

Lattice Regression

Author: Garcia, Gupta Type: Paper URL: <https://papers.nips.cc/paper/3694-lattice-regression>

1. Introduction

In high throughput regression settings, the cost of evaluating a test sample is just as important as the accuracy of the regression. For functions with a known and bounded domain, a standard *efficient* regression approach is to store a regular lattice of function values spanning the domain, then interpolate each test sample from the lattice vertices that surround it. Evaluating the lattice is then independent in the size of the original training data, but scales exponentially in the dimension of the input space.

Here, a solution is proposed that is termed *lattice regression*, that jointly estimates all of the lattice outputs by *minimizing the regularized interpolation error on the training data*.

2. Lattice Regression

The motivation behind the proposed lattice regression is to jointly choose outputs for lattice nodes that interpolate the training data accurately. The key here is that the linear interpolation operation can be directly inverted to solve for the node outputs that minimize the squared error of the training data. Two forms of regularization are added: Laplacian regularization and a global bias.

2.1 Empirical Risk

Training data is from a bounded input space $\mathcal{D} \subset \mathbb{R}^d$ and output space $\mathcal{Y} \subset \mathbb{R}^p$. Consider a lattice consisting of m nodes where $m = \prod_{j=1}^d m_j$ and m_j is the number of nodes along dimension j . Each node consists of an input-output pair $(a_i \in \mathbb{R}^d, b_i \in \mathbb{R}^p)$ and the inputs $\{a_i\}$ form a grid that contains \mathcal{D} within its convex hull. Let A be the $d \times m$ matrix $A = [a_1, \dots, a_m]$ and B be the $p \times m$ matrix $B = [b_1, \dots, b_m]$.

For any $x \in \mathcal{D}$, there are $q = 2^d$ nodes in the lattice that form a cell containing x ; denote the indices of these nodes by $c_1(x), \dots, c_q(x)$. Here the interpolation is restricted to linear interpolation of the surrounding node outputs $\{b_{c_1(x)}, \dots, b_{c_q(x)}\}$, i.e. $\hat{f}(x) = \sum_i w_i(x) b_{c_i(x)}$.

The lattice output B^* that minimize the total squared- L_2 distortion between the lattice-interpolated training outputs BW and the given training outputs Y are

$$B^* = \arg \min_{\{B\}} \text{tr} \left(\text{Big} (\text{Big}(BW - Y) \text{Big} (BW - Y)^T \text{Big}) \right)$$

where $W = [W(x_1), \dots, W(x_n)]$ where $W(x)$ is the $m \times 1$ sparse vector with $c_j(x)$ th entry $w_j(x)$ for $j = 1, \dots, 2^d$ and zeros elsewhere.

2.2 Laplacian Regularization

Is used to penalize the average squared difference of the output on adjacent lattice nodes.

The graph Laplacian of the lattice is fully defined by the $m \times m$ lattice adjacency matrix E where $E_{ij} = 1$ for adjacent nodes and zeroes elsewhere.

2.3 Global Bias

To improve the ability to extrapolate and regularize towards trends in data, the global bias term is included. It penalizes the divergence of lattice node outputs from some global function $\tilde{f} : \mathbb{R}^d \rightarrow \mathbb{R}^p$.

It was hypothesized that the lattice regression performance would be better if the \tilde{f} was itself a good regression of the training data. Surprisingly, the experiments showed little difference in using an accurate, an inaccurate or no bias at all.

2.4 Lattice Regression Objective Function

Here a closed form exists. Please look in the [paper](#) for a detailed description.

3. Experiments

Were done on

- randomly generated functions
- real geospatial data
- real color management task

4. Conclusion

Simulations showed that lattice regression was statistically significantly better than the standard approach of first fitting a function then evaluating it at the lattice points. Surprisingly, although the motivation was computational efficiency, both the simulated and real-world data experiments showed that the proposed lattice regression can work better than state-of-the-art regression of test samples without a lattice.
