

Robust Regression (Huber Regression)

March 9, 2021

1 Introduction

Let's make a quick summary of what we have:

A set of data points $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$ where $\mathbf{x}_i \in \mathbb{R}^d$. Huber regression is solving this optimization problem:

$$\text{minimize } \sum_{i=1}^n \phi(\boldsymbol{\theta}^T \mathbf{x}_i - y_i) \quad (1)$$

where variable $\boldsymbol{\theta} \in \mathbb{R}^d$ and ϕ is a Huber penalty function:

$$\phi(x) = \begin{cases} x^2, & \text{if } |x| \leq M, \\ M(2|x| - M) & \text{otherwise} \end{cases}$$

Reader can easily verify that Huber function is a convex function therefore this problem is a convex optimization problem

Note that, in this document, whenever I mention vector, it means that it is a column vector.

Before we left, let's take a look to the differences between Huber Regression and Linear Regression

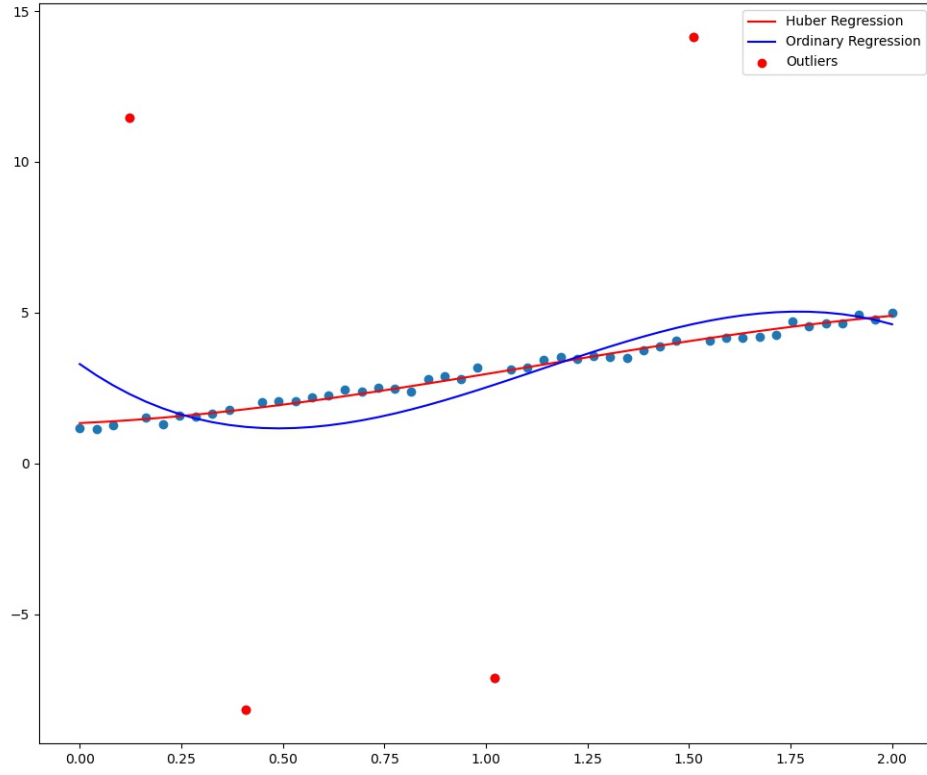


Figure 1: Compare Huber Polynomial Regression and Ordinary Polynomial Regression

As you can see, Huber Regression is **sensitive with outliers** whereas the Ordinary Regression is easily 'tilted' by outliers

2 Rewrite problem

Simply let $u_i = \theta^T \mathbf{x}_i - y_i \forall i$, then the original problem will become:

$$\begin{aligned} & \text{minimize} \sum_{i=1}^n \phi(u_i) \\ & \text{subject to } u_i = \theta^T \mathbf{x}_i - y_i \forall i \quad (2) \end{aligned}$$

Let:

- $f : \mathbb{R}^{2d} \rightarrow \mathbb{R}$ where $f(\mathbf{x}) = \sum_{i=1}^n \phi(x_i)$

- $\mathbf{t} = \begin{pmatrix} u_1 \\ u_2 \\ \dots \\ u_n \\ \boldsymbol{\theta} \end{pmatrix}$

- $A = \begin{pmatrix} -\mathbf{I}_d & \mathbf{1} & \mathbf{X} \end{pmatrix}$ where matrix $\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \dots \\ \mathbf{x}_n^T \end{pmatrix}$

- $\mathbf{b} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix}$

Then, problem (2) will become:

$$\begin{aligned} & \text{minimize } f(\mathbf{t}) \\ & \text{subject to } A\mathbf{t} = \mathbf{b} \quad (3) \end{aligned}$$

(3) is a equality constrain convex optimization problem that we need
What about initial starting point ?

Consider vector $\mathbf{u} = \begin{pmatrix} 1 - y_1 \\ 1 - y_2 \\ \dots \\ 1 - y_n \\ 1 \\ 0 \\ 0 \\ \dots \\ 0 \end{pmatrix}$, reader can verify that $A\mathbf{u} = \mathbf{b}$. Therefore, vector \mathbf{u} is a feasible

starting point of this robust regression

3 Compute its gradient ∇f and its Hessian $\nabla^2 f$

Reader can verify that:

$$\nabla f = \begin{pmatrix} \phi'(x_1) \\ \phi'(x_2) \\ \dots \\ \phi'(x_n) \\ 0 \\ \dots \\ 0 \end{pmatrix}$$

$$\nabla^2 f = \text{diag}(a_1, a_2, \dots, a_{2n}) = \begin{pmatrix} a_1 & 0 & \dots & 0 \\ 0 & a_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{2n} \end{pmatrix} \text{ where } a_i = \begin{cases} 2, & \text{if } |x_i| \leq M, i = 1, 2, \dots, n \\ 0, & \text{otherwise} \end{cases}$$

That's it! That's all you need to know to understand what I am doing in the code. Reader want to know about the algorithm can check [1] in the **References** section

References

- [1] Stephen Boyd, Lieven Vandenberghe, *Convex Optimization*.
- [2] My figure and code on GitHub:
<https://github.com/PhuThanh-Nguyen/Small-Projects/tree/main/Robust%20Regression>