Principle of operation of a decisive tree

1 grudnia 2022

1 Overview

To start from the very beginning: having data of form $(y, \mathbf{x}) \in \mathbb{R}^{n+1}$ we want to find a function $f : \mathbb{R}^n \to \mathbb{R}$ that minimizes somehow the error $|f(\mathbf{x} - y)|$. I suppose that the exact criterion we will be using, will be RSS. Okay, and we want to find function f with a method of a decisive tree. Firstly we will try to make the tree as accurate as possible, later we will change the value of hyperparameters to avoid overfitting.

2 How to find variables and points to split nodes?

I think that variable choice should be done each time by choosing a variable of greatest variance. It will however need much operational power.
When it comes to choosing a threshold i suggest we use some brutal force. First we use Chebyshev Inequality to narrow the problem to the area with probably something like 90 % of points (or here we can also brute force). Later we choose three points - edges of the area and the very mean of that. For each point we calculate RSS (with $f =$ mean) for both sides of the point. Then we go to that half (mean or rather median?) having greater RSS. We divide it by half and again calculate RSS of 'left' part and RSS of 'right' part. Iterate. So, after as little as 12 calculations we have tested over a 1000 points. It should be enough, but if we want to increas accuracy, we can make 22 calculations (it will make algorithm slow no more than two times) - if dataset consists of less than a million points, then we will examine them all.

3 How to store a decisive tree?

Isn't it actually quite simple? It should be vector of points of form

$(variable,\ threshold,\ type,\ leftValue,\ rightValue,\ leftVariance,\ rightVariance,\ leftVariable,\ rightVariable)$

and then, if type is 'numeric', it would simply be the the end of a node - so such a point would save information for two leafes. Otherwise, with type qual to 'reference', this point would be an instruction to what point of vector go in the next turn. Of course there is a need to have a 'mixed type'

4 Hyperparameters

- **depth** - main (?) stopping criterion
- **minimum leaves on a node** - suppose it should be default 5, but with option of a change
- **how many points to test** - while wanting to find a perfect cut, how many points do we want to test?
- **number of leaves** -another possible stopping criterion