|  |  |
| --- | --- |
| Algorithm: Gradient Boosting | |
| USN : 1MS17CS143 | NAME : Sathvik K P |
| USN : 1MS17CS148 | NAME : Sathvik B |

**Description of the Algorithm:**

The term ‘Boosting’ refers to a family of algorithms which converts weak learner to strong learners. To convert weak learner to strong learner, we’ll combine the prediction of each weak learner using methods like:

• Using average/ weighted average

• Considering prediction has higher vote

To find weak rule, we apply base learning (ML) algorithms with a different distribution. Each time base learning algorithm is applied, it generates a new weak prediction rule. This is an iterative process. After many iterations, the boosting algorithm combines these weak rules into a single strong prediction rule.

For choosing the right distribution, here are the following steps:

Step 1: The base learner takes all the distributions and assign equal weight or attention to each observation.

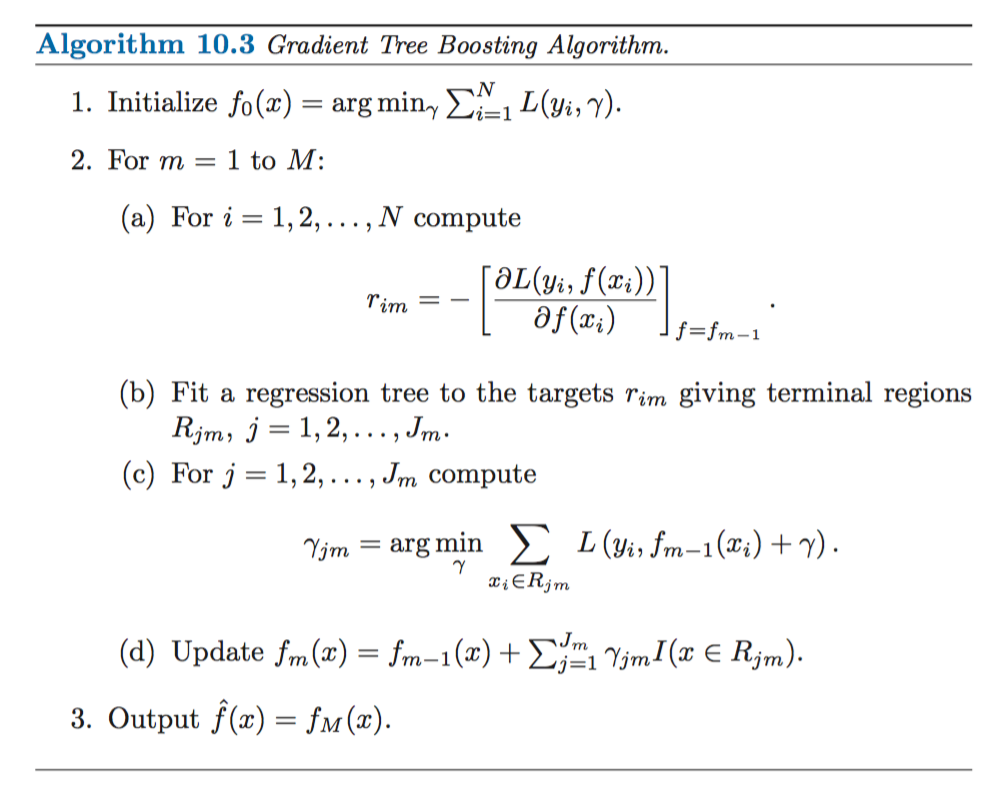
Step 2: If there is any prediction error caused by first base learning algorithm, then we pay higher attention to observations having prediction error. Then, we apply the next base learning algorithm.

Step 3: Iterate Step 2 till the limit of base learning algorithm is reached or higher accuracy is achieved.

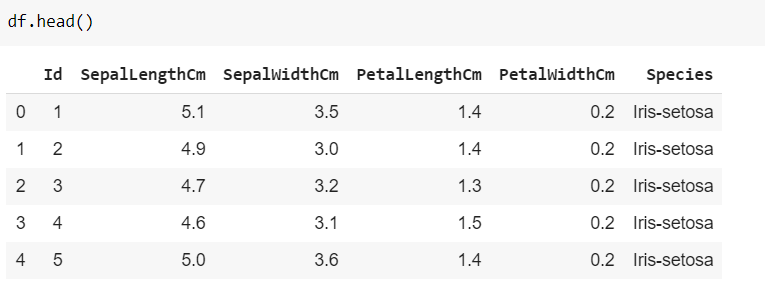
In gradient boosting, it trains many model sequentially. Each new model gradually minimizes the loss function (y = ax + b + e, e needs special attention as it is an error term) of the whole system using Gradient Descent method. The learning procedure consecutively fit new models to provide a more accurate estimate of the response variable.

The principle idea behind this algorithm is to construct new base learners which can be maximally correlated with negative gradient of the loss function, associated with the whole ensemble.

**Algorithm Pseudocode:**



**Data set Used: (Attach Screen shot of the few rows)**



**Challenges faced during the implementation of the program:**

1. Low size of dataset, therefore high accuracy
2. Huge number of iterations

**Code:**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.preprocessing import StandardScaler, MinMaxScaler

from sklearn.decomposition import PCA

from sklearn.feature\_selection import chi2, SelectKBest

from sklearn.metrics import accuracy\_score

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import confusion\_matrix,roc\_auc\_score,average\_precision\_score

from sklearn.ensemble import GradientBoostingClassifier

df = pd.read\_csv("iris.csv")

df.isna().sum()

sns.FacetGrid(df, hue="Species", size=5) \

   .map(plt.scatter, "SepalLengthCm", "SepalWidthCm") \

   .add\_legend()

ax = sns.boxplot(x="Species", y="PetalLengthCm", data=df)

ax = sns.stripplot(x="Species", y="PetalLengthCm", data=df, jitter=True, edgecolor="gray")

corr\_matrix = df.corr()

corr\_matrix

fig, ax = plt.subplots(figsize = (9,8))

sns.heatmap(corr\_matrix, ax=ax, cmap='YlGnBu', linewidths=0.1)

df['Species'].unique()

df['Species'] = df['Species'].apply(lambda x: 0 if x=='Iris-setosa' else (1 if x=='Iris-versicolor' else 2))

X=df.drop(['Species','Id'],axis=1)

y=df['Species']

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

clf = GradientBoostingClassifier(random\_state=0)

clf.fit(X\_train, Y\_train)

y\_pred = clf.predict(X\_test)

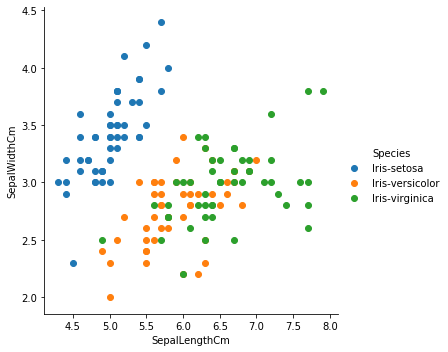
y\_pred\_train = clf.predict(X\_train)

print(accuracy\_score(Y\_test,y\_pred))

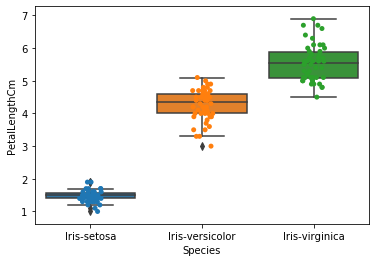
print(accuracy\_score(Y\_train,y\_pred\_train))

**Output: (Screen shots)**

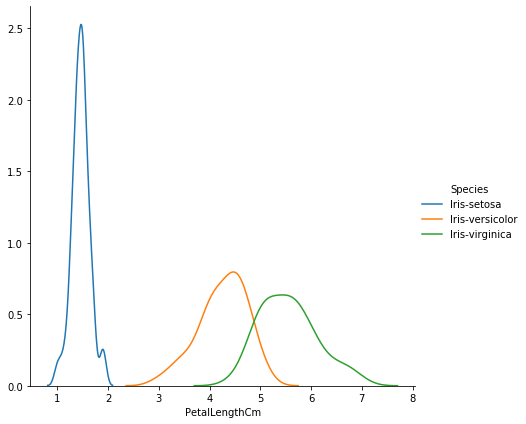
Scatter plot of the different species with SepalLength as X axis and SepalWidth as Y axis



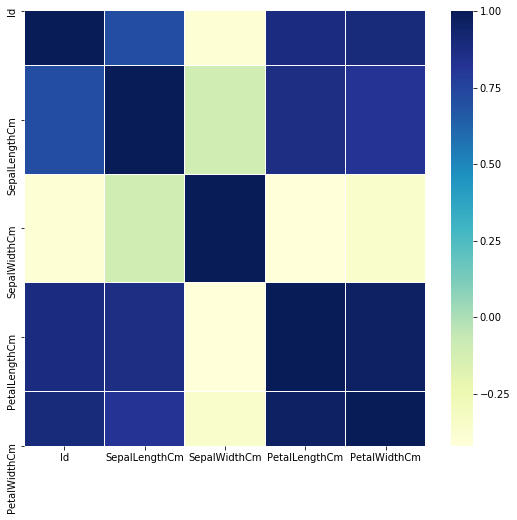
Boxplot to determine if outliers are present



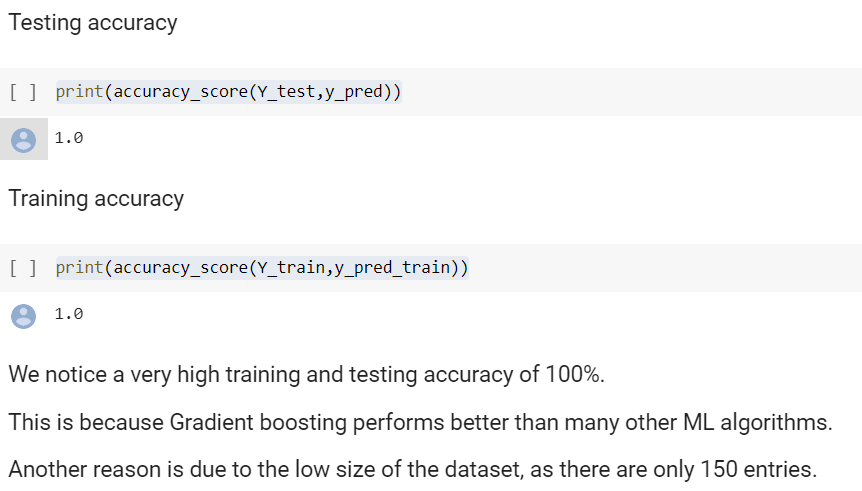
Graph to plot petal length per species



Correlation matrix



Results



**References:**

1. <https://www.analyticsvidhya.com/blog/2015/11/quick-introduction-boosting-algorithms-machine-learning/>
2. <https://en.wikipedia.org/wiki/Gradient_boosting>
3. <https://medium.com/mlreview/gradient-boosting-from-scratch-1e317ae4587d>
4. <https://towardsdatascience.com/boosting-algorithms-explained-d38f56ef3f30>