Sparse Convex Regression

1 Introduction

In the nonparametric regression problem

$$y = f(\boldsymbol{x}) + \epsilon,$$

we model $f(\boldsymbol{x})$ as a convex piecewise linear function consisting of K hyperplanes

$$f(\boldsymbol{x}) = \max_{k=1,2,\cdots,K} \{\alpha_k + \boldsymbol{x}^\top \boldsymbol{\beta}_k\}, \tag{1}$$

and the parameters can be estimated via the following optimization problem

$$\min_{\{\boldsymbol{\alpha}_{1:K}, \boldsymbol{\beta}_{1:K}\}} \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \max_{k=1,2,\cdots,K} \{\alpha_k + \boldsymbol{x}_i^\top \boldsymbol{\beta}_k\} \right)^2 + \lambda \|(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \cdots, \boldsymbol{\beta}_K)\|_{2,1}. \tag{2}$$

Here $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$ are n training points, and $\|\cdot\|_{2,1}$ enforces joint sparsity for automatic feature selection. Notice that if $K = 1$, the above problem reduces to LASSO regression [1]. An example of the function in (1) is plotted in Figure 1.

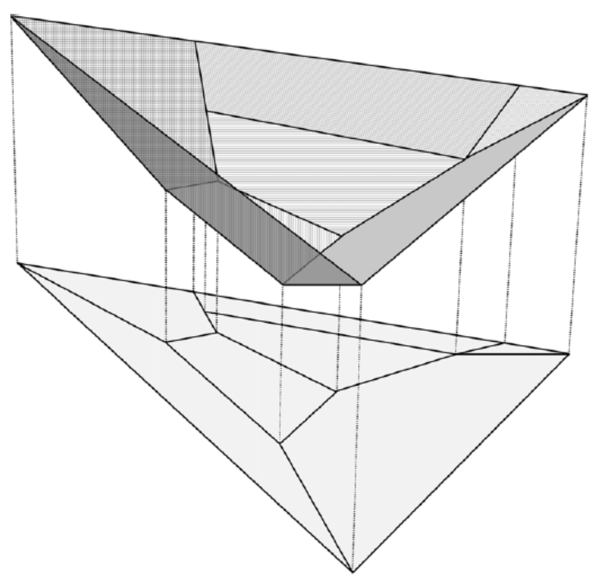

Figure 1: Example of a convex piecewise linear function [B. Williamsa, M. Eatonb and D. Breiningerc, 2011].

2 A Convex Formulation for $K = n$

At each point $\boldsymbol{x}_i$, we could construct a supporting hyperplane

$$f(\boldsymbol{x}) \geq f(\boldsymbol{x}_i) + \boldsymbol{\beta}_i^\top (\boldsymbol{x} - \boldsymbol{x}_i).$$

Hence we can define

$$f(\boldsymbol{x}) = \max_{i=1,2,\cdots,n} \{f(\boldsymbol{x}_i) + \boldsymbol{\beta}_i^\top (\boldsymbol{x} - \boldsymbol{x}_i)\}.$$

Notice that this function form is consistent with (1) by setting $\alpha_i = f(\boldsymbol{x}_i) - \boldsymbol{\beta}_i^\top \boldsymbol{x}_i$ and $K = n$. Then we can formulate a convex program [2] to estimate the function values $f(\boldsymbol{x}_{1:n}) \triangleq \boldsymbol{h}$ and the sub-gradients $\boldsymbol{\beta}_{1:n} \triangleq \boldsymbol{\beta}$:

$$\min_{\{\boldsymbol{h},\boldsymbol{\beta}\}} \frac{1}{2} \sum_{i=1}^{n} (y_i - h_i)^2 + \lambda \|\boldsymbol{\beta}\|_{2,1} \quad \text{s.t.} \quad h_j \geq h_i + \boldsymbol{\beta}_i^\top (\boldsymbol{x}_j - \boldsymbol{x}_i) \quad (\forall i, j). \tag{3}$$

An ADMM algorithm to solve the above convex program is derived in the Appendix.
As a toy example, we learn a convex piecewise linear function based on a few sample points on a curve $f(x) = x + x^{-1}$ $(x > 0)$. The result is plotted in Figure 2. Since there is no feature selection in this example, we set $\lambda = 0$. More experiments on high dimensional data with feature selection will be provided later.

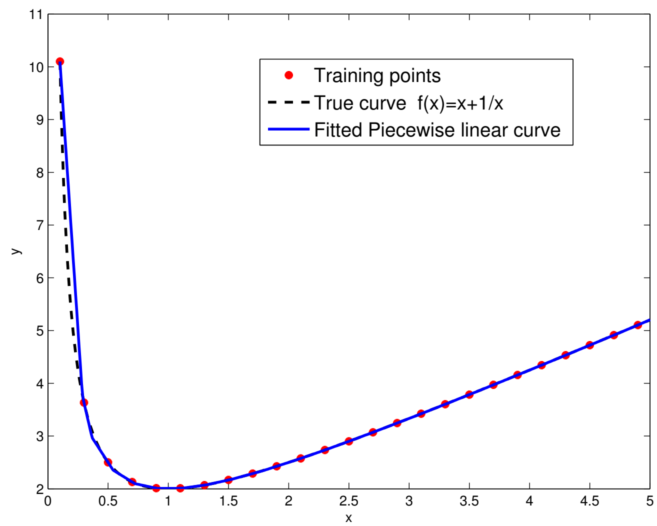

Figure 2: Convex piecewise linear fitting for a function $f(x) = x + x^{-1}$ $(x > 0)$.

3 A K-means Type Algorithm for $K < n$

We could use much fewer hyperplanes than n by partitioning the samples into 'clusters'. A K-means type algorithm was derived in [3] to simultaneous partition the samples and estimate the parameters in the convex regression problem. [4] also proposed a K-means type algorithm for the convex regression problem, the difference being that the number of partitions K changes adaptively via a splitting procedure. We adopt the same idea in [3] for our sparse convex regression problem in (2). First, each data sample is assigned to a hyperplane via

$$z_i = \arg \max_{k=1,2,\cdots,K} \{\alpha_k + \boldsymbol{x}_i^\top \boldsymbol{\beta}_k\} \quad (i = 1, 2, \cdots, n).$$

Here z_i denotes which hyperplane sample i belongs to. Second, since the max operator has been performed in the first step, we can solve the following (convex) joint sparse regression problem to find the hyperplanes:

$$\min_{\{\boldsymbol{\alpha}_{1:K},\boldsymbol{\beta}_{1:K}\}} \frac{1}{2} \sum_{k=1}^{K} \sum_{i:z_i=k} \left(y_i - (\alpha_k + \boldsymbol{x}_i^\top \boldsymbol{\beta}_k)\right)^2 + \lambda \|(\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \cdots, \boldsymbol{\beta}_K)\|_{2,1}$$

The above two steps are iterated until a satisfactory fitting is obtained. Since the whole process is not a convex program, we can only expect local optimal estimation.

4 Plan

The following is a tentative list of things to do next:

1. Implement the above two algorithms and apply to some toy examples.
2. Provide theoretical analysis (especially for the convex formulation).
3. Study the computation-accuracy trade-off of the convex piecewise linear approximation.
4. Extend the convex regression to convex dictionary learning, where we infer both $\boldsymbol{x}$ and $\{\alpha, \boldsymbol{\beta}\}$.
5. Relate to the point-based value iteration algorithm in POMDP [J. Pineau, G. Gordon and S. Thrun, 2003].
6. Extend to graph estimation, where $\boldsymbol{y}$ is a node on the graph, and $\boldsymbol{x}$ represents the remaining nodes.

5 Appendix: ADMM for the Convex Formulation

The convex program (3) is equivalent to

$$\min_{\{\boldsymbol{h},\boldsymbol{\beta},\boldsymbol{C},\boldsymbol{S}\}} \frac{1}{2}\sum_{i=1}^{n}(y_i - h_i)^2 + \lambda\|\boldsymbol{C}\|_{2,1} \;\; \text{s.t.} \;\; \boldsymbol{C} = \boldsymbol{\beta}, \;\; h_j = h_i + \boldsymbol{\beta}_i^\top(\boldsymbol{x}_j - \boldsymbol{x}_i) + S_{ji}, \;\; S_{ji} \ge 0, \quad (\forall i, j),$$

for which we could construct the following ADMM objective function

$$\begin{aligned}\min_{\{\boldsymbol{h},\boldsymbol{\beta},\boldsymbol{C},\boldsymbol{S},\boldsymbol{W},\boldsymbol{M}\}} \; & \frac{1}{2}\sum_{i=1}^{n}(y_i - h_i)^2 + \lambda\|\boldsymbol{C}\|_{2,1} \\ & + \sum_{i=1}^{n}\sum_{j=1}^{n}\Big(W_{ji}\cdot(h_j - (h_i + \boldsymbol{\beta}_i^\top(\boldsymbol{x}_j - \boldsymbol{x}_i) + S_{ji})) + \frac{\mu}{2}\|h_j - (h_i + \boldsymbol{\beta}_i^\top(\boldsymbol{x}_j - \boldsymbol{x}_i) + S_{ji})\|^2\Big) \\ & + \operatorname{tr}(\boldsymbol{M}^\top(\boldsymbol{C} - \boldsymbol{\beta})) + \frac{\mu}{2}\|\boldsymbol{C} - \boldsymbol{\beta}\|^2 \qquad \text{s.t.} \;\; S_{ji} \ge 0, \quad (\forall i, j).\end{aligned}$$

Here $\boldsymbol{W}$ and $\boldsymbol{M}$ are the Lagrange multipliers and μ is a hyper-parameter in ADMM. Equations for updating the parameters are summarized as follows:

1. Update $\boldsymbol{C}$.

$$\min_{\boldsymbol{C}} \lambda\|\boldsymbol{C}\|_{2,1} + \frac{\mu}{2}\|\boldsymbol{C} - (\boldsymbol{\beta} - \mu^{-1}\boldsymbol{M})\|^2 \quad \Rightarrow \quad \boldsymbol{C}_{t\cdot} = \max\left(1 - \frac{\lambda\mu^{-1}}{\|\boldsymbol{D}_{t\cdot}\|_2}, 0\right)\boldsymbol{D}_{t\cdot}$$

   where $\boldsymbol{D} \triangleq \boldsymbol{\beta} - \mu^{-1}\boldsymbol{M}$ and $\boldsymbol{C}_{t\cdot}$ denotes row t of matrix $\boldsymbol{C}$.

2. Update $\boldsymbol{h}$.

$$\min_{\boldsymbol{h}} \frac{1}{2}\sum_{i=1}^{n}(y_i - h_i)^2 + \frac{\mu}{2}\|h_j - (h_i + \boldsymbol{\beta}_i^\top(\boldsymbol{x}_j - \boldsymbol{x}_i) + S_{ji}) + \mu^{-1}W_{ji}\|^2 \quad \Rightarrow$$

$$\boldsymbol{h} = \Big(\mu^{-1}\boldsymbol{I} + \sum_{i=1}^{n}\sum_{j=1}^{n}(\boldsymbol{e}_j - \boldsymbol{e}_i)(\boldsymbol{e}_j - \boldsymbol{e}_i)^\top\Big)^{-1}\Big(\mu^{-1}\boldsymbol{y} + \sum_{i=1}^{n}\sum_{j=1}^{n}(\boldsymbol{e}_j - \boldsymbol{e}_i)((\boldsymbol{x}_j - \boldsymbol{x}_i)^\top\boldsymbol{\beta}_i + S_{ji} - \mu^{-1}W_{ji})\Big)$$

   where $\boldsymbol{e}_i \in \mathbb{R}^n$ is all zero except a one in element i. The summations on i and j in the above equation can be computed efficiently using vector operator.

3. Update $\boldsymbol{\beta}$.

$$\min_{\boldsymbol{\beta}_i} \sum_{j=1}^{n}\frac{\mu}{2}\|h_j - (h_i + \boldsymbol{\beta}_i^\top(\boldsymbol{x}_j - \boldsymbol{x}_i) + S_{ji}) + \mu^{-1}W_{ji}\|^2 + \frac{\mu}{2}\|\boldsymbol{\beta}_i - (\boldsymbol{C}_i + \mu^{-1}\boldsymbol{M}_i)\|^2 \quad \Rightarrow$$

$$\boldsymbol{\beta}_i = \Big(\boldsymbol{I} + \sum_{j=1}^{n}(\boldsymbol{x}_j - \boldsymbol{x}_i)(\boldsymbol{x}_j - \boldsymbol{x}_i)^\top\Big)^{-1}\Big(\boldsymbol{C}_i + \mu^{-1}\boldsymbol{M}_i + \sum_{j=1}^{n}(\boldsymbol{x}_j - \boldsymbol{x}_i)(h_j - h_i - S_{ji} + \mu^{-1}W_{ji})\Big).$$

4. Update $\boldsymbol{S}$.

$$\min_{S_{ji}} \frac{\mu}{2}\|h_j-(h_i+\boldsymbol{\beta}_i^\top(\boldsymbol{x}_j-\boldsymbol{x}_i)+S_{ji})+\mu^{-1}W_{ji}\|^2 \text{ s.t. } S_{ji}\geq 0 \ \Rightarrow S_{ji}=\max(h_j-(h_i+\boldsymbol{\beta}_i^\top(\boldsymbol{x}_j-\boldsymbol{x}_i))+\mu^{-1}W_{ji},0).$$

5. Update $\boldsymbol{W}$ and $\boldsymbol{M}$.

$$W_{ji}=W_{ji}+\mu\cdot(h_j-(h_i+\boldsymbol{\beta}_i^\top(\boldsymbol{x}_j-\boldsymbol{x}_i)+S_{ji})), \quad \boldsymbol{M}=\boldsymbol{M}+\mu\cdot(\boldsymbol{C}-\boldsymbol{\beta}).$$

References

[1] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[2] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[3] A. Magnani and S. Boyd. Convex piecewise-linear fitting. *Optimization and Engineering*, 10:1–17, 2009.

[4] L. Hannah and D. Dunson. Multivariate convex regression with adaptive partitioning. *arXiv preprint arXiv:1105.1924v2*, 2011.