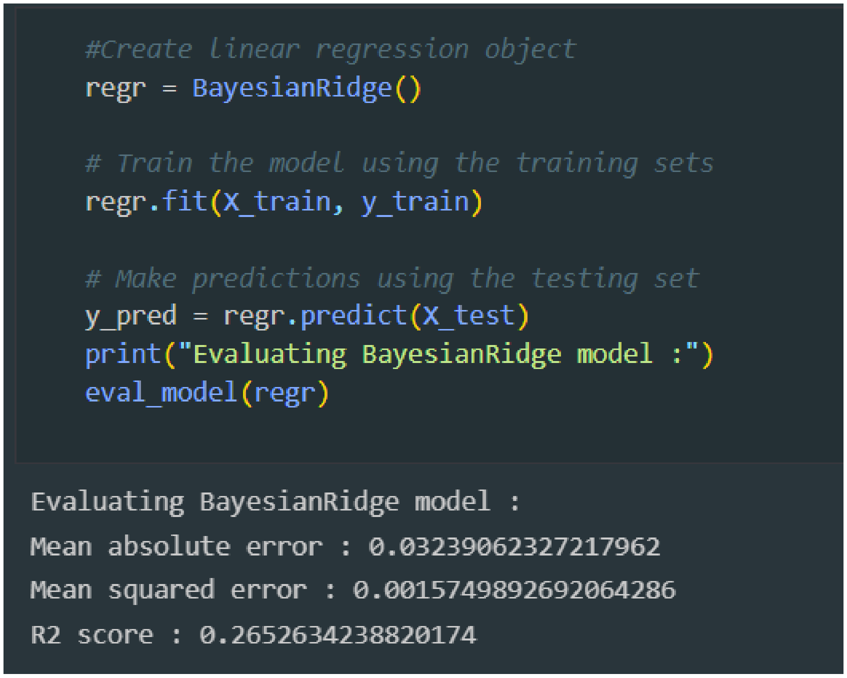


```
#Create linear regression object
regr = Ridge()

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)
print("Evaluating BayesianRidge model :")
eval_model(regr)
```

```
Evaluating BayesianRidge model :
Mean absolute error : 0.032492456592651424
Mean squared error : 0.0015845280417253927
R2 score : 0.26081356177947534
```

- Bayesian regression allows a natural mechanism to survive insufficient data or poorly distributed data by formulating linear regression using probability distributors rather than point estimates. The output or response ‘y’ is assumed to be drawn from a probability distribution rather than estimated as a single value.
  We set our model for train and prediction as shown in figure 11.

Figure 11.

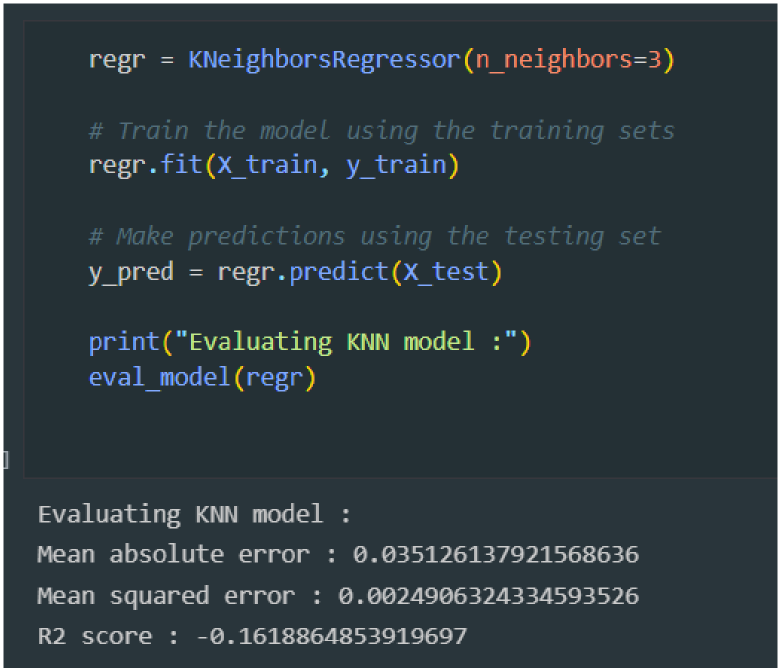


```
#Create linear regression object
regr = BayesianRidge()

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)
print("Evaluating BayesianRidge model :")
eval_model(regr)
```

```
Evaluating BayesianRidge model :
Mean absolute error : 0.03239062327217962
Mean squared error : 0.0015749892692064286
R2 score : 0.2652634238820174
```

- KNN:”K-nearest neighbor” is one of the simplest machine learning algorithms based on classification supervised learning technique. The KNN algorithm assumes the similarity between the new case/data and available cases and puts the new case into the category that is most similar to the available categories.
  We set our model for train and prediction as shown in figure 12.

  Figure 12.

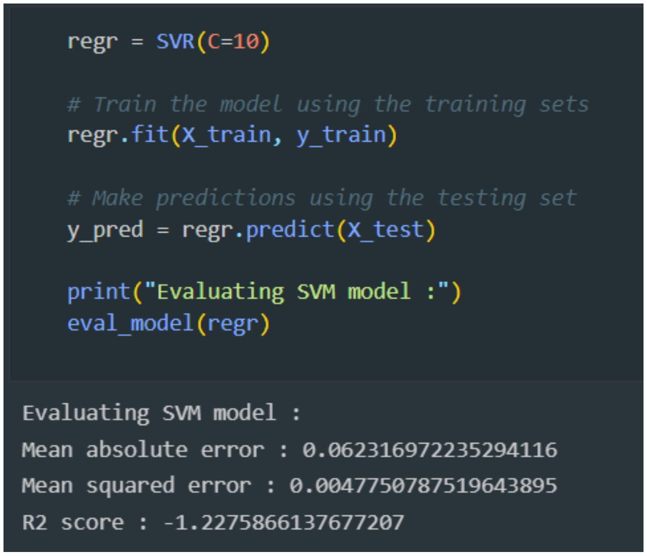


```
regr = KNeighborsRegressor(n_neighbors=3)

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)

print("Evaluating KNN model :")
eval_model(regr)
```

```
Evaluating KNN model :
Mean absolute error : 0.035126137921568636
Mean squared error : 0.0024906324334593526
R2 score : -0.1618864853919697
```

- SVM ”Support Vector Machine”: The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n − dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called an optimal hyperplane.
  We set our model for train and prediction as shown in figure 13.

Figure 13.

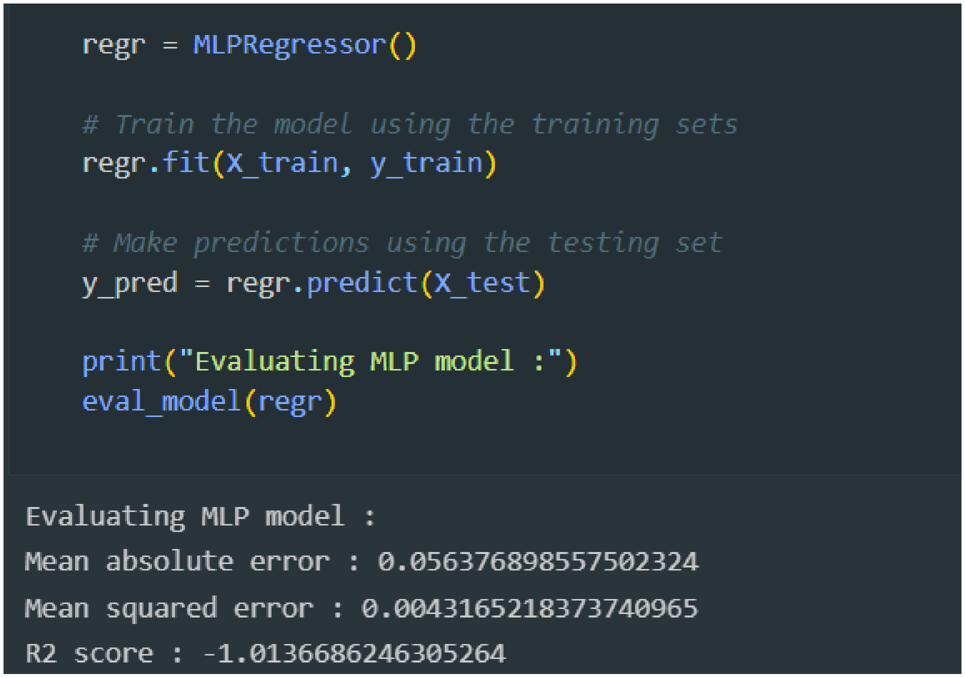


```
regr = SVR(C=10)

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)

print("Evaluating SVM model :")
eval_model(regr)
```

```
Evaluating SVM model :
Mean absolute error : 0.062316972235294116
Mean squared error : 0.0047750787519643895
R2 score : -1.2275866137677207
```

- A multilayer perceptron (MLP): is a fully connected class of feedforward artificial neural network (ANN).
  We set our model for train and prediction as shown in figure 14.

Figure 14.



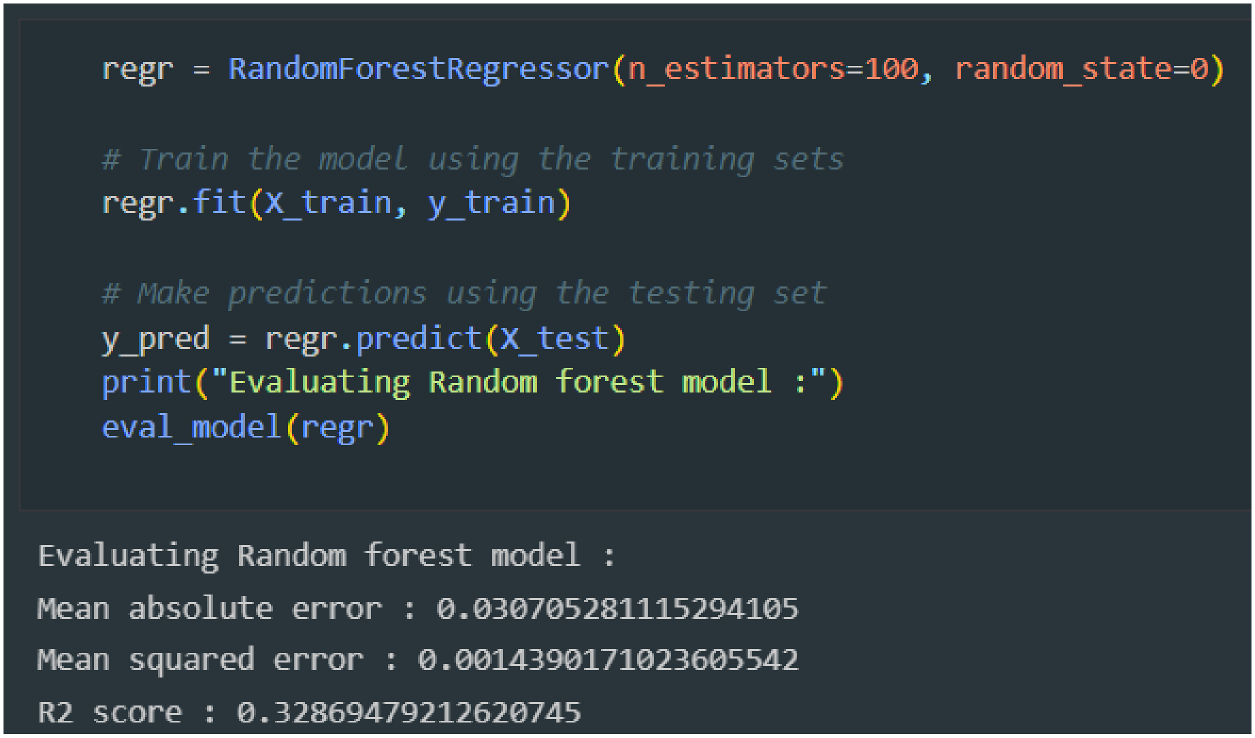
```
regr = MLPRegressor()

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)

print("Evaluating MLP model :")
eval_model(regr)
```

```
Evaluating MLP model :
Mean absolute error : 0.056376898557502324
Mean squared error : 0.0043165218373740965
R2 score : -1.0136686246305264
```

- Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.

For regression tasks, the mean or average prediction of the individual trees is returned. Random decision forests correct for decision trees' habit of overfitting to their training set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance.
We set our model for train and prediction as shown in figure 15.

Figure 15.



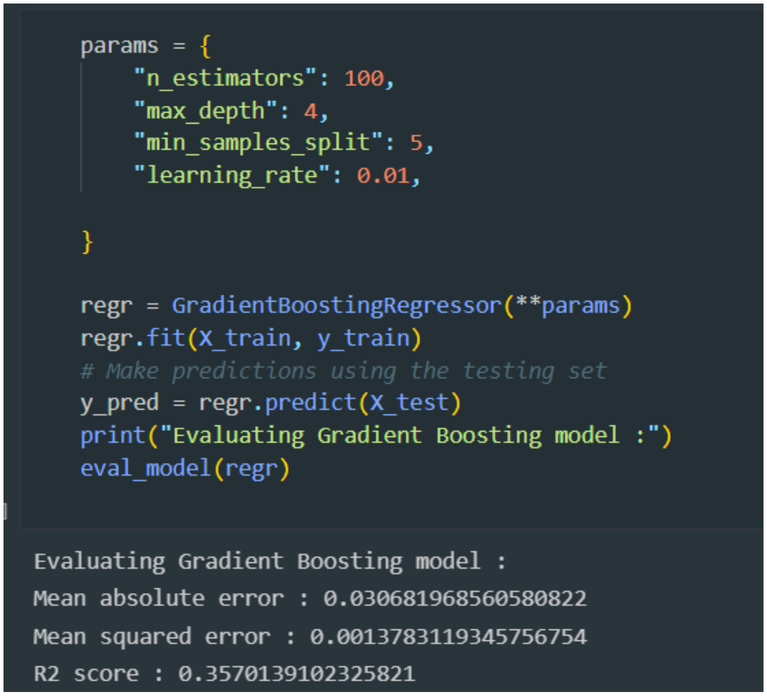
```
regr = RandomForestRegressor(n_estimators=100, random_state=0)

# Train the model using the training sets
regr.fit(X_train, y_train)

# Make predictions using the testing set
y_pred = regr.predict(X_test)
print("Evaluating Random forest model :")
eval_model(regr)
```

```
Evaluating Random forest model :
Mean absolute error : 0.030705281115294105
Mean squared error : 0.0014390171023605542
R2 score : 0.32869479212620745
```

- GBR “Gradient Boosting Regressor” :"Boosting" in machine learning is a way of combining multiple simple models into a single composite model. This is also why boosting is known as an additive model, since simple models (also known as weak learners) are added one at a time, while keeping existing trees in the model unchanged. As we combine more and more simple models, the complete final model becomes a stronger predictor. The term "gradient" in "gradient boosting" comes from the fact that the algorithm uses gradient descent to minimize the loss.

  When gradient boost is used to predict a continuous value – like age, weight, or cost – we're using gradient boost for regression. This is not the same as using linear regression. This is slightly different from the configuration used for classification.
  We set our model for train and prediction as shown in figure 16.

Figure 16.



```
params = {
    "n_estimators": 100,
    "max_depth": 4,
    "min_samples_split": 5,
    "learning_rate": 0.01,

}

regr = GradientBoostingRegressor(**params)
regr.fit(X_train, y_train)
# Make predictions using the testing set
y_pred = regr.predict(X_test)
print("Evaluating Gradient Boosting model :")
eval_model(regr)
```

```
Evaluating Gradient Boosting model :
Mean absolute error : 0.030681968560580822
Mean squared error : 0.0013783119345756754
R2 score : 0.3570139102325821
```

Then after applying the four models, we calculated the R^2 , MAE “Mean absolute error” and MSE Mean squared error. As we said before, the best model depends on the higher value for R^2 and lower for both MAE and MSE as shown in the next table and figures.

Table 4

Shows performance metrics values of applied models “the best model is in **bold**”

MODEL	R^2	MAE	MSE
Decision Tree	0.798713979	0.041387051	0.003855743
Lasso	-0.008605202	0.038477432	0.002162057
Ridge	0.260813562	0.032492457	0.001584528
Bayesian Ridge	0.265263424	0.032390623	0.001574989
KNN	-0.161886485	0.035126138	0.002490632
SVM	-1.227586614	0.062316972	0.004775079
MLP	-1.013668625	0.056376899	0.004316522
Random Forest	0.328694792	0.030705281	0.001439017
GBR	**0.35701391**	**0.030681969**	**0.001378312**

The most efficient prediction model is Gradient Boosting regressor because it gives the higher value for R^2 which indicates the best accuracy, and it gives the lowest values for MAE and MSE which indicates the best result.

Figure 17.

Shows the models' performance in R^2 .

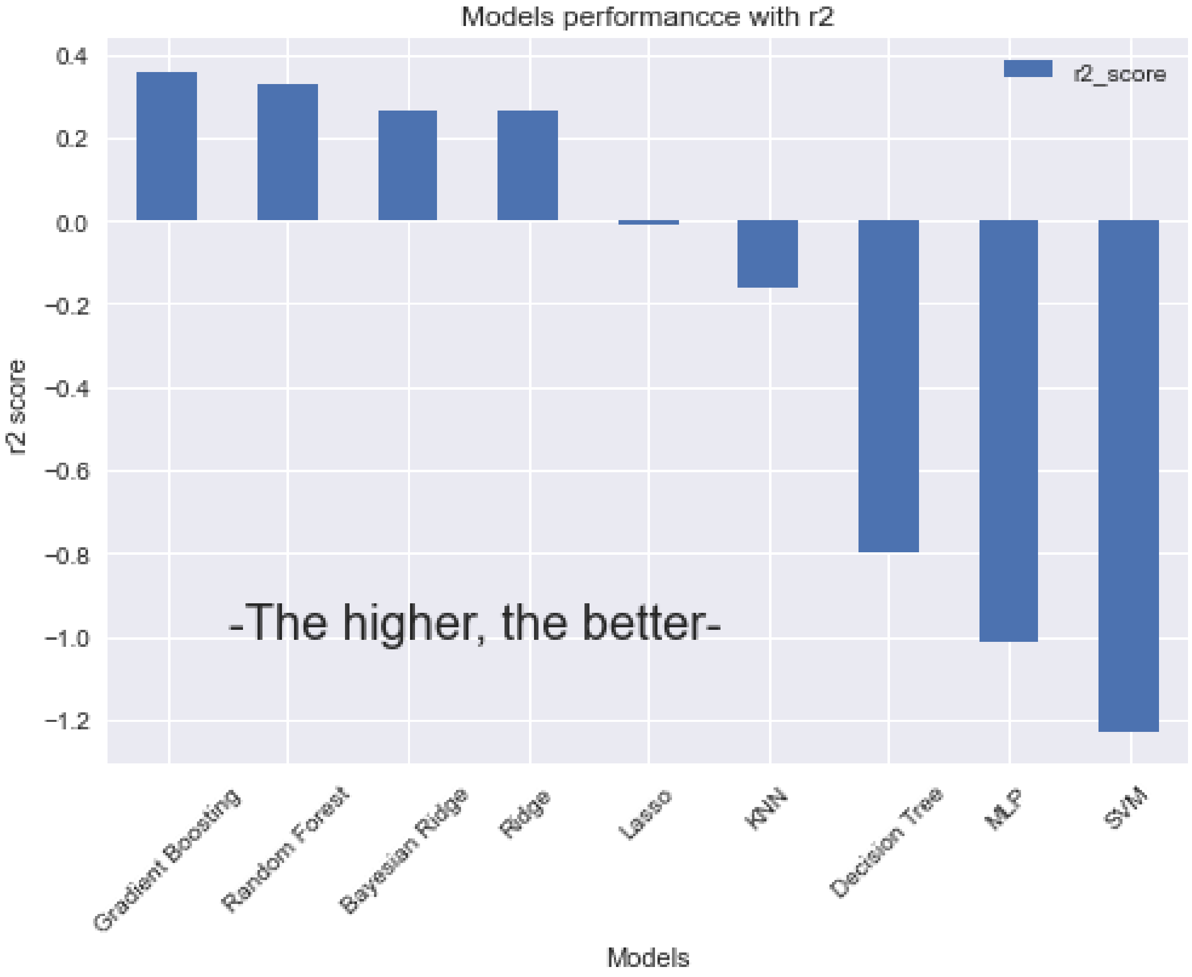


Figure 18.

Shows the models' performance in MAE .

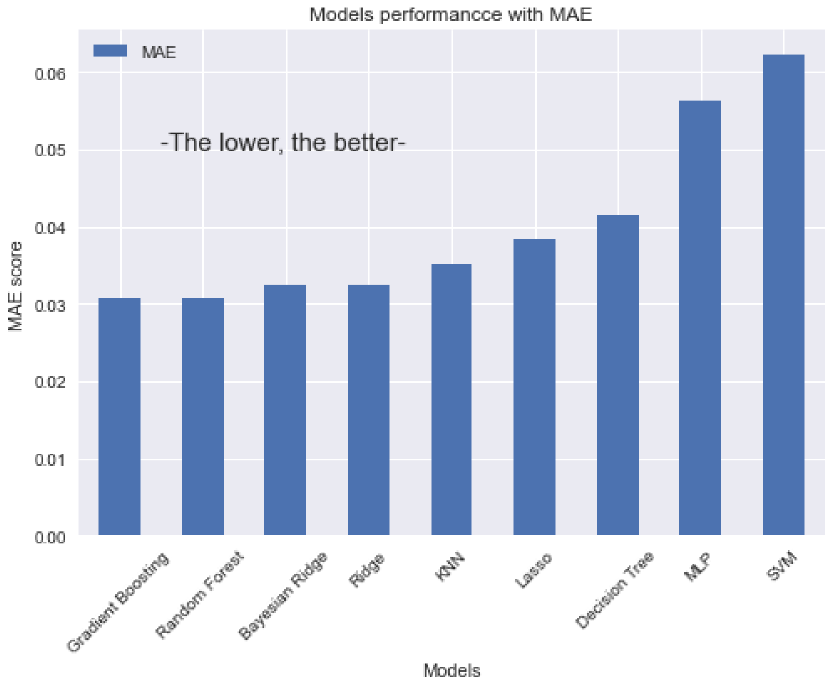
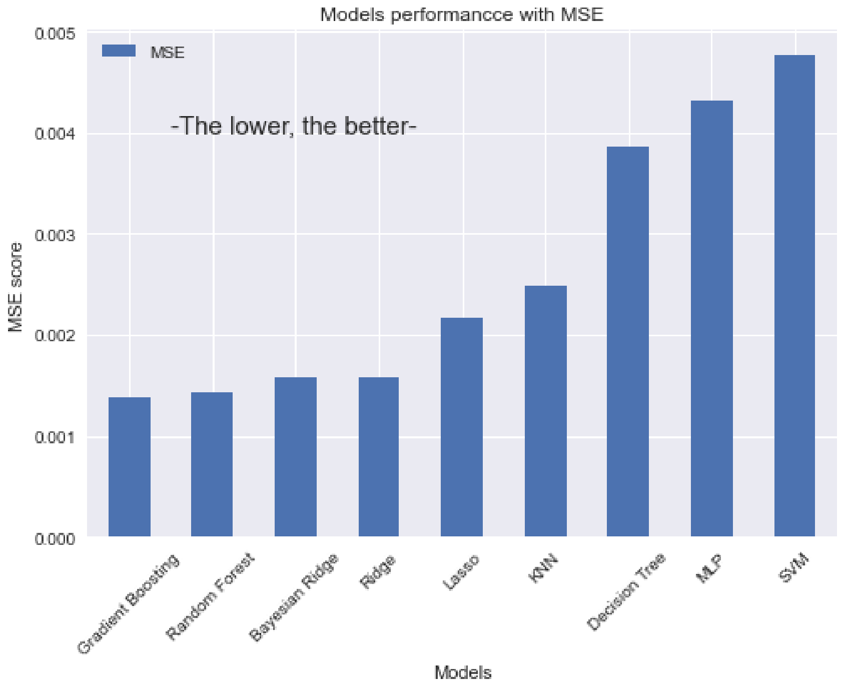


Figure 19.

Shows the models' performance in MSE.



5.5 An optimal solution using GridSearchCV

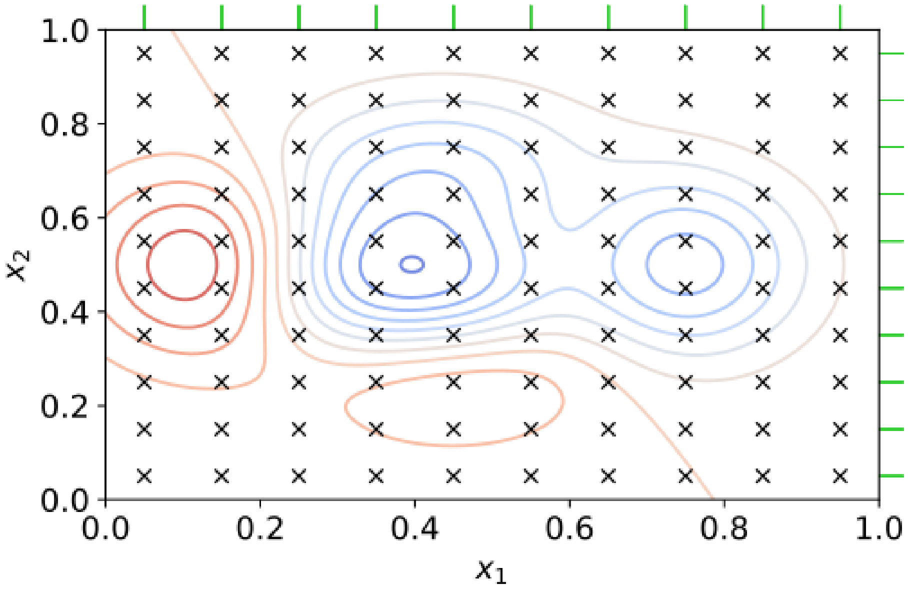
Many times while working on a dataset and using a Machine Learning model we don't know which set of hyperparameters will give us the best result. Passing all sets of hyperparameters manually through the model and checking the result might be a hectic work and may not be possible to do.

To get the best set of hyperparameters we can use Grid Search. Grid Search passes all combinations of hyperparameters one by one into the model and checks the result. Finally, it gives us the set of hyperparameters which gives the best result after passing in the model.

Grid Search uses a different combination of all the specified hyperparameters and their values and calculates the performance for each combination and selects the best value for the hyperparameters. This makes the processing time-consuming and expensive based on the number of hyperparameters involved.

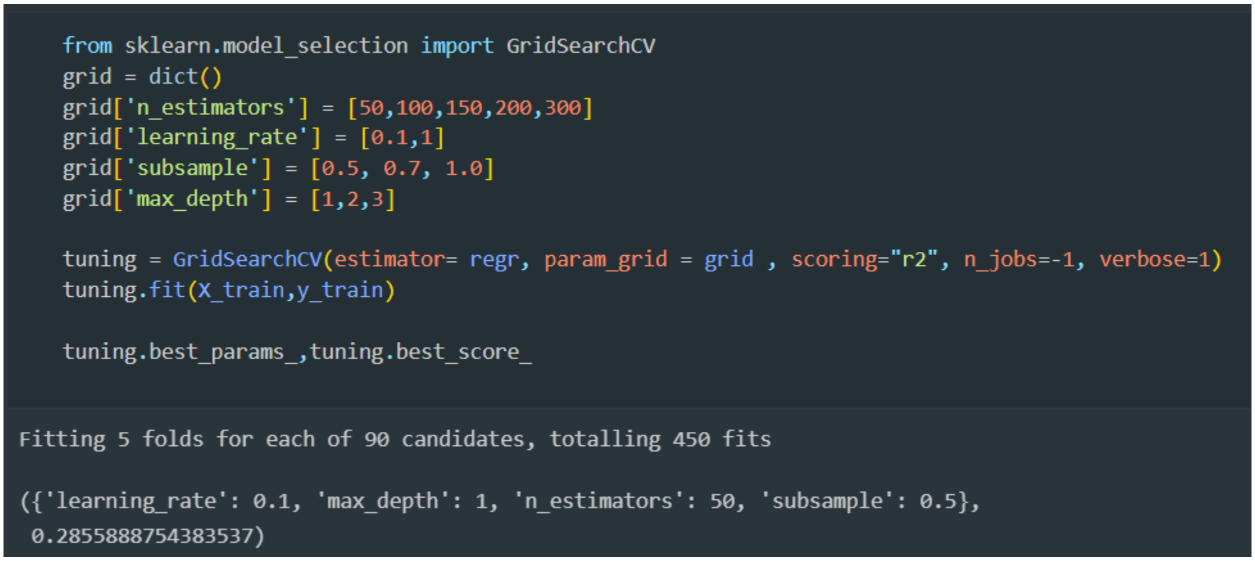
Figure 20.

GridSearch categorizes the hyperparameters depending on their performance.



We set the hyper parameters values of the GridSearch for the Gradient Boosting regressor for the tuning as shown in figure 21.

Figure 21.



```
from sklearn.model_selection import GridSearchCV
grid = dict()
grid['n_estimators'] = [50,100,150,200,300]
grid['learning_rate'] = [0.1,1]
grid['subsample'] = [0.5, 0.7, 1.0]
grid['max_depth'] = [1,2,3]

tuning = GridSearchCV(estimator= regr, param_grid = grid , scoring="r2", n_jobs=-1, verbose=1)
tuning.fit(X_train,y_train)

tuning.best_params_,tuning.best_score_
```

```
Fitting 5 folds for each of 90 candidates, totalling 450 fits

({'learning_rate': 0.1, 'max_depth': 1, 'n_estimators': 50, 'subsample': 0.5},
 0.2855888754383537)
```

As shown in figure 11 we set some hyperparameters to the GridSearch model, then the model found the best values to make the optimal solution:

- Learning-rate : 0.1
- Max-depth : 1
- N-estimators : 50
- Subsample : 1.0

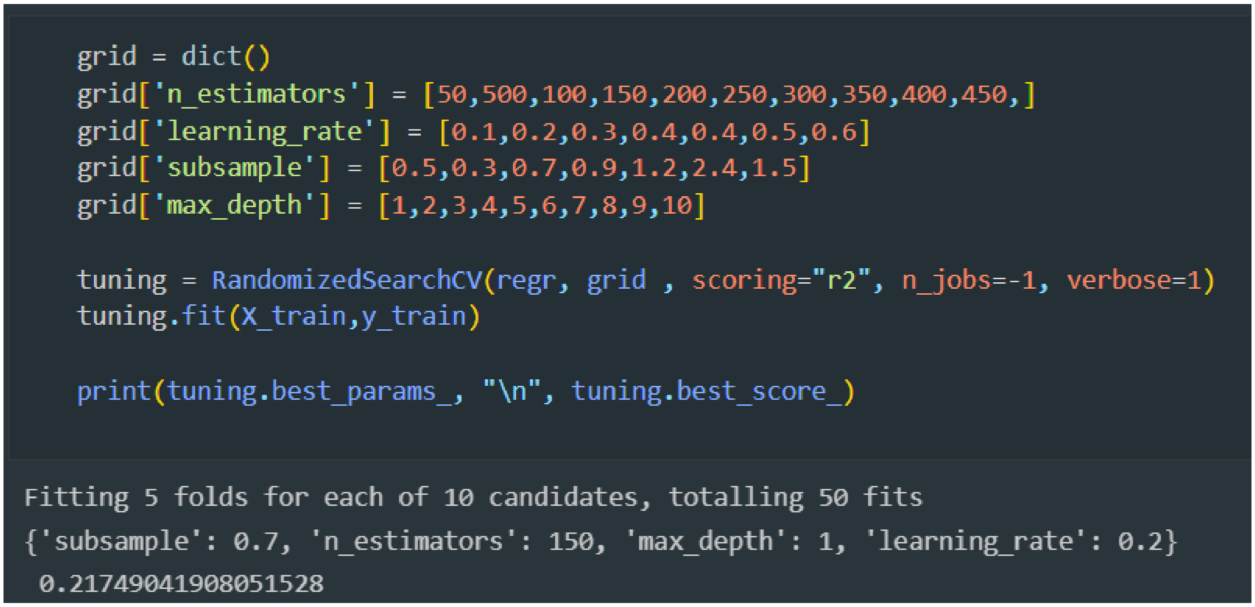
After applying our GridSearch model, we calculated the R^2 , MAE and MSE and the results were 0.414156408 for R^2 , 0.027496502 for MAE and 0.001255821 for MSE.

5.6 An optimal solution using Random Search “RS”

RandomSearch is a family of numerical optimization methods that do not require the gradient of the problem to be optimized, and RS can hence be used on functions that are not continuous or differentiable. Such optimization methods are also known as direct-search, derivative-free, or black-box methods.

We set the hyper parameters values of the RandomSearch for the Gradient Boosting regressor for the tuning as shown in figure 13.

Figure 22.



```
grid = dict()
grid['n_estimators'] = [50,500,100,150,200,250,300,350,400,450,]
grid['learning_rate'] = [0.1,0.2,0.3,0.4,0.4,0.5,0.6]
grid['subsample'] = [0.5,0.3,0.7,0.9,1.2,2.4,1.5]
grid['max_depth'] = [1,2,3,4,5,6,7,8,9,10]

tuning = RandomizedSearchCV(regr, grid , scoring="r2", n_jobs=-1, verbose=1)
tuning.fit(X_train,y_train)

print(tuning.best_params_, "\n", tuning.best_score_)
```

```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
{'subsample': 0.7, 'n_estimators': 150, 'max_depth': 1, 'learning_rate': 0.2}
 0.21749041908051528
```

As shown in figure 12 we set some hyperparameters to the Random Search model, then the model found the best values to make the optimal solution:

- Learning-rate : 0.2
- Max-depth : 1
- N-estimators : 150
- Subsample : 0.7

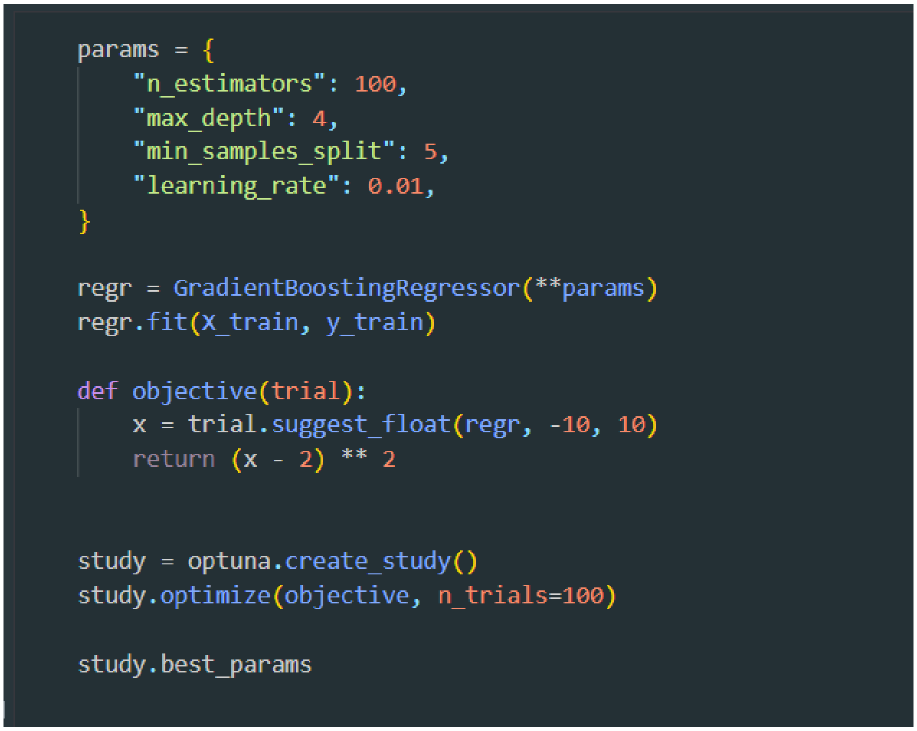
After applying our Random Search model, we calculated the R^2 , MAE and MSE and the results were 0.436324862 for R^2 , 0.025965509 for MAE and 0.0012083 for MSE.

5.7 An optimal solution using Optuna

Optuna is an automatic hyperparameter optimization software framework, particularly designed for machine learning. It features an imperative, define-by-run style user API. This notebook describes the basic usage of Optuna with simple optimization tasks of quadratic function and linear regression.

We set the hyper parameters values of the Otuna for the Gradient Boosting regressor for the tuning as shown in figure 23.

Figure 23.



```
params = {
    "n_estimators": 100,
    "max_depth": 4,
    "min_samples_split": 5,
    "learning_rate": 0.01,
}

regr = GradientBoostingRegressor(**params)
regr.fit(X_train, y_train)

def objective(trial):
    x = trial.suggest_float(regr, -10, 10)
    return (x - 2) ** 2


study = optuna.create_study()
study.optimize(objective, n_trials=100)

study.best_params
```

As shown in figure 13 we set some hyperparameters to the Optuna model, then the model found the best values to make the optimal solution:

- Learning-rate : 0.01
- Max-depth : 4
- min_samples_split : 5

After applying our Optuna model, we calculated the R^2 , MAE and MSE and the results were 0.353611526 for R^2 , 0.030693705 for MAE and 0.001385605 for MSE.

5.8 Comparison between the different optimization techniques applied on Grid Search

Due to the previous results after applying the different optimization techniques, the tuned GridSearch with Random Search gives the best results in R^2, MAE and MSE as demonstrated in the next table and figures.

Table 5

Shows performance metrics for the different optimization techniques applied on Gradient Boosting regressor.

Model	MSE	MAE	R2
GBR GridSearchCV	0.001255821	0.027496502	0.414156408
GBR Random Search	**0.0012083**	**0.025965509**	**0.436324862**
GBR Optuna	0.001385605	0.030693705	0.353611526
GBR	0.001378312	0.030681969	0.35701391

Figure 24.

Shows the optimization models' performance in R^2

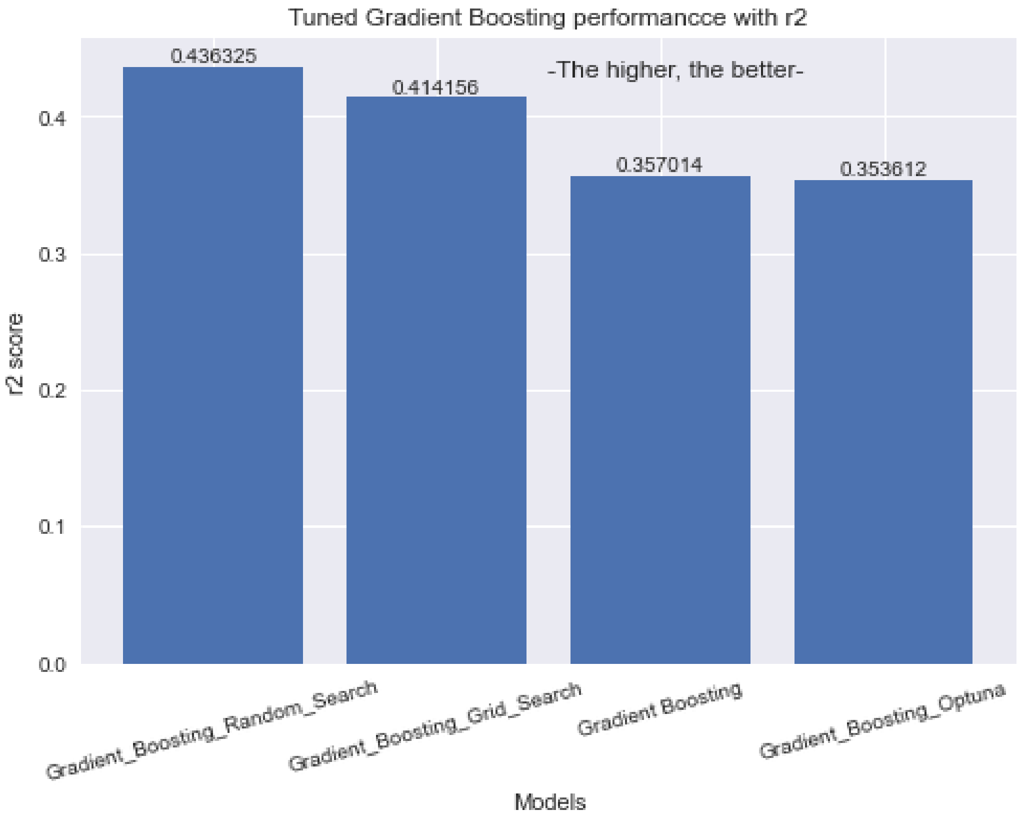


Figure 25.

Shows the optimization models' performance in MSE

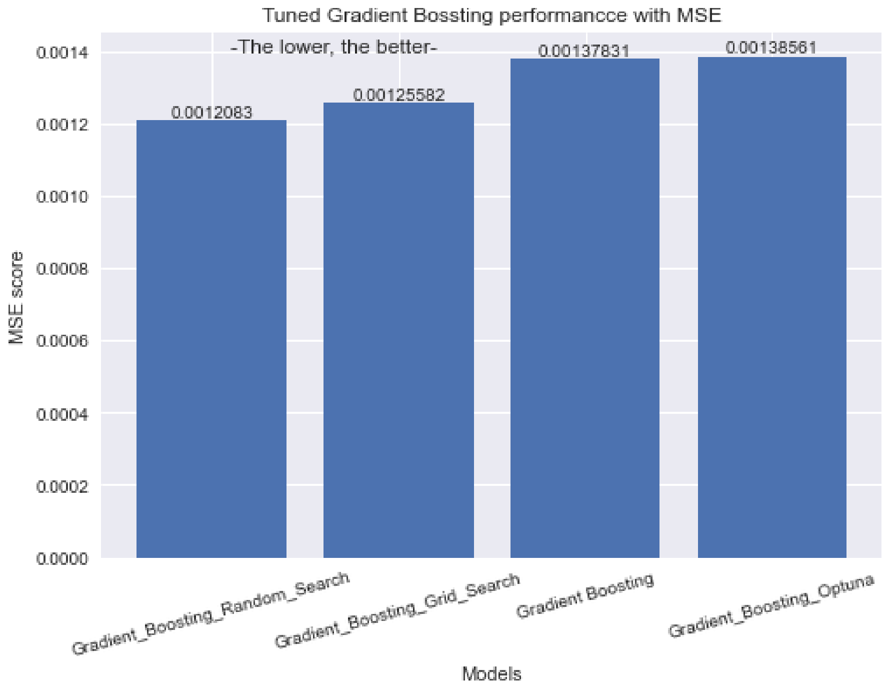
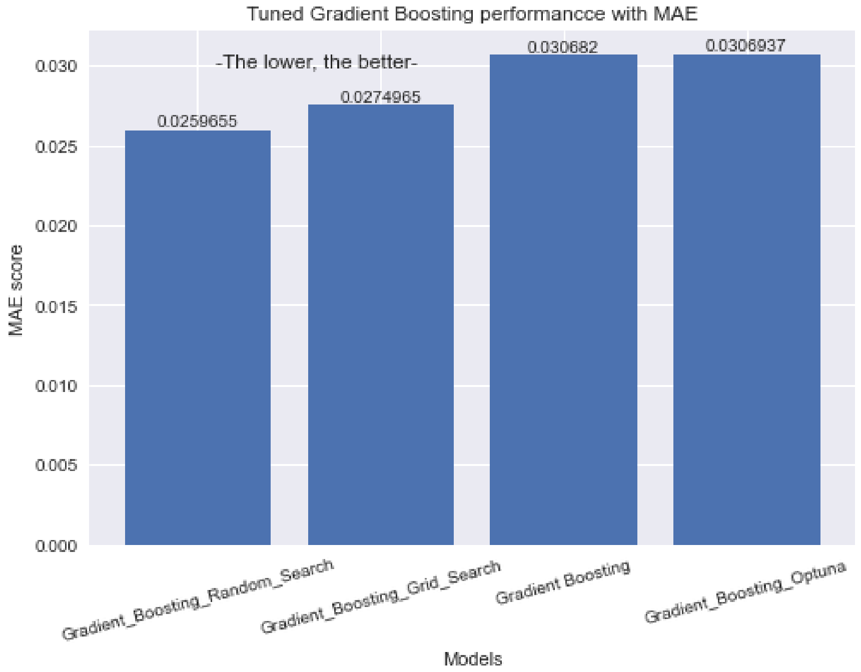


Figure 26.

Shows the optimization models' performance in MAE



6. conclusion

Nutrition plays an essential role in maintaining the health of the body and strengthening the immune system against various diseases and epidemics. For example, alternating one food component, such as fat, may increase the chance of human infection with epidemics and diseases such as Covid 19, and the use of modern technical means to know nutrients such as machine learning is one of the most important and accurate ways to reduce death rates.

In our research we applied Decision Tree, Lasso ,Ridge ,Bayesian Ridge ,KNN, SVM, MLP ,Random Forest and Gradient Boosting on the nutrition dataset of Covid-19, then we calculated R^2, MAE, and MSE. We found that Gradient Boosting Regressor gives the higher value of R2 than other models and lower values for both MAE and MSE which indicates that Gradient Boosting Regressor gives the best results with a good level of accuracy.

The second stage was applying three different optimization techniques on Gradient Boosting Regressor which are:

- Random Search
- GridSearch
- Optuna

We calculated R^2, MAE, and MSE for each of the optimization techniques on Gradient Boosting Regressor and found out that tuned GBR with Random Search gives the best performance and level of accuracy as it scored the highest in R^2 and lowest in MSE and MAE.

You can access the research source code from GitHub [26].

7. References

[1] S. Snuggs, S. McGregor, Food & meal decision making in lockdown: how and who has Covid-19 affected? Food Qual. Prefer. 89 (2021) 104145.

[2] C.M. Galanakis, The food systems in the era of the coronavirus (COVID-19) pandemic crisis, Foods 9 (4) (2020) 523.

[3] M. Rizou, I.M. Galanakis, T.M. Aldawoud, C.M. Galanakis, Safety of foods, food supply chain and environment within the COVID-19 pandemic, Trends Food Sci. Technol. 102 (2020) 293–299.

[4] K. Mishra, J. Rampal, The COVID-19 Pandemic and Food Insecurity: A Viewpoint on India, vol. 135, World Development, 2020, p. 105068.

[5] Y. Zhou, Y. Lu, Z. Pei, "Intelligent diagnosis of Alzheimer's disease based on internet of things monitoring system and deep learning classification method, Microprocessor. Microsyst. 83 (2021) 104007.

[6] O. M. Elzeki, M. Abd Elfattah, H. Salem, A. E. Hassanien, and M. Shams, "A novel perceptual two layer image fusion using deep learning for imbalanced COVID-19 dataset," PeerJ Computer Science, vol. 7, 2021.

[7] O.M. Elzeki, M. Shams, S. Sarhan, M. Abd Elfattah, A.E. Hassanien, COVID-19: a new deep learning computer-aided model for classification, PeerJ Computer Science 7 (2021) e358.

[8] T. Alafif, A.M. Tehame, S. Bajaba, A. Barnawi, S. Zia, Machine and deep learning towards COVID-19 diagnosis and treatment: survey, challenges, and future directions, Int. J. Environ. Res. Publ. Health 18 (3) (2021). Art. no. 3.

[9] A.K. Gopalakrishnan, "A Food Recommendation System Based on BMI, BMR, K-NN Algorithm, and a BPNN," in Machine Learning for Predictive Analysis, Springer, 2021, pp. 107–118.

[10] C. P ́erez-Rodrigo, et al., Patterns of Change in dietary habits and physical activity during lockdown in Spain due to the COVID-19 pandemic, Nutrients 13 (2) (2021). Art. no. 2.

[11] R.M. Fanelli, Changes in the food-related behaviour of Italian consumers during the COVID-19 pandemic, Foods 10 (1) (2021). Art. no. 1.

[12] Z. Shen, A. Shehzad, S. Chen, H. Sun, J. Liu, Machine learning based approach on food recognition and nutrition estimation, Procedia Comput. Sci. 174 (2020) 448–453.

[13] C.E. Onu, P.K. Igbokwe, J.T. Nwabanne, C.O. Nwajinka, P.E. Ohale, Evaluation of optimization techniques in predicting optimum moisture content reduction in drying potato slices, Artif. Intell. Agric. 4 (2020) 39–47.

[14] A. Tonda, et al., “Interactive Machine Learning for Applications in Food Science,” in Human and Machine Learning, Springer, 2018, pp. 459–477.

[15] S. Usha, M. Karthik, R. Jenifer, P.G. Scholar, Automated sorting and grading of vegetables using image processing, Int. J. Eng. Res. Gen. Sci. 5 (6) (2017). Art. no. 6.

[16] T. Kodama, Y. Hata, “Development of classification system of rice disease using artificial intelligence,” in 2018, IEEE Int. Conf. Syst. Man Cybern. (2018) 3699–3702.

[17] N.E.M. Khalifa, M.H.N. Taha, L.M. Abou El-Maged, A.E. Hassanien, “Artificial Intelligence in Potato Leaf Disease Classification: A Deep Learning Approach,” in Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges, Springer, 2021, pp. 63–79.

[18] B. Liu, Y. Zhang, D. He, Y. Li, Identification of apple leaf diseases based on deep convolutional neural networks, Symmetry 10 (1) (2018). Art. no. 1.

[19] R. De Amicis, et al., Patients with severe obesity during the COVID-19 pandemic: how to maintain an adequate multidisciplinary nutritional rehabilitation program? Obes. Facts (2021) 1–9.

[20] M. Marazuela, A. Giustina, M. Puig-Domingo, Endocrine and metabolic aspects of the COVID-19 pandemic, Rev. Endocr. Metab. Disord. 21 (4) (2020) 495–507.

[21] S. Camar´ena, Artificial intelligence in the design of transition to sustainable food systems, J. Clean. Prod. (2020) 122574.

[22] J.M. Soon, I. Vanany, I.R.A. Wahab, R.H. Hamdan, M.H. Jamaludin, Food safety and evaluation of intention to practice safe eating out measures during COVID-19: cross sectional study in Indonesia and Malaysia, Food Contr. 125 (2021) 107920.

[23] Best metric for Regression? RMSE, MSE, MAE, R2 | by Nicolas Maurer | Medium.

[24] Kaggle Covid-19 healthy diet dataset

[25] HANA: A Healthy Artificial Nutrition Analysis model during COVID-19 pandemic.

[26] Death Rate Prediction Based on Nutrition using Several Optimization Techniques.