

Regression costs for decision trees

John Alan McDonald

2018-01-19

The purpose of this document is work thru an alternative to L_2 cost that is a bit more efficient to compute, and gives the same results when choosing split predicates in decision tree growing.

I Greedy decision trees

A general binary decision tree consists of

- internal *split* nodes, each containing a predicate that determines whether a record goes to the left or right child of that node.
- terminal *leaf* nodes, each containing a leaf model function whose value is the tree's prediction for any record that ends up in that node.

Greedy split optimization — choose the best out of all feasible splits, and repeat on the resulting child nodes until there are no feasible splits — is the most common way of growing decision trees. It depends on several things:

1. A cost function c used to define 'best'.
2. An enumeration of splits to consider. Pure greedy splitting considers all 'feasible' splits on all attributes.
 - (a) For categorical attributes, that, in general, means considering every partition of the categories into 2 subsets. However, for some important cost functions (eg 2-class Gini, L_2), it can be shown that the optimal split can be found by

sorting the categories by the corresponding score function (eg the response mean for L_2 cost), and then considering only splits by score.

- (b) For numerical attributes, the most general split would come from treating the distinct values like the categories of a categorical variable. However, no one does that, mostly because there are usually too many distinct values. Instead, only splits by \leq vs $>$ one of the distinct values are considered.
- 3. A feasibility test that determines whether a given split on a attribute is allowed. The most common case here is to require both children of the split contain some minimum number of training records.

2 Cost functions for L_2 numerical regression

Let $\mathcal{T} = \{(y, \mathbf{x})\}$ be the training data in the node to be split. It is a set of pairs of predictor record \mathbf{x} and ground truth response y , where $y \in \mathbb{R}$ for numerical regression. We are considering splits on some particular predictor field x_k , which might be numerical or categorical.

The cost function for L_2 regression is the sum of squared deviations from the mean:

$$L_2(\mathcal{T}) = \sum_{y \in \mathcal{T}} (y - \bar{y}_{\mathcal{T}})^2, \text{ where } \bar{y}_{\mathcal{T}} = \frac{1}{\#\mathcal{T}} \sum_{y \in \mathcal{T}} y.$$

Note that computing this *accurately*, in an online fashion, for moderate $\#\mathcal{T}$, the number of records in \mathcal{T} , allowing for the updating/downdating needed for fast split optimization, requires some care.

However, a little bit of algebra will let us use a simpler alternative to get the same splits.

Any split partitions the training y -values $\mathcal{T} = \{y\}$ into left and right subsets: $\mathcal{T} = \mathcal{L} \uplus \mathcal{R}$. The split cost is:

$$\begin{aligned}
c(\mathcal{L}, \mathcal{R}) &= L_2(\mathcal{L}) + L_2(\mathcal{R}) \\
&= \sum_{y \in \mathcal{L}} (y - \bar{y}_{\mathcal{L}})^2 + \sum_{y \in \mathcal{R}} (y - \bar{y}_{\mathcal{R}})^2 \\
&= \sum_{y \in \mathcal{L}} [y^2 - 2\bar{y}_{\mathcal{L}}y + \bar{y}_{\mathcal{L}}^2] + \sum_{y \in \mathcal{R}} [y^2 - 2\bar{y}_{\mathcal{R}}y + \bar{y}_{\mathcal{R}}^2] \\
&= \sum_{\mathcal{L} \uplus \mathcal{R}} y^2 - \frac{(\sum_{\mathcal{L}} y)^2}{\#\mathcal{L}} - \frac{(\sum_{\mathcal{R}} y)^2}{\#\mathcal{R}}
\end{aligned}$$

Since $\sum_{\mathcal{L} \uplus \mathcal{R}} y^2$ doesn't depend on the split, minimizing $c(\mathcal{L}, \mathcal{R})$ is equivalent to minimizing $-\left[\frac{(\sum_{\mathcal{L}} y)^2}{\#\mathcal{L}} + \frac{(\sum_{\mathcal{R}} y)^2}{\#\mathcal{R}}\right]$, so we can use $\frac{-(\sum_{\mathcal{T}} y)^2}{\#\mathcal{T}}$ as our cost function in split optimization.

3 Cost functions for L_2 vector-valued regression

Let $\mathcal{T} = \{(y, \mathbf{x})\}$ be the training data in the node to be split. Here the ground truth response y is a vector, $y \in \mathbb{R}^m$, rather than a single number.

The cost function for L_2 vector-valued regression is the sum of squared L_2 distances from the mean vector:

$$\begin{aligned}
L_2(\mathcal{T}) &= \sum_{y \in \mathcal{T}} \|y - \bar{y}_{\mathcal{T}}\|_2^2 \\
&= \sum_{y \in \mathcal{T}} \sum_{i=0}^{m-1} (y_i - \bar{y}_{i\mathcal{T}})^2
\end{aligned}$$

Following the same reasoning as in section 2, we get for a simpler cost:

$$c(\mathcal{L}, \mathcal{R}) = - \left[\frac{\sum_{i=0}^{m-1} (\sum_{\mathcal{L}} y_i)^2}{\#\mathcal{L}} + \frac{\sum_{i=0}^{m-1} (\sum_{\mathcal{R}} y_i)^2}{\#\mathcal{R}} \right]$$

4 Typesetting

This document was typeset using MikTeX 2.9 [3] and T_EXworks 0.6.1 [2] on Windows 10. I used arara [1] to run xelatex, biber, xelatex, and xelatex. An alternative is to call these 4 commands by hand.

I believe only MikTeX and T_EXworks are Windows specific; the actual typesetting tools should be usable on Linux and MacOS as well.

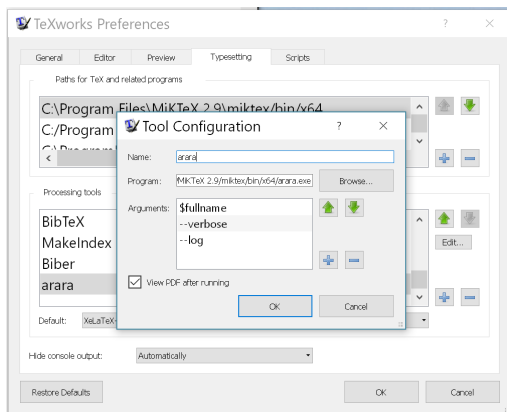


Figure 4.1: Configuring T_EXworks for arara.

References

- [1] Paulo Roberto Massa CEREDA, Marco DANIEL, Brent LONGBOROUGH, and Nicola Louise Cecilia TALBOT.
arara: The cool T_EX automation tool.
URL: <https://github.com/cereda/arara> (visited on 2017-03-13).
- [2] Jonathan KEW and Stefan LÖFFLER.
T_EXworks: A simple interface for working with T_EX documents.
URL: <https://github.com/texworks/> (visited on 2017-03-13).
- [3] Christian SCHENK.
MikTeX: typesetting beautiful documents.
URL: <https://miktex.org/> (visited on 2017-03-13).