Arif R Follow
May 2, 2020 · 5 min read

Regression in Decision Tree — A Step by Step CART (Classification And Regression Tree)

Decision Tree Algorithms — Part 3

1. Introduction

In previous learning has been explained about The Basics of Decision Trees and A Step by Step Classification in CART, This section will explain A Step by Step Regression in CART.

As has been explained, Decision Trees is the non-parametric supervised learning approach. In addition to classification with continuous data on the target, we also often find cases with discrete data on the target called regression. In the regression, the simple

For regression trees, two common impurity measures are:

- Least squares. This method is similar to minimizing least squares in a linear model. Splits are chosen to minimize the residual sum of squares between the observation and the mean in each node.
- Least absolute deviations. This method minimizes the mean absolute deviation from the median within a node. The advantage of this over least squares is that it is not as sensitive to outliers and provides a more robust model. The disadvantage is in insensitivity when dealing with data sets containing a large proportion of zeros [1].

Note : mostly people implement regression case with scikit-learn library, Based on documentation, scikit-learn uses an optimised version of the CART algorithm

2. How Does CART Work in Regression with one predictor?

CART in classification cases uses Gini Impurity in the process of splitting the dataset into a decision tree. On the other hand CART in regression cases uses least squares, intuitively splits are chosen to minimize the **residual sum of squares** between the observation and the mean in each node. Mathematically, we can write residual as follow

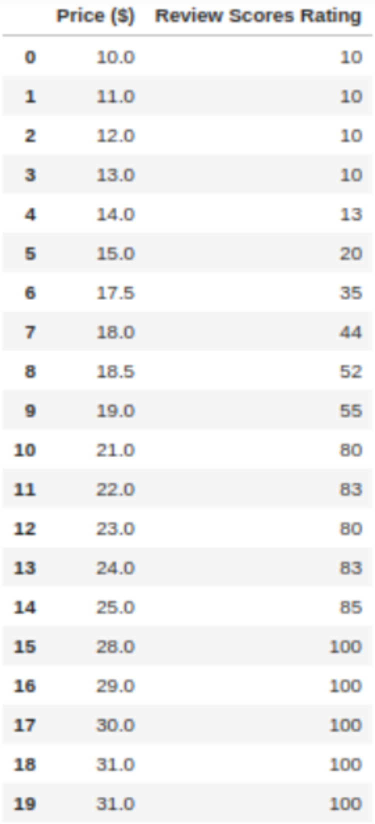
$$\varepsilon_i = y_i - \hat{y}_i \quad (1.0)$$

Mathematically, we can write **RSS (residual sum of squares)** as follow

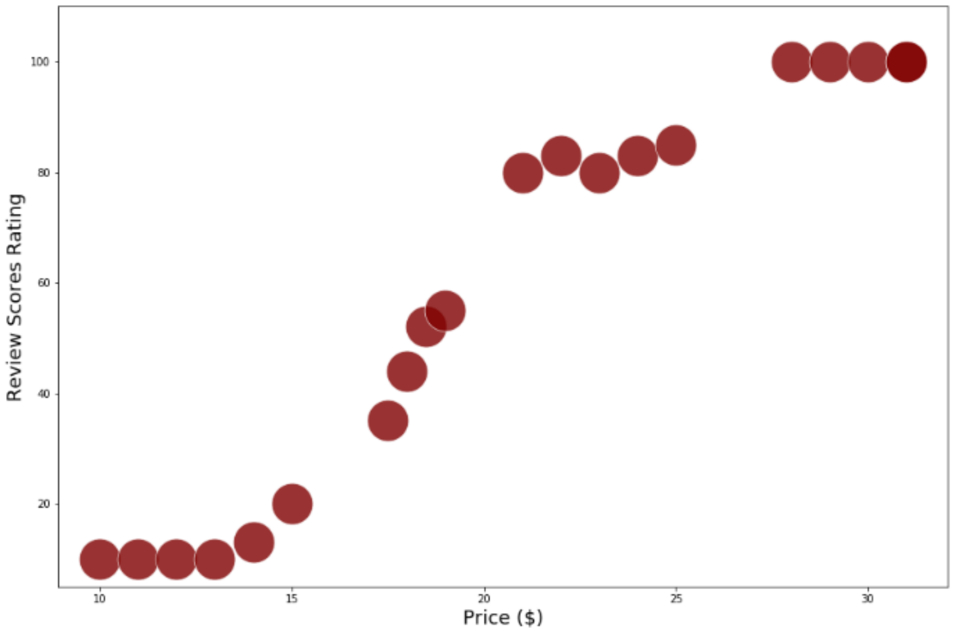
$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \quad (2.0)$$

$$RSS = \varepsilon_1^2 + \varepsilon_2^2 + .. + \varepsilon_n^2 \quad (2.1)$$

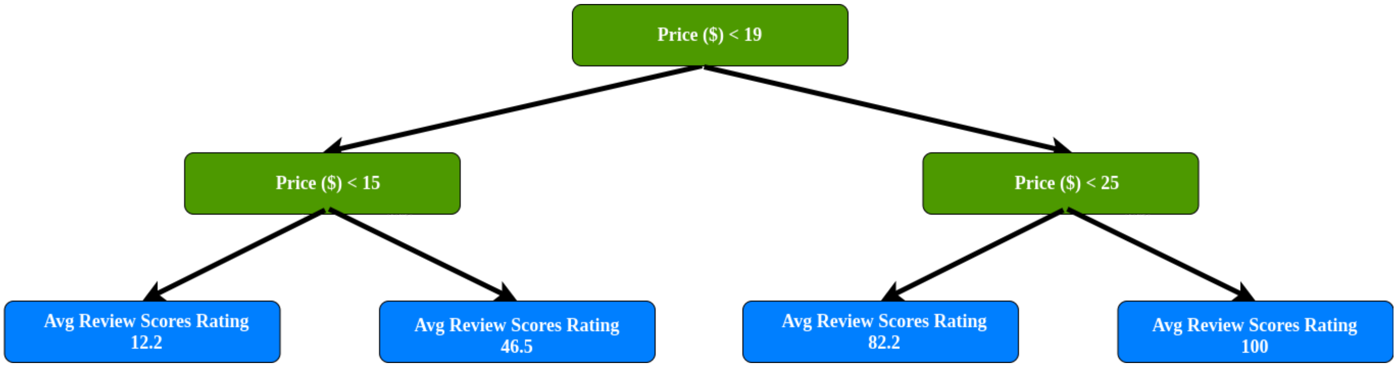
In order to find out the "best" split, we must minimize the RSS



	Price ($)	Review Scores Rating
0	10.0	10
1	11.0	10
2	12.0	10
3	13.0	10
4	14.0	13
5	15.0	20
6	17.5	35
7	18.0	44
8	18.5	52
9	19.0	55
10	21.0	80
11	22.0	83
12	23.0	80
13	24.0	83
14	25.0	85
15	28.0	100
16	29.0	100
17	30.0	100
18	31.0	100
19	31.0	100



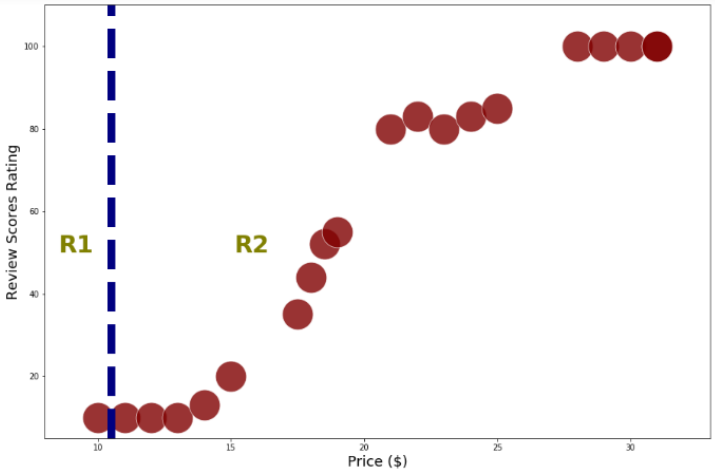
The decision tree as follow

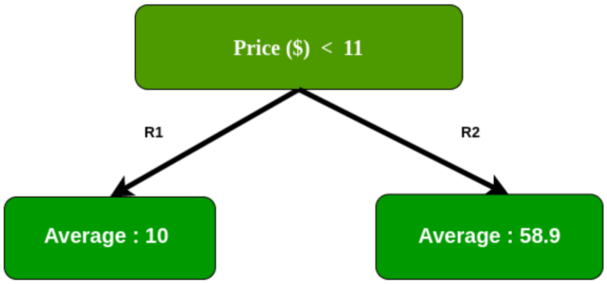

Price ($) < 19
Price ($) < 15
Price ($) < 25
Avg Review Scores Rating 12.2
Avg Review Scores Rating 46.5
Avg Review Scores Rating 82.2
Avg Review Scores Rating 100

2.2 How does CART process the splitting of the dataset (predictor =1)

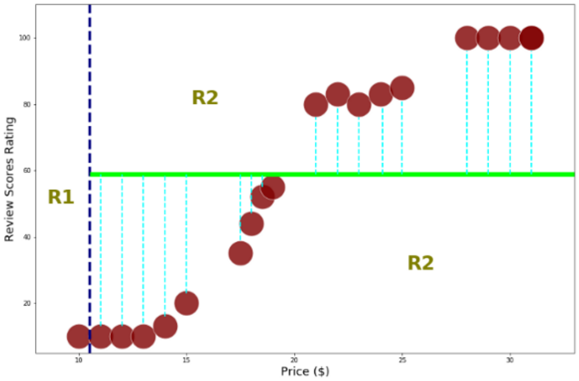
As mentioned before, **In order to find out the "best" split, we must minimize the RSS.** first, we calculate **RSS** by split into two regions, start with index 0

Start within index 0





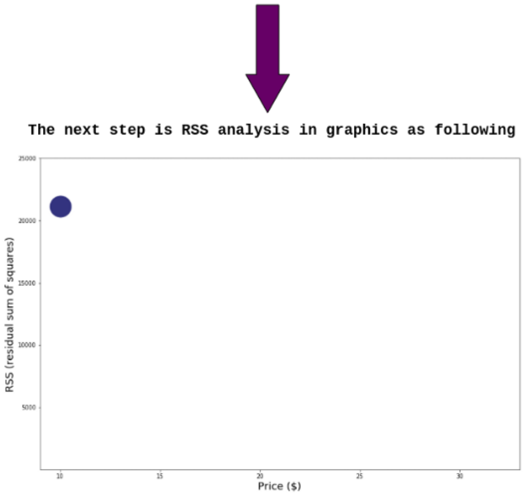
The data already split into two regions, we add up the squared residual for every index data. furthermore we calculate **RSS** each node using equation 2.0



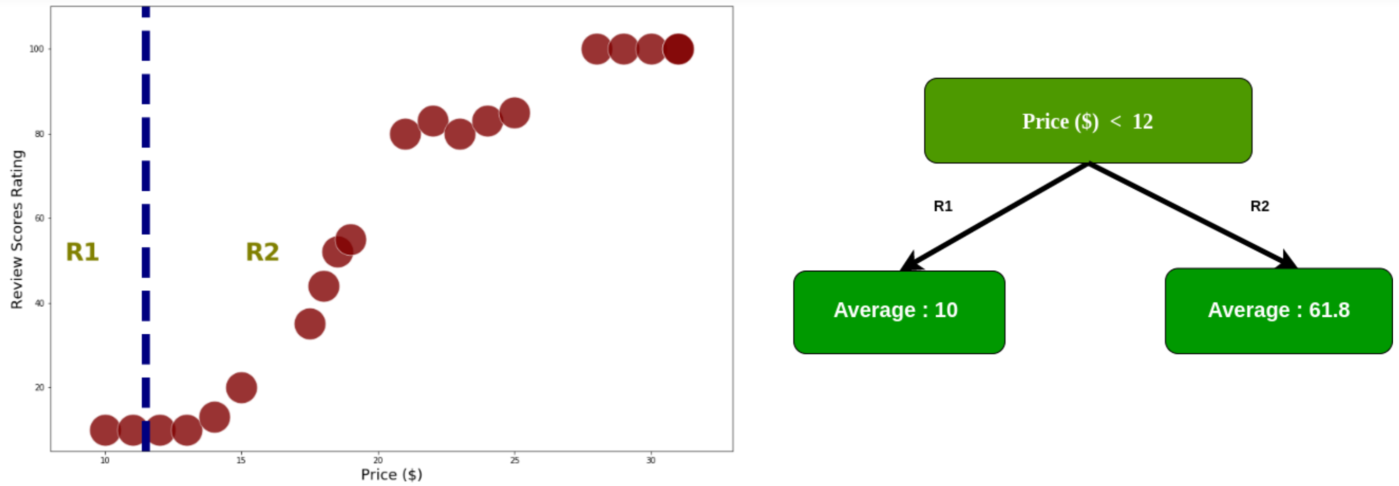
$$\varepsilon = \sum_{i=1}^{n} (actual\ value\ -\ average\ value\ in\ each\ region)^2$$

$$RSS = \varepsilon_1^2 + \varepsilon_2^2 + .. + \varepsilon_n^2$$

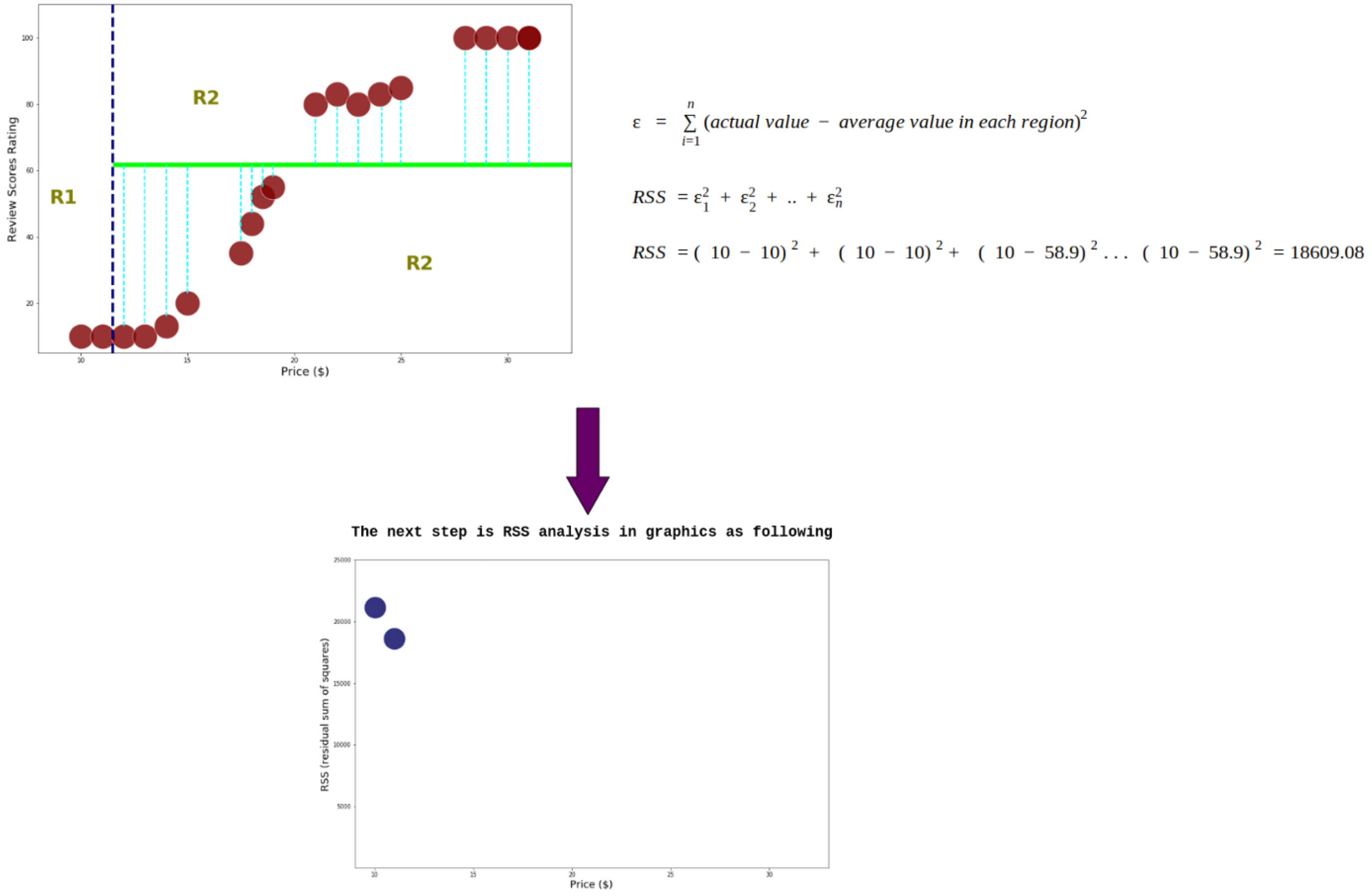
$$RSS = (\ 10 - 10)^2 + (\ 10 - 58.9)^2 + (\ 10 - 58.9)^2 ... (\ 10 - 58.9)^2 = 21139.78$$


The next step is RSS analysis in graphics as following

Start within index 1



after the data is divided into two regions then calculate **RSS** each node using equation 2.0



$$\varepsilon = \sum_{i=1}^{n} (\textit{actual value} - \textit{average value in each region})^2$$

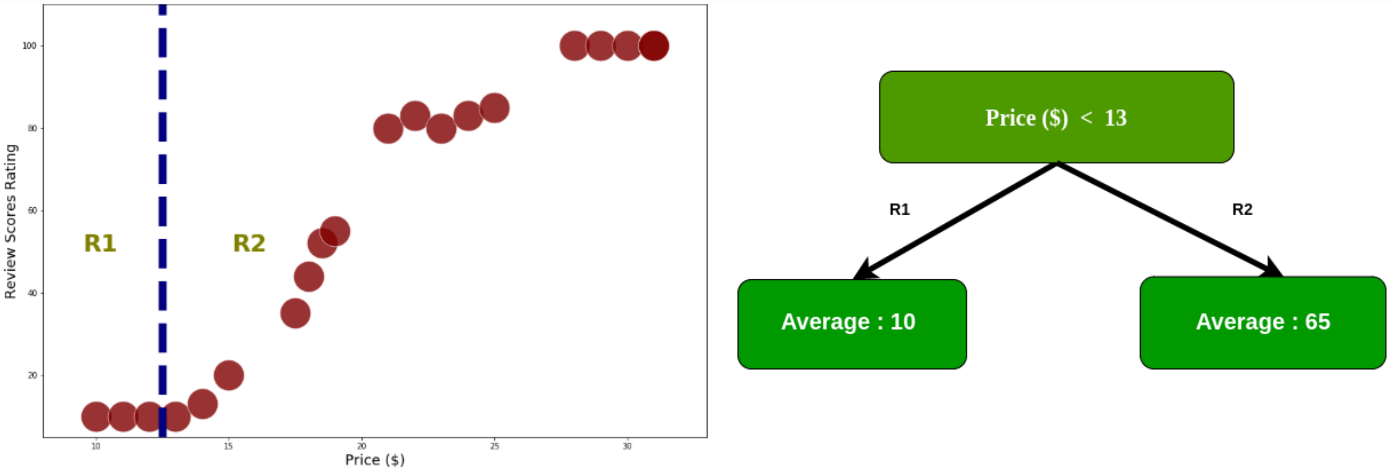
$$RSS = \varepsilon_1^2 + \varepsilon_2^2 + .. + \varepsilon_n^2$$

$$RSS = (\ 10 - 10)^2 + (\ 10 - 10)^2 + (\ 10 - 58.9)^2 \ldots (\ 10 - 58.9)^2 = 18609.08$$

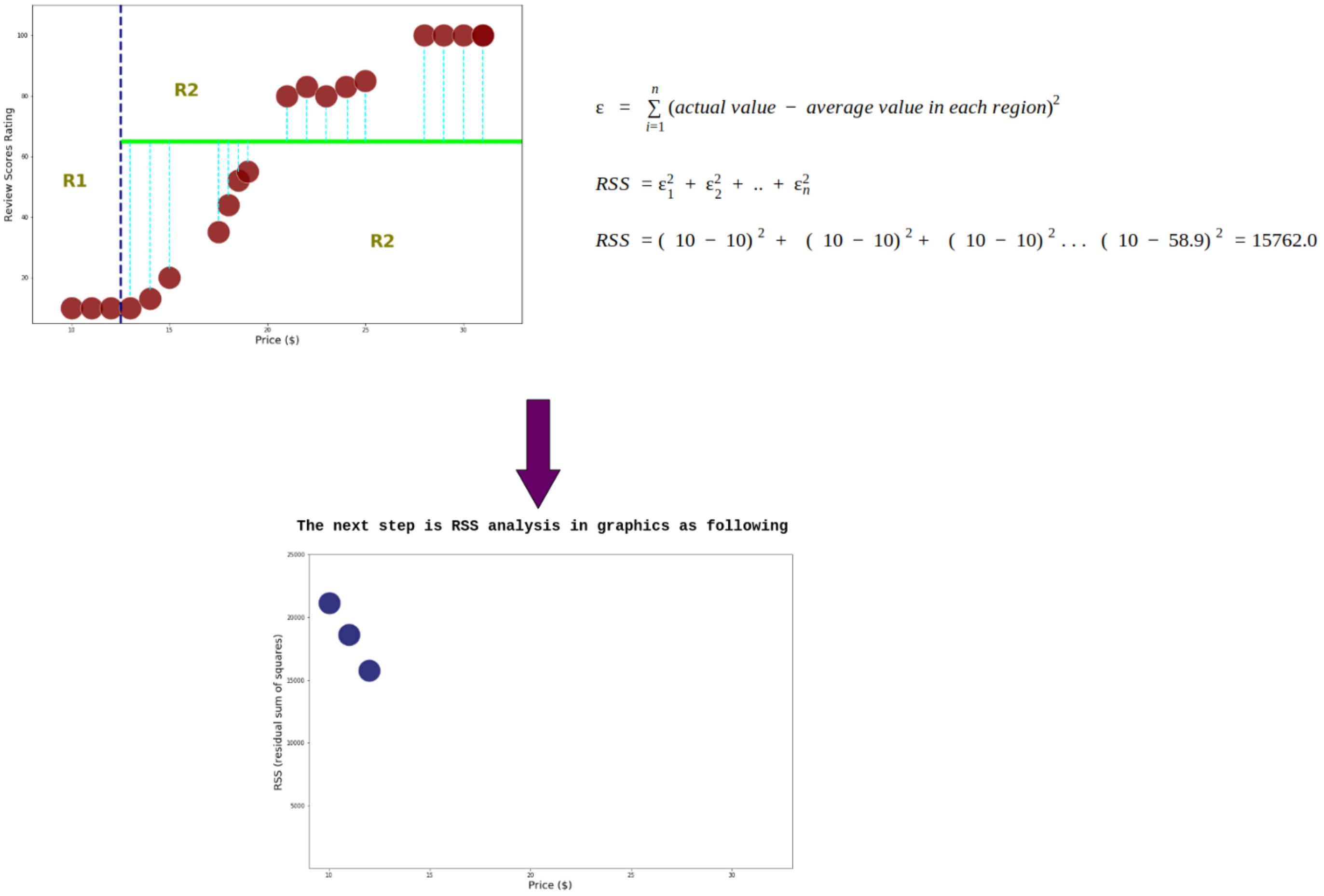
The next step is RSS analysis in graphics as following

Start within index 2

calculate RSS by split into two regions within index 2



calculate **RSS** each node



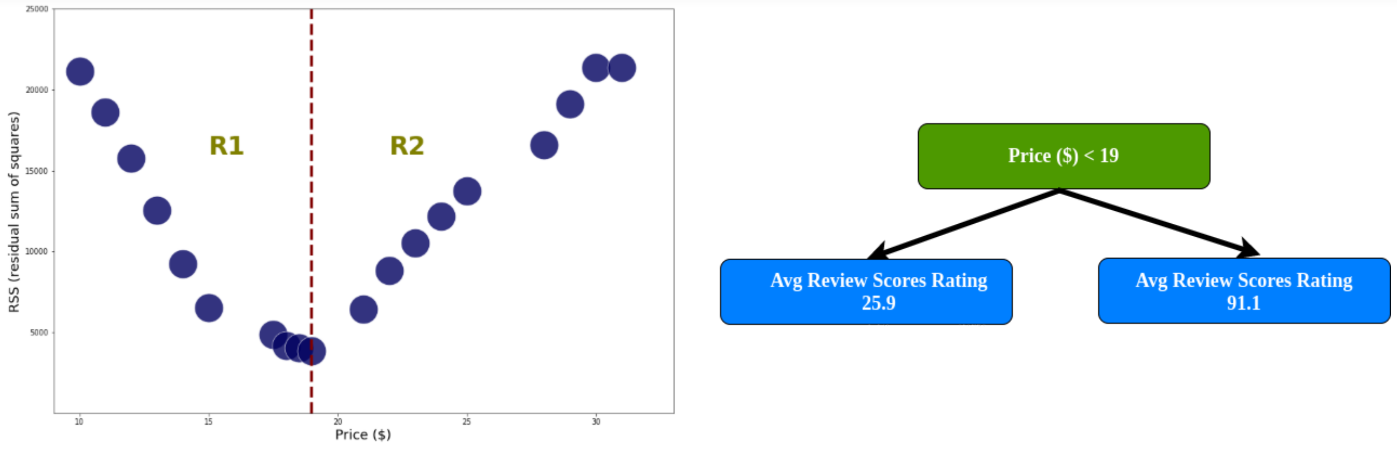
$$\varepsilon = \sum_{i=1}^{n} (actual\ value\ -\ average\ value\ in\ each\ region)^2$$

$$RSS = \varepsilon_1^2 + \varepsilon_2^2 + .. + \varepsilon_n^2$$

$$RSS = (\ 10 - 10)^2 + (\ 10 - 10)^2 + (\ 10 - 10)^2 ... (\ 10 - 58.9)^2 = 15762.0$$

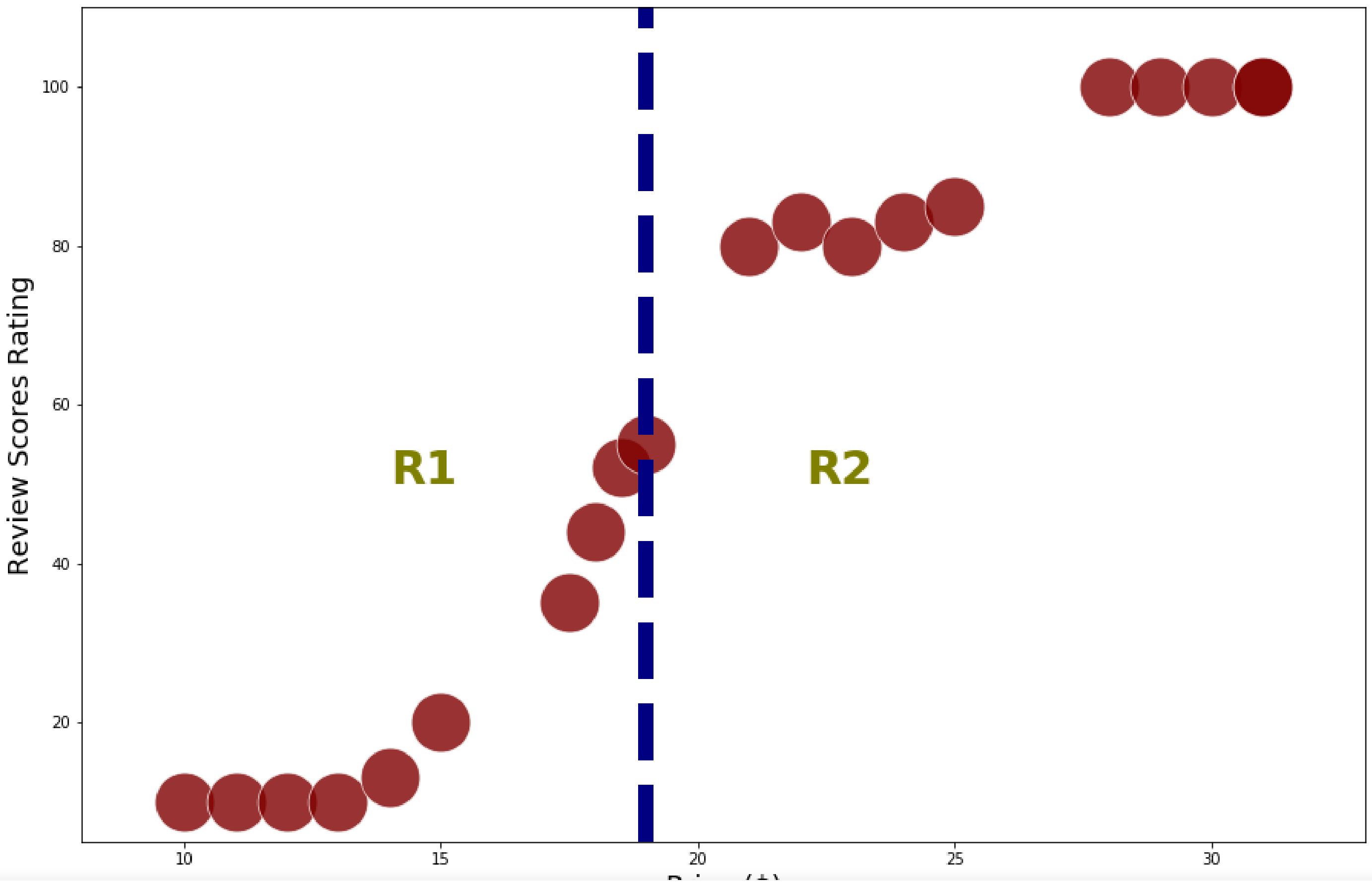
The next step is RSS analysis in graphics as following

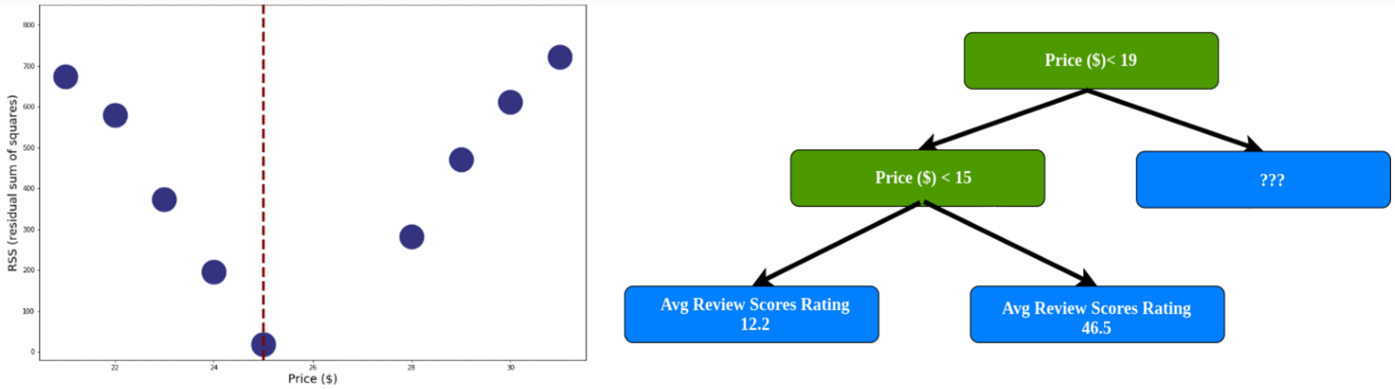
This process continues until the calculation of RSS in the last index



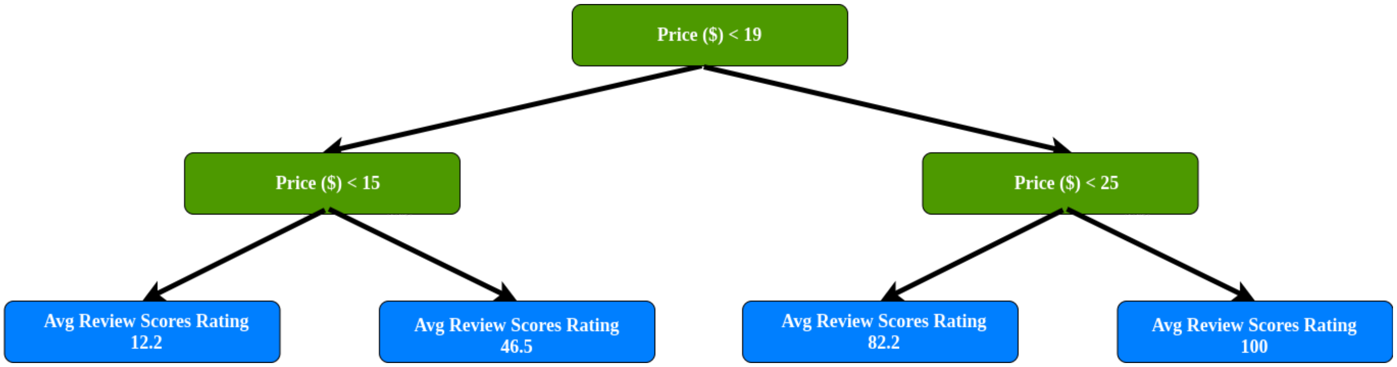
Price with threshold 19 has a smallest RSS, in R1 there are 10 data within price < 19, so we'll split the data in R1. In order to avoid overfitting, we define the minimum data for each region >= 6. If the region has less than 6 data, the split process in that region stops.

Split the data with threshold 19



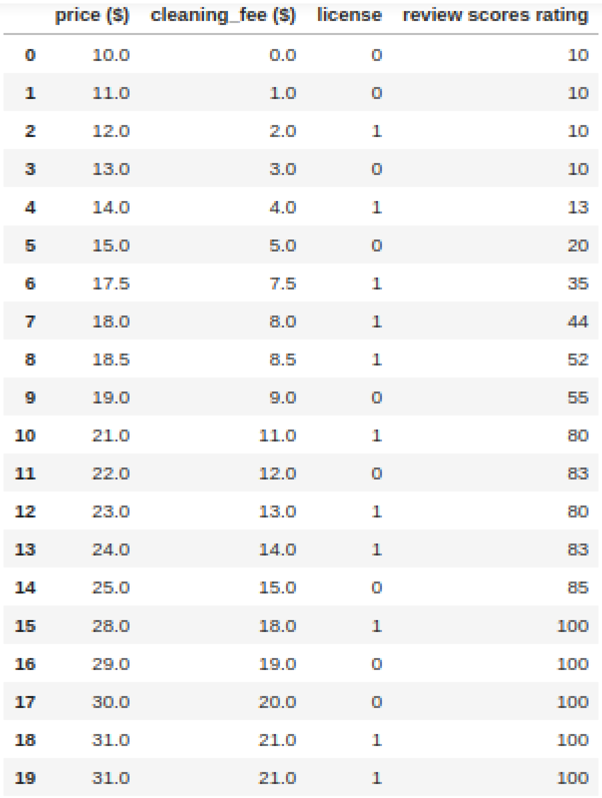


Do the same thing on the right branch, so the end result of a tree in this case is



2.3 How does CART process the splitting of the dataset (predictor > 1)

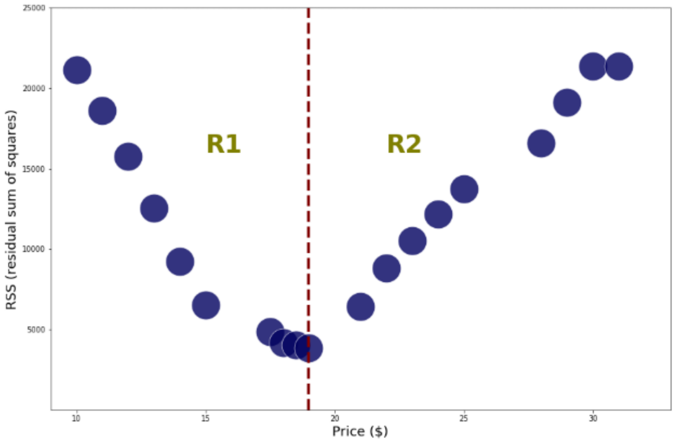
This simulation uses a **dummy data** as following

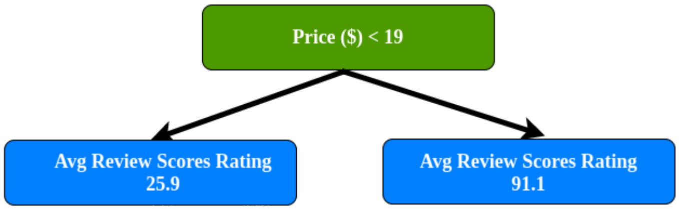


	price ($)	cleaning_fee ($)	license	review scores rating
0	10.0	0.0	0	10
1	11.0	1.0	0	10
2	12.0	2.0	1	10
3	13.0	3.0	0	10
4	14.0	4.0	1	13
5	15.0	5.0	0	20
6	17.5	7.5	1	35
7	18.0	8.0	1	44
8	18.5	8.5	1	52
9	19.0	9.0	0	55
10	21.0	11.0	1	80
11	22.0	12.0	0	83
12	23.0	13.0	1	80
13	24.0	14.0	1	83
14	25.0	15.0	0	85
15	28.0	18.0	1	100
16	29.0	19.0	0	100
17	30.0	20.0	0	100
18	31.0	21.0	1	100
19	31.0	21.0	1	100

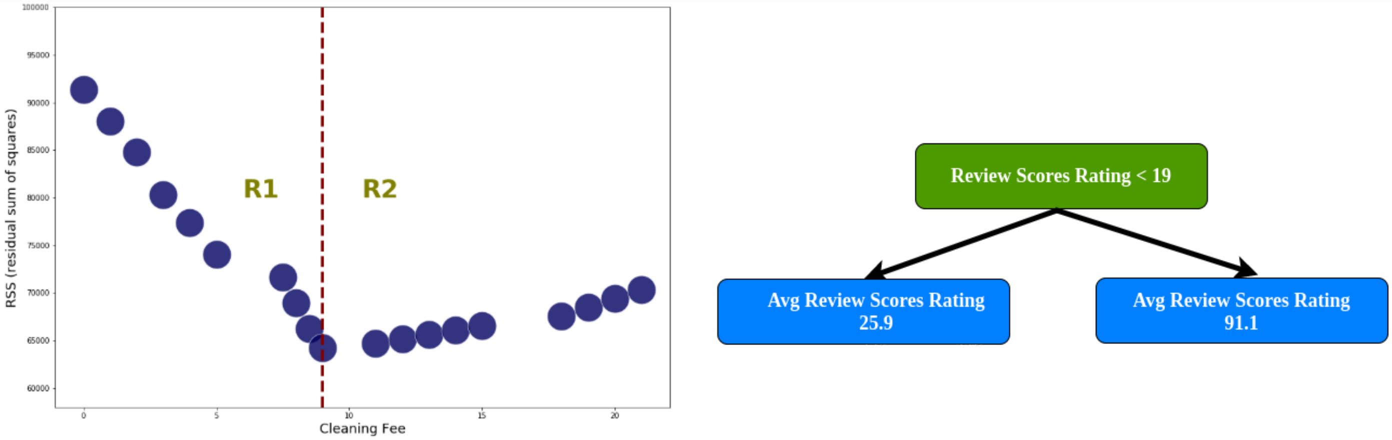
Find out the minimum RSS each predictor

Price with RSS = 3873.79

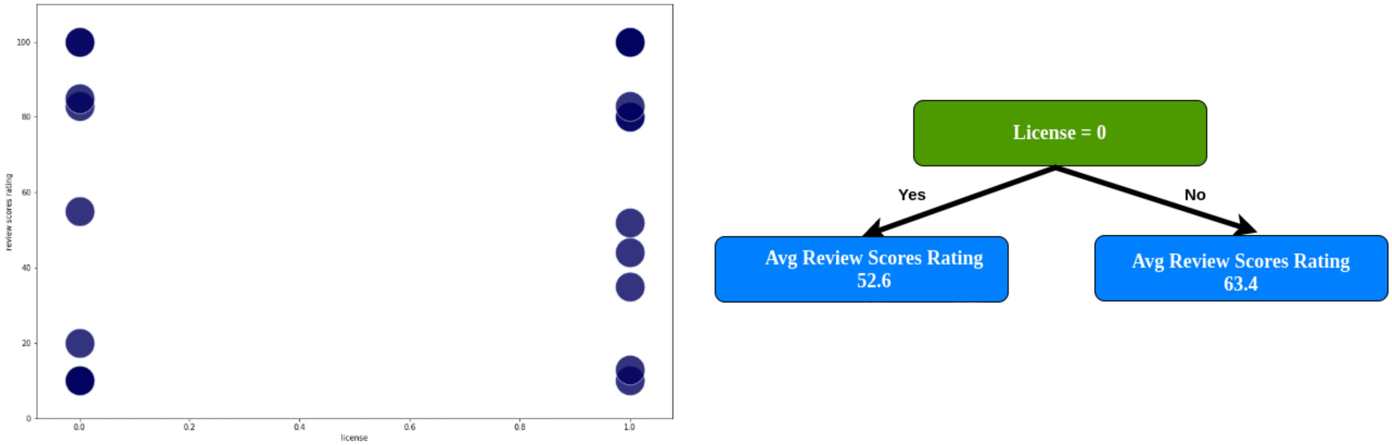




Cleaning fee with RSS = 64214.8



There is only one threshold in License, 1 or 0. So we use that threshold to calculate RSS. **License with RSS = 11658.5**



We already have RSS every predictor, compare RSS for each predictor, and find the lowest RSS value. If we analyze, License has the lowest value so it becomes root.

The next step can follow the intuition of the Classification in Decision Tree, in the case of classification calculates Gini Impurity, while in the case of regression calculates the minimum RSS. So this is a challenge for you if want to calculate RSS to the end :)

About Me

I'm a Data Scientist, Focus on Machine Learning and Deep Learning. You can reach me from Medium and Linkedin

1. Introduction to Statistical Learning

2. Ecological Informatics — Classification and Regression Trees

3. *Adapted from YouTube Channel of “StatQuest with Josh Stamer”*