Gurucharan M K Follow

Jul 15, 2020 · 5 min read ★ · Listen

Machine Learning Basics: Decision Tree Regression

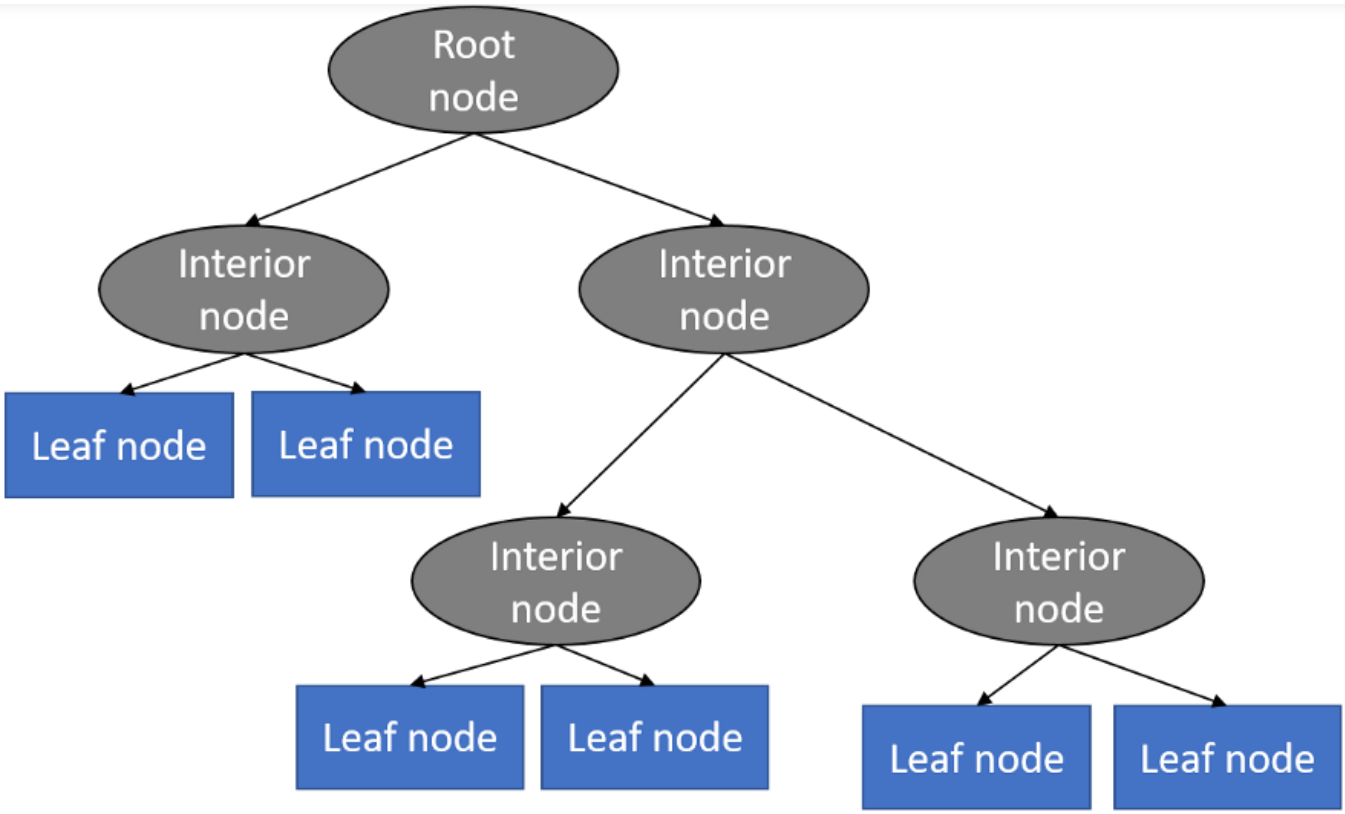
Implement the Decision Tree Regression algorithm and plot the results.

Previously, I had explained the various Regression models such as Linear, Polynomial and Support Vector Regression. In this article, I will walk you through the Algorithm and Implementation of Decision Tree Regression with a real-world example.

Overview of Decision Tree Algorithm

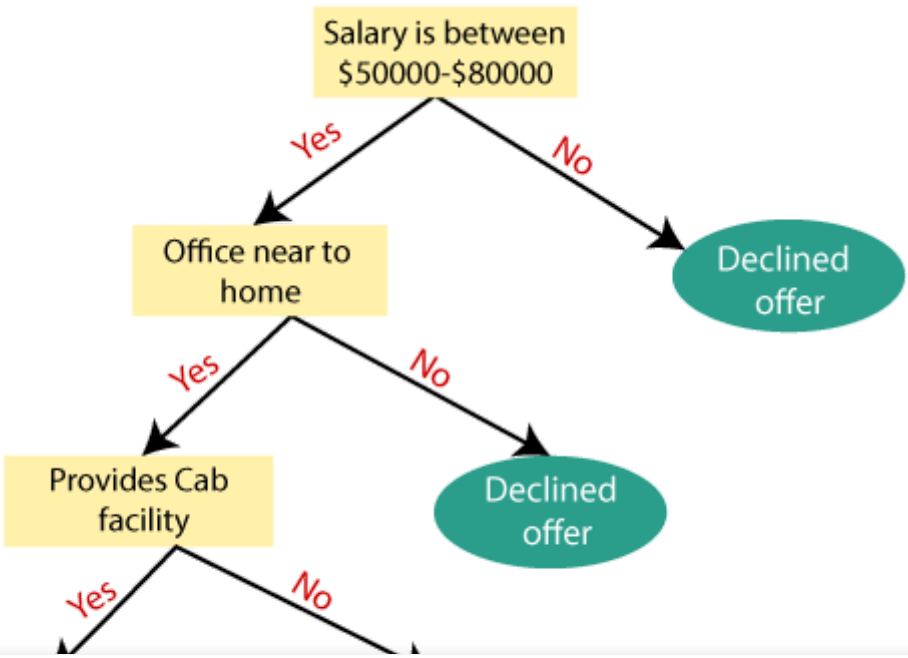
Decision Tree is one of the most commonly used, practical approaches for supervised learning. It can be used to solve both Regression and Classification tasks with the latter being put more into practical application.

It is a tree-structured classifier with three types of nodes. The ***Root Node*** is the initial node which represents the entire sample and may get split further into further nodes. The ***Interior Nodes*** represent the features of a data set and the branches represent the decision rules. Finally, the ***Leaf Nodes*** represent the outcome. This algorithm is very useful for solving decision-related problems.



Source

With a particular data point, it is run completely through the entirely tree by answering *True/False* questions till it reaches the leaf node. The final prediction is the average of the value of the dependent variable in that particular leaf node. Through multiple iterations, the Tree is able to predict a proper value for the data point.



The above diagram is a representation for the implementation of a Decision Tree algorithm. Decision trees have an advantage that it is easy to understand, lesser data cleaning is required, non-linearity does not affect the model's performance and the number of hyper-parameters to be tuned is almost null. However, it may have an over-fitting problem, which can be resolved using the ***Random Forest*** algorithm which will be explained in the next article.

In this example, we will go through the implementation of ***Decision Tree Regression***, in which we will predict the revenue of an ice cream shop based on the temperature in an area for 500 days.

Problem Analysis

In this data, we have one independent variable *Temperature* and one independent variable *Revenue* which we have to predict. In this problem, we have to build a Decision Tree Regression Model which will study the correlation between the Temperature and Revenue of the Ice Cream Shop and predict the revenue for the ice cream shop based on the temperature on a particular day.

Step 1: Importing the libraries

The first step will always consist of importing the libraries that are needed to develop the ML model. The ***NumPy***, ***matplotlib*** and the ***Pandas libraries*** are imported.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Step 2: Importing the dataset

In this step, we shall use pandas to store the data obtained from my github repository and store it as a Pandas DataFrame using the function '*pd.read_csv*'. In this, we assign the independent variable (X) to the '*Temperature'* column and the dependent variable (y) to the '*Revenue'* column.

```
dataset = pd.read_csv('https://raw.githubusercontent.com/mk-
```

```
dataset.head(5)

>>

Temperature   Revenue
24.566884     534.799028
26.005191     625.190122
27.790554     660.632289
20.595335     487.706960
11.503498     316.240194
```

Step 3: Splitting the dataset into the Training set and Test set

In the next step, we have to split the dataset as usual into the *training set* and the *test set*. For this we use `test_size=0.05` which means that 5% of 500 data rows (*25 rows*) will only be used as test set and the remaining *475 rows* will be used as training set for building the model.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
= 0.05)
```

Step 4: Training the Decision Tree Regression model on the training set

We import the `DecisionTreeRegressor` class from `sklearn.tree` and assign it to the variable '***regressor***'. Then we fit the X_train and the y_train to the model by using the `regressor.fit` function. We use the `reshape(-1,1)` to reshape our variables to a single column vector.

```
# Fitting Decision Tree Regression to the dataset
from sklearn.tree import DecisionTreeRegressor
regressor = DecisionTreeRegressor()
regressor.fit(X_train.reshape(-1,1), y_train.reshape(-1,1))
```

Step 5: Predicting the Results

In this step, we predict the results of the test set with the model trained on the training

Step 6: Comparing the Real Values with Predicted Values

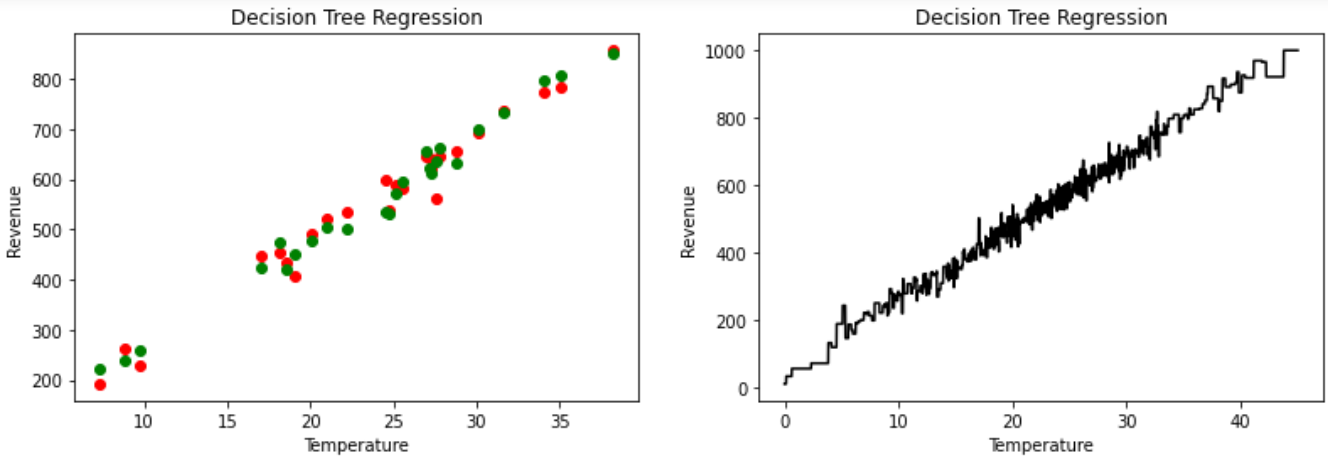
In this step, we shall compare and display the values of y_test as '**Real Values**' and y_pred as '**Predicted Values**' in a Pandas dataframe.

```
df = pd.DataFrame({'Real Values':y_test.reshape(-1), 'Predicted
Values':y_pred.reshape(-1)})
df

>>
Real Values    Predicted Values
448.325981     425.265596
535.866729     500.065779
264.123914     237.763911
691.855484     698.971806
587.221246     571.434257
653.986736     633.504009
538.179684     530.748225
643.944327     660.632289
771.789537     797.566536
644.488633     654.197406
192.341996     223.435016
491.430500     477.295054
781.983795     807.541287
432.819795     420.966453
623.598861     612.803770
599.364914     534.799028
856.303304     850.246982
583.084449     596.236690
521.775445     503.084268
228.901030     258.286810
453.785607     473.568112
406.516091     450.473207
562.792463     634.121978
642.349814     621.189730
737.800824     733.215828
```

From the above values, we infer that the model is able to predict the values of the y_test with a good accuracy.

Step 7: Visualising the Decision Tree Regression Results

Temperature vs Revenue(Decision Tree Regression)

In this graph, the Real values are plotted with "***Red***" color and the Predicted values are plotted with "***Green***" color. The plot of the Decision Tree Regression model is also drawn in "***Black***" color.

I am attaching a link of my github repository where you can find the Google Colab notebook and the data files for your reference.

I do hope that I have been able to explain the ML code for building a Decision Tree Regression model with an example.

You can also find the explanation of the program for other Regression models below:

- Simple Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- Support Vector Regression
- Decision Tree Regression
- Random Forest Regression

We will come across the more complex models of Regression, Classification and Clustering in the upcoming articles. Till then, Happy Machine Learning!