上海财经大学

毕 业 论 文

题目 非线性模型在分位数回归中的应用探索

姓　　名　韩修齐

学　　号　2020110868

学　　院　信息管理与工程学院

专　　业　计算机科学与技术（双）

指导教师　周志明

定稿日期　2024　年　4　月

声　　明

本人郑重声明所呈交的论文是我个人在指导老师的指导下进行的研究工作及取得的研究成果，不存在任何剽窃、抄袭他人学术成果的现象。我同意（　　）/不同意（　　）本论文作为学校的信息资料使用。

论文作者（签名）_____

年　　月　　日

非线性模型在分位数回归中的应用探索

摘　要

本文对分位数的定义进行了深入阐释，探究了线性背景下分位数回归的参数估计原理，同时扩展地类比了非线性背景下分位数回归估计方法。验证了分位数损失 Pinball Loss 被平滑损失 Huber Loss 逼近的可行性，解决了 Pinball Loss 的不可导性质导致了部分非线性模型模型中无法拟合分位数回归的问题。深入探究了经典非线性模型（XGBoost 和神经网络）的原理，与分位数回归结合，在不同类型的数据集上进行了实验，并对拟合结果进行了评估。

本文深入探究了机器学习模型的数学原理，对文献中的数学推导进行了复现；对机器学习模型的源码进行了探究。在商业决策、科学及社会科学研究中，了解分位数回归原理并利用多样的模型进行拟合有利于洞悉数据的生成机制，相关理论和模型值得进一步推广。

在非线性背景下 Huber Loss 对 Pinball Loss 尚待验证，模型实验的调优有待进一步提升。

关键词：机器学习理论，分位数回归，Huber 损失，XGBoost，神经网络

Abstract

This paper delves deeply into the definition of quantiles, exploring the principles of parameter estimation in quantile regression within a linear context, and analogously extends to estimation methods in quantile regression under a nonlinear setting. It verifies the feasibility of approximating Pinball Loss with the smooth Huber Loss, addressing the issue that the non-differentiability of Pinball Loss hinders the fit of quantile regression in certain nonlinear models. The study thoroughly investigates the principles of classic nonlinear models (such as XGBoost and neural networks), integrates them with quantile regression, conducts experiments on various types of datasets, and evaluates the fitting results.

The paper deeply explores the mathematical principles of machine learning models, reproducing the mathematical derivations found in the literature; it also examines the source code of machine learning models. Understanding the principles of quantile regression and fitting using a variety of models in business decision-making, scientific, and social scientific research helps to gain insight into the mechanisms of data generation. The related theories and models are worth further promotion.

The validation of Huber Loss against Pinball Loss in a nonlinear context remains to be verified, and the tuning of model experiments requires further enhancement.

Key words: Machine Leaning theories, quantile regression, Huber Loss, XGBoost, Neural Network

目 录

一、 引言

（一）研究背景及意义

自提出以来，分位数回归已被广泛应用于经济学、医学、环境科学等多个领域。分位数回归以其对异常值的高度鲁棒性、能够揭示变量之间复杂关系的能力，成为统计学和实证研究中一个重要的工具。尽管它在计算上比最小二乘法更为复杂，但随着计算技术的发展和相关软件工具的完善，分位数回归的应用越来越广泛。深入理解分位数回归的原理，才能够将分位数回归与日益丰富的各类机器学习模型相结合，以不断提升模型的效果，为数据分析提供更丰富的方法。

（二）国内外研究现状及文献综述

Koenker 和 Bassett（1978）首次提出分位数回归，旨在超越传统的最小二乘法回归模型，特别是在处理非对称误差分布和异常值时的局限性。与最小二乘法关注于条件均值的估计不同，分位数回归关注于条件分布的不同分位数，为研究变量之间的复杂关系提供了更丰富的视角。分位数回归通过最小化加权残差的绝对值来估计回归系数，允许研究者探究自变量对因变量不同分位数（如中位数、四分位数）的影响。这种方法特别适用于数据分布偏斜或包含异常值的情况，可以获得比传统回归模型更稳健的估计结果。

当线性分位数回归已经得到了深入研究和广泛应用，非线性分位数回归的算法发展却相对滞后。Koenker 和 Park（1992）填补了这一空白，提出了一种计算非线性分位数回归估计的内点算法。他们总结了内点方法在解决线性问题上的应用，然后详细描述了如何将这些方法扩展到非线性问题上。内点算法的核心在于将原始问题转化为一系列线性子问题，通过迭代求解，最终逼近非线性问题的解。特别是，作者提出了一种新的算法，能够有效处理响应函数在参数上的非线性性质。一系列测试结果表明与现有的非线性最小二乘法和其他非线性分位数回归方法相比，该算法显示出良好的性能和稳健性，丰富了分位数回归的理论和方法学，也为处理复杂数据结构提供了强有力的工具。

Chen（2007）引入了一种新的计算回归分位数函数的方法，该方法基于平滑非可微的量化回归目标函数 pT，通过有限的平滑算法实现。该文所提出的有限平滑算法，展示了与经典单纯形算法相比，在计算速度上的显著优势，特别是当分位数回归设计矩阵中包含大量协变量时。与先前介绍的内点算法相比，在处理大量协变量的设计矩阵时，该算法的速度要明显快于内点算法，在精度上也

与单纯形算法持平，而内点算法在理论上仅能提供近似解，实践中可能需要四舍五入来提高解的精度。Chen 的研究不仅在算法设计上取得了进步，还在分位数回归的理论和应用领域内提供了新的见解，进一步扩大了分位数回归的应用范围，尤其是在处理大规模数据集时，能够有效提高计算效率和结果的可靠性。

Chen 等（2016）提出 XGBoost 算法，是当下最受欢迎的机器学习方法之一。它基于决策树的 boosting 算法，李航（2012）进行了详尽的介绍。其简单的实现、快速计算和序列学习，使其预测相较于其他方法更为准确。然而，如 XGBoost 这类机器学习模型的不确定性确定技术在其不同的应用中尚未得到普遍认可。Yin 等（2023）提出了对 XGBoost 的增强，通过将修改后的分位数回归用作目标函数——在分位数回归模型中引入了 Huber 范数，以构建分位数回归误差函数的可微近似——来估计不确定性（QXGBoost）。这一关键步骤允许使用基于梯度的优化算法的 XGBoost 高效地进行概率预测，为 XGBoost 的不确定性量化提供了一种易于理解和实施的方式。

根据邱锡鹏（2022）介绍，神经网络，也称为人工神经网络（Artificial Neural Network, ANN），是一种受人脑结构和功能启发的计算系统。它通过大量的简单处理单元（神经元）之间的相互连接来模拟人脑处理信息的方式，从而能够学习和识别复杂的模式和数据。Cannon（2011）讨论了分位数回归神经网络（Quantile Regression Neural Networks, QRNNs）的发展和应用，特别强调了其在 R 语言中的实现和用于降尺度预测降水量的应用。QRNN 结合了传统的线性分位数回归模型和人工神经网络的优点，提供了一种非参数的、非线性的建模方法，适合于处理环境数据预测问题，尤其是在涉及到混合离散-连续变量（如降水量、风速或污染物浓度）的情况下。QRNN 不仅可以预测中位数或其他特定分位数值，还能通过预测一系列分位数来完整地描述预测分布，从而直接评估预测的不确定性。

Davino 等（2013）总结了分位数回归的理论和应用。在现代应用研究中，Smelyakov 等（2023）在交通数据集上使用随机森林、梯度提升树、XGBoost 算法进行了训练，比较了决策树各变体模型的性能；Tyurin 等（2023）从利用提升树模型，深入讨论了分位数回归的稳健性；März（2019）则对 XGBoost 模型进行了扩展——利用 XGBoost 的变体 XGBoostLSS 进行概率建模，充分适应分位数回归的要求；Taylor（2000）使用神经网络在金融数据集上进行分位数回归，取得了优良的效果；Xu 等（2017）研发了神经网络变体——CQRNN，利用神经网络合成的方法进行分位数回归，更好地探究变量间的非线性关系；Jantre 等（2021）利用贝叶斯方法，提升了神经网络对分位数回归的效果。

二、 分位数回归及近似原理

（一）Pinball Loss

Koenker 和 Bassett（1978）提出，线性背景（因变量 y 与自变量 X 之间的关系为线性，y_i，x_i 表示第 i 个样本的自变量和因变量，权重为 b）下，τ回归分位数定义为以下目标函数(Pinball Loss)的极小值解：

$$\psi(b;\tau,y,X)=\sum_{\{i:y_i>x_ib\}}\tau|y_i-x_ib|+\sum_{\{i:y_i<x_ib\}}(1-\tau)|y_i-x_ib| \qquad \text{式(2.1)}$$

Yin 等（2023）证明，上述问题可以转化为求解$\psi_\tau(y-\hat{y})=\sum_{\{t:y_i>\hat{y_i}\}}\tau|y_i-\hat{y_i}|+\sum_{\{t:y_i<x_ib\}}(1-\tau)|y_i-\hat{y_i}|$的最小值，其中$\hat{y}$，$\hat{y_i}$均表示估计值。若已知$y$具有分布 F，通过求期望一阶导为零，可以得到解$\hat{y}=F^{-1}(\tau)$，即$\hat{y}$为y的τ分位数。Koenker 和 Bassett 给出了b的估计β^*（b的向量形式）以及唯一解的必要条件。为了便于表示，他们首先进行了如下标记。

对X,y进行划分，其中X有 K 个特征，X,y均有 T 个样本：

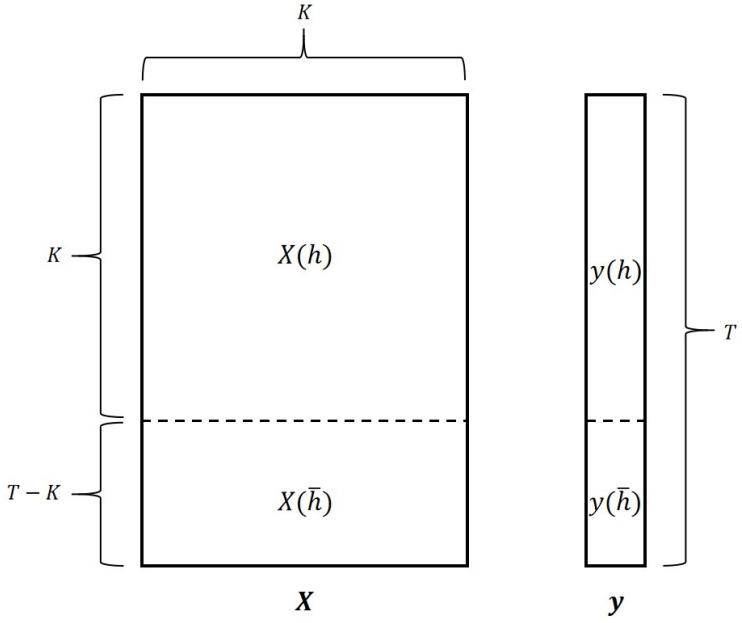

图 2-1 denote 的示意图

其中，h 是全体 H 的任意一个子集，$\bar{h}$是全体 H 除 h 之外的部分，且满足 X(h)的秩为 K。Koenker 和 Bassett 提出，问题(1)的解的集合$B^*(\tau)$至少有一个元素$\beta^*(\tau)$的形式为$X(h)^{-1}y(h)$ 。其唯一性的必要条件由如下过程导出。将式(2.1)表示为

$$\psi(b;\tau,y,X)=\sum_{t=1}^{T}[\tau-\frac{1}{2}+\frac{1}{2}sgn(y_t-x_tb)][y_t-x_tb] \qquad \text{式(2.2)}$$

sgn为示性函数，当 u 大于，等于或小于 0 时，$sgn\,u$ 分别取 1，0 和-1。式(2.2)对 b 求偏导，并与

向量w求点积，得到目标函数ψ在w方向上的变化率：

$$\psi'(b;w)=\sum_{i=1}^{T}[\frac{1}{2}-\tau-\frac{1}{2}sgn^*(y_i-x_ib;-x_iw)]x_iw，\ sgn^*(u;z)=\begin{cases}sgn\,u,\ 当u\neq 0\\ sgn\,z,\ 当u=0\end{cases} \qquad 式(2.3)$$

由于$\psi(b)$是凸性的，当且仅当对于任意w，$\psi'(\beta^*;w)>0$ 时，即ψ在任意方向上的偏导数都大于 0 时，$\psi(b)$在β^*处取得唯一极小值。当$\beta^*=X(h)^{-1}y(h)$时，

$$\psi'(\beta^*;w)=\sum_{t\in h}[\frac{1}{2}-\tau+\frac{1}{2}sgn(x_iw)]x_iw+\sum_{t\in\bar{h}}[\frac{1}{2}-\tau-\frac{1}{2}sgn^*(y_i-x_i\beta^*;-x_iw)]x_iw \qquad 式(2.4)$$

用v表示$X(h)w$，则

$$w=X(h)^{-1}v$$

$$v_k=x_kw，\ 当1\leq k\leq K$$

注意，v_k此时是一个标量。代入式(2.4)，若要$\psi'(\beta^*;w)>0$，则

$$\sum_{k=1}^{K}[(\frac{1}{2}-\tau)v_k+\frac{1}{2}|v_k|]+\sum_{t\in\bar{h}}[\frac{1}{2}-\tau-\frac{1}{2}sgn^*(y_t-x_t\beta^*;x_tX(h)^{-1}v)]x_tX(h)^{-1}v>0$$

$$若v_k>0，(\frac{1}{2}-\tau)v_k+\frac{1}{2}|v_k|=(1-\tau)v_k \qquad 式(2.5)$$

$$若v_k<0，(\frac{1}{2}-\tau)v_k+\frac{1}{2}|v_k|=\tau v_k$$

由于$\sum_{k=1}^{K}[(\frac{1}{2}-\tau)v_k+\frac{1}{2}|v_k|]$的向量形式为$\begin{bmatrix}\frac{1}{2}-\tau\\ \cdots\\ \frac{1}{2}-\tau\end{bmatrix}^T\cdot\begin{bmatrix}v_1\\ \cdots\\ v_K\end{bmatrix}+\begin{bmatrix}\frac{1}{2}\\ \cdots\\ \frac{1}{2}\end{bmatrix}^T\cdot\begin{bmatrix}|v_1|\\ \cdots\\ |v_K|\end{bmatrix}$，在每一维度上消去$v_k$，定义$l_K$表示$K$维向量[1 … 1]，式(2.5)等价于

$$(\tau-1)l_K\leq\sum_{i\in\bar{h}}[\frac{1}{2}-\tau-\frac{1}{2}sgn^*(y_i-x_i\beta^*;x_iX(h)^{-1}v)]x_iX(h)^{-1}\leq\tau l_K \qquad 式(2.6)$$

由于真实残差$u_i=y_i-x_i\beta$的分布F是连续的，因此$y_i-x_i\beta^*$一定不为 0。式(2.6)可以继续简化为：

$$(\tau-1)l_K\leq\sum_{i\in\bar{h}}[\frac{1}{2}-\tau-\frac{1}{2}sgn(y_i-x_i\beta^*)]x_iX(h)^{-1}\leq\tau l_K \qquad 式(2.7)$$

由此，β^*作为问题(1)唯一解的必要条件即β^*满足式(2.7)。

（二）Pinball Loss 在非线性背景下的推广

Koner 和 Park（1994）利用内点（interior point）算法，以对称的 L1 损失的极小值求解为例，将该极小值求解问题从线性背景推广至非线性背景，说明了非线性分位数回归模型也能被估计。

首先考虑线性模型下，L1 正则损失$R(b)$表示为

$$R(b)=\sum_{i=1}^{n}|y_i-x_i^Tb| \qquad \text{式(2.8)}$$

这与(1)式中的目标函数相同。对上述损失进行最小化，该问题的对偶问题可表示为

$$max\{y^Td|d\in\Omega=\{d\in[-1,1]^n,X^Td=0\}\} \qquad \text{式(2.9)}$$

为 d 设置一个初始的可行解，例如 d 为 n 维的 0 向量。令$D=diag(min\{1+d_i，1-d_i\})$，其中$d_i$为对偶问题中第 i 个样本（$y_i$，$x_i$）对应的权重。则在变换坐标系$D^{-1}d$中，残差的投影为$D\hat{u}$。为了在变换的坐标系中沿着负梯度方向$-y^T$移动，同时又保持解的可行性，我们将 y 投影到 X 的零空间上以确保等式约束能够被满足。

$$D\hat{u}=(I-DX(X^TD^2X)^{-1}X^TD)Dy=D(y-Xb) \qquad \text{式(2.10)}$$

则其中$b=(X^TD^2X)^{-1}X^TD^2y$。若$e_i^{'}$为第 i 个基向量，令

$$\alpha=max\{max\{\frac{e_i^{'}D^2\hat{u}}{1+d_i},\frac{-e_i^{'}D^2\hat{u}}{1-d_i}\}\} \qquad \text{式(2.11)}$$

则配合学习率η，d 可以这样更新：

$$d\leftarrow d+(\frac{\eta}{\alpha})D^2\hat{u} \qquad \text{式(2.12)}$$

随着每一轮 k 的更新（由 d 得到 D），原问题中的参数也将不断更新

$$b_k=(X^TD_k^2X)^{-1}X^TD_k^2y \qquad \text{式(2.13)}$$

接下来希望将上述解法推广至非线性背景。设非线性模型为f_0，非线性模型的损失最小化问题可以表示为：

$$min_{t\in R^p}\sum|y_i-f_0(x_i,t)| \qquad \text{式(2.14)}$$

其中 t 是模型中的参数。令$f(t)=(f_i(t))$，$J(t)=\frac{\partial f_i(t)}{\partial t_j}$，则$J$是雅可比矩阵。上式的最小化问题可以转化为迭代地最小化

$$\sum\left|f_i(t)-J_i(t)^{'}\delta\right| \qquad \text{式(2.15)}$$

其对偶问题为

$$max\{f^{'}d\in[-1,1]^n,J^{'}d=0\} \qquad \text{式(2.16)}$$

由于非线性，每轮迭代中无法彻底求解式(2.16)。在类似线性模型迭代地更新f和J的过程中，添加两步：

(i)Dual step

$$\begin{gathered}s=D^2(I-J(J^{'}D^2J)^{-1}J^{'}D^2)f\\ d\leftarrow d+(\frac{\eta}{\alpha})s\\ \alpha=\max_i\{max\{\frac{s_i}{1+d_i},\frac{s_i}{1-d_i}\}\}\end{gathered} \qquad \text{式(2.16.1)}$$

该步骤用于更新 d 和α；然后由下一步 Primal Step 更新 t。

(ii)Primal Step

$$\delta = (J^{'}D^2J)^{-1}J^{'}D^2f^{'}$$

$$t \leftarrow t + \lambda^*\delta，\ \lambda^* = \underset{\lambda}{argmin}\,||f(t+\lambda\delta)||_1 \qquad \text{式(2.16.2)}$$

同时调整 d，将当前的 d 投影到新的 J 的零空间上，即

$$\widehat{d} = (I - J(J^{'}J)^{-1}J^{'})d$$

$$d \leftarrow \widehat{d} \times \frac{1}{\underset{i}{max}\{|\widehat{d}_i|\} + \varepsilon} \qquad \text{式(2.16.3)}$$

直至收敛。

对于非对称的分位数损失，调整常参数即可得到上述结论。该方法给出了解的形式，并未直接应用于后续的机器学习算法中。

（三）Huber Loss 近似 Pinball Loss

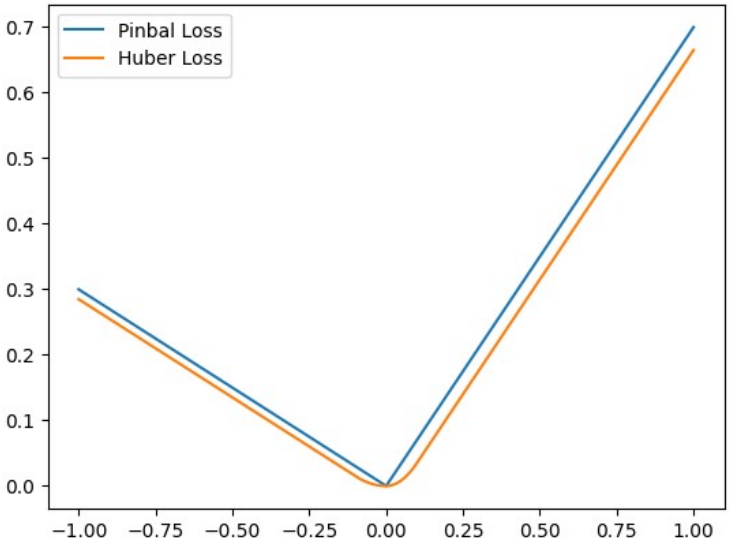

图 2-2 Huber Loss 与 Pinball Loss

Chen(2007)提出了一种平滑算可以用于近似式(2.1)中的损失函数，称为 Huber 损失。第 i 个样本为（y_i，x_i），β为线性模型的权重，则令$r_i(\beta) = y_i - x_i^T\beta$，Huber 函数定义如下：

$$D_\gamma(\beta) = \sum_{i=1}^{n} H_\gamma(r_i(\beta))，\ 其中 H_\gamma(t) = \begin{cases} \frac{t^2}{2\gamma}，\ 当|t| \le \gamma \\ |t| - \frac{\gamma}{2}，\ 当|t| > \gamma \end{cases} \qquad \text{式(2.17)}$$

γ是一个接近 0 的超参数。$r_i(\beta)$代入式（2.1），ψ可以表示为$\sum_{i=1}^{n}\rho_\tau(r_i)$，其中是关于 0 的分段函数。分位数损失$\psi = D_{\rho_\tau}(\beta) = \sum_{i=1}^{n}\rho_\tau(r_i)$的近似 Huber 损失为：

$$D_{\gamma,\tau}(\beta)=\sum_{i=1}^{n}H_{\gamma,\tau}(r_i)\text{，}\ H_{\gamma,\tau}(t)=\begin{cases}t(\tau-1)-\dfrac{1}{2}(\tau-1)^2\gamma\text{，}\ \text{当}t\le(\tau-1)\gamma\\ \dfrac{t^2}{2\gamma}\text{，}\ \text{当}(\tau-1)\gamma\le t\le\tau\gamma\\ t\tau-\dfrac{1}{2}\tau^2\gamma\text{，}\ \text{当}t>\tau\gamma\end{cases}\qquad\text{式(2.18)}$$

Pinball 和 Huber 函数在$\tau=0.7$时的示意如图 2-2。Chen(2007)证明，在线性背景下，通过最小化该近似平滑损失，估计出的参数能够近似极小化分位数损失的估计参数。证明如下。

1. 定义和 denote

定义向量$s_{\gamma,\tau}(\beta)=\begin{bmatrix}s_1(\beta)\\ \cdots\\ s_n(\beta)\end{bmatrix}$，其中

$$s_i(\beta)=\begin{cases}-1\text{，}\ \text{当}r_i\le(\tau-1)\gamma\\ 0\text{，}\ \text{当}(\tau-1)\gamma\le r_i\le\tau\gamma\text{，}i\text{为样本的编号}\\ 1\text{，}\ \text{当}r_i>\tau\gamma\end{cases}\qquad\text{式(2.19)}$$

令$w_i(\beta)=1-s_i^2(\beta)$。用$s_i$表示$H_{\gamma,\tau}$，有

$$H_{\gamma,\tau}(r_i)=\frac{1}{2\gamma}w_ir_i^2+s_i[\frac{1}{2}r_i+\frac{1}{4}(1-2\tau)\gamma+s_i(r_i(\tau-\frac{1}{2})-\frac{1}{4}(1-2\tau+2\tau^2)\gamma)]\qquad\text{式(2.20)}$$

为了将式(2.11)表示为矩阵形式，定义

$$W_{\gamma,\tau}=\begin{bmatrix}w_1&\cdots&0\\ \vdots&\ddots&\vdots\\ 0&\cdots&w_n\end{bmatrix}$$

$$g(s)=\begin{bmatrix}g_1(s_1)\\ \cdots\\ g_n(s_n)\end{bmatrix}=\begin{bmatrix}\dfrac{s_1}{2}((2\tau-1)s_1+1)\\ \cdots\\ \dfrac{s_n}{2}((2\tau-1)s_n+1)\end{bmatrix}\qquad\text{式(2.21)}$$

$$c(s)=\sum_{i=1}^{n}\left[\frac{1}{4}(1-2\tau)\gamma s_i-\frac{1}{4}{s_i}^2(1-2\tau+2\tau^2)\gamma\right]$$

$$r=\begin{bmatrix}r_1\\ \cdots\\ r_n\end{bmatrix}$$

式（2.21）中的 s，w 和 g(s)与 r 之间的关系如下：

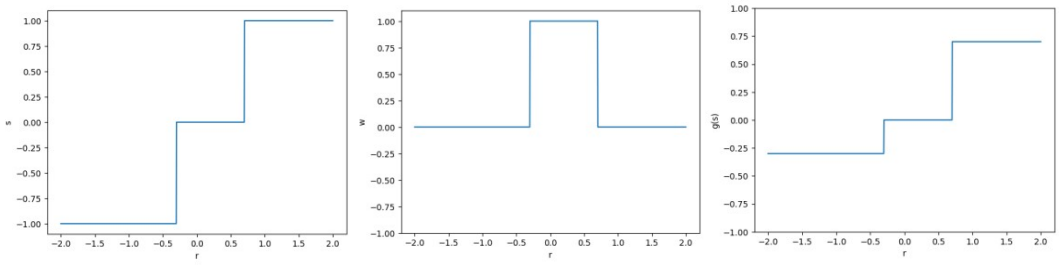

图 2-3 denote 的可视化

2. 近似原理

接下来将说明，为什么 Huber 损失的最优参数解$\beta_{\gamma,\tau}$能近似分位数损失的最优参数解$\beta_{0,\tau}$。

（1）Huber 求最优解

上式的矩阵表达为

$$D_{\gamma,\tau}(\beta)=\frac{1}{2\gamma}r^T W_{\gamma,\tau}r+g^T(s)r+c(s) \qquad \text{式(2.22)}$$

其梯度（一阶导）为

$$D^{(1)}_{\gamma,\tau}(\beta)=-X^T\left[\frac{1}{\gamma}W_{\gamma,\tau}r+g(s)\right] \qquad \text{式(2.23)}$$

其 Hessian（二阶导）为

$$D^{(2)}_{\gamma,\tau}(\beta)=\frac{1}{\gamma}X^T W_{\gamma,\tau}X \qquad \text{式(2.24)}$$

定义一个集合$A_{\gamma,\tau}=\{j|1\le j\le n\wedge s_j(\beta)=0\}$，这是一个索引$j$的集合，被这些 j 标志的所有的示性函数s_j都取 0。因此我们可以利用该集合表示 Huber 梯度，并推导出$\beta_{\gamma,\tau}$成为$D_{\gamma,\tau}(\beta)$的极小值点的必要条件，即梯度为 0：

$$D^{(1)}_{\gamma,\tau}(\beta)=-\frac{1}{\gamma}\sum_{j\in A_{\gamma,\tau}(\beta)}r_j x_j-\sum_{j\notin A_{\gamma,\tau}(\beta)}g_j(s_j)x_j=0 \qquad \text{式(2.25)}$$

注意，$D_{\gamma,\tau}(\beta)$的极小值解可能是不唯一的。假设其全部解的集合为$M_{\gamma,\tau}$，则$\beta_{\gamma,\tau}\in M_{\gamma,\tau}$。

（2）Huber 最优解和 Pinball 最优解的关系

由式(2.23)和式(2.25)中的必要条件，

$$-X^T\left[\frac{1}{\gamma}W_{\gamma,\tau}r+g(s)\right]=0 \qquad \text{式(2.26)}$$

在等式两边分别乘上$-\gamma$ ，并将$r_i(\beta)=y_i-x_i^T\beta$代入

$$\gamma X^T g_{\gamma,\tau}(s_{\gamma,\tau})+X^T W_{\gamma,\tau}(y-X\beta_{\gamma,\tau})=0 \qquad \text{式(2.27)}$$

由上式，又可推得

$$(\gamma-\epsilon)X^T g_{\gamma-\epsilon,\tau}(s_{\gamma-\epsilon,\tau})+X^T W_{\gamma-\epsilon,\tau}(y-X\beta_{\gamma-\epsilon,\tau})=0 \quad \text{式(2.28)}$$

为了继续化简式(2.28)，希望对s及相关函数g、W进行处理，须引入两个引理。引理及证明如下：

引理 1：固定一 $\tau\in(0,1)$，在$\beta\in M_{\delta,\tau}$的条件下，$s_{\gamma,\tau}(M)$是恒定不变的。

假设$\beta\in M_{\gamma,\tau}$，即β为$D_{\gamma,\tau}$的一个极小值解。设$s=s_{\gamma,\tau}(\beta)$。接下来假设两种情况：

①当$\alpha\in C_S$，$C_S=\{\alpha|s_{\gamma,\tau}(\alpha)=s\}$，则仅$D_{\gamma,\tau}^{(1)}$中存在含有$r(\alpha)$的项，$D_{\gamma,\tau}^{(2)}$为不含有未知参数的常数项，更高阶的导数全部为 0。因此由泰勒展开，可得

$$D_{\gamma,\tau}(\alpha)=Q_s(\alpha)=\frac{1}{2}(\alpha-\beta)^T D_{\gamma,\tau}^{(2)}(\beta)(\alpha-\beta)+D_{\gamma,\tau}^{(1)}(\beta)(\alpha-\beta)+D_{\gamma,\tau}(\beta) \quad \text{(附 1)}$$

②当$\alpha\in M_{\gamma,\tau}$，则$\alpha$使得$D_{\gamma,\tau}$取得极小值，$D_{\gamma,\tau}(\alpha)=D_{\gamma,\tau}(\beta)$且$D_{\gamma,\tau}^{(1)}(\beta)=0$。

综上，当$\alpha\in C_S\cap M_{\gamma,\tau}$，(附 1)式中二阶展开项为 0，代入式(15)，即：

$$\frac{1}{\gamma}(\alpha-\beta)^T X^T W_{\gamma,\tau}X(\alpha-\beta)=0 \quad \text{(附 2)}$$

考虑$j\in A_{\gamma,\tau}$时，代入$w_j(\beta)=1-s_j^2(\beta)=1$，有${x_j}^T(\alpha-\beta)=0$。因此

$$r_j(\alpha)=y-{x_j}^T\alpha=y-{x_j}^T\beta=r_j(\beta) \quad \text{(附 3)}$$

对于任意α，只要$\alpha\in C_S\cap M_{\gamma,\tau}$，$j\in A_{\gamma,\tau}$，$r_j(\alpha)$就是常数。再取一与$C_S$接触较小的相邻集合$U$，满足$U\cap M_{\gamma,\tau}\neq\emptyset$，则存在$\alpha\in U\cap M_{\gamma,\tau}$。同理，由于$r_j$的连续性，对于这些$\alpha$，$r_j(\alpha)$为常数。而$M_{\gamma,\tau}$具有连通性，不断满足上述条件的$U$，则在整个$M_{\gamma,\tau}$上$r_j$为常数。进而可以推广到$j\notin A_{\gamma,\tau}$的情况，引理 1 由此得证。

引理 2：令 $0<\delta\le\eta<\gamma$，固定一 $\tau\in(0,1)$。如果$s_{\delta,\tau}(M_{\delta,\tau})=s_{\gamma,\tau}(M_{\gamma,\tau})$，则$s_{\eta,\tau}(M_{\eta,\tau})=s_{\gamma,\tau}(M_{\gamma,\tau})$。

证明：首先需要得到两组关系。

①假设$D_{\delta,\tau}(\beta)$和$D_{\gamma,\tau}(\beta)$分别有极小解及其集合$\beta_{\delta,\tau}\in M_{\delta,\tau}$，$\beta_{\gamma,\tau}\in M_{\gamma,\tau}$。设其线性组合为$\beta_{\eta,\tau}$，

$$\beta_{\eta,\tau}=\epsilon\beta_{\delta,\tau}+(1-\epsilon)\beta_{\gamma,\tau},\ \epsilon=\frac{\gamma-\eta}{\gamma-\delta} \quad \text{(附 4)}$$

根据线性性$r_i(\beta)=y_i-x_i^T\beta$，$r(\beta_{\eta,\tau})=\epsilon r(\beta_{\delta,\tau})+(1-\epsilon)r(\beta_{\gamma,\tau})$。代入$\epsilon$可得：

$$(\gamma-\delta)r(\beta_{\eta,\tau})=(\eta-\delta)r(\beta_{\gamma,\tau})+(\gamma-\eta)r(\beta_{\delta,\tau}) \quad \text{(附 5)}$$

排除非零解，有

$$r(\beta_{\eta,\tau}):r(\beta_{\gamma,\tau}):r(\beta_{\delta,\tau})=\eta:\gamma:\delta \quad \text{(附 6)}$$

②已知$s_{\delta,\tau}(M_{\delta,\tau})=s_{\gamma,\tau}(M_{\gamma,\tau})$，$\epsilon\in(0,1)$不改变残差的线性组合与示性函数的映射，$s_{\eta,\tau}(\beta_{\eta,\tau})=s_{\gamma,\tau}(\beta_{\gamma,\tau})$。引理 1 已经给出，一个$\beta_{\gamma,\tau}$能够决定其所在的集合$M$的映射$s$为某一常值，因此$s_{\eta,\tau}(\beta_{\eta,\tau})=s_{\gamma,\tau}(M_{\gamma,\tau})$，对应的$g$、$W$也相等。

将上述两组等式代入必要条件式(25)，有

$$\frac{1}{\eta}\sum_{j\in A_{\eta,\tau}(\beta)}r_j x_j+\sum_{j\notin A_{\eta,\tau}(\beta)}g_j(s_j)x_j=0 \quad \text{(附 7)}$$

因此$\beta_{\eta,\tau}$是$D_{\eta,\tau}(\beta)$的极小值解，$\beta_{\eta,\tau}\in M_{\eta,\tau}$，$s_{\eta,\tau}(M_{\eta,\tau})=s_{\gamma,\tau}(M_{\gamma,\tau})$。

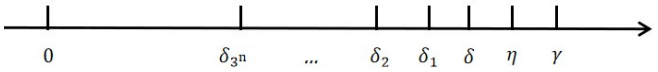

图 2-4 一维示意图

为了表达更简便，将$s_{\gamma,\tau}(M_{\gamma,\tau})$用$s_{\gamma,\tau}(M)$表示；为了便于理解，在 1 维情况下讨论。假设对于所有$\eta\in(0,\gamma)$，$s_{\eta,\tau}(M)\neq s_{\gamma,\tau}(M)$，则对任意$\delta\in(0,\eta]$，$s_{\delta,\tau}(M)\neq s_{\gamma,\tau}(M)$，否则由引理 2，$s_{\eta,\tau}(M)=$

$s_{\gamma,\tau}(M)$，与假设相悖。接下来考虑极限情况。对于 n 个样本的数据集，向量s的全部可能取值为$P=3^n$个（每个样本i可能使s_i取-1，0，1 三个值）。不断取新的δ，假设每个新δ都产生一个不同的s，则最多到第3^n+1个δ且这个δ无限接近 0 时，产生的s会与之前某一个δ_p相同，则在$(0,\delta_p)$范围内任意δ产生的s都相等。将δ_p记为γ_0，只要γ在$(0,\gamma_0)$内变化，$s_{\gamma,\tau}(M)$为常数，进而直接由s决定的W和g也不变。

因此我们有

$$\begin{aligned} s_{\gamma,\tau} &= s_{\gamma-\epsilon,\tau} \\ W_{\gamma,\tau} &= W_{\gamma-\epsilon,\tau} \\ g_{\gamma,\tau}(s_{\gamma,\tau}) &= g_{\gamma-\epsilon,\tau}(s_{\gamma-\epsilon,\tau}) \end{aligned} \qquad \text{式(2.29)}$$

（27）和（28）两式相减，

$$\epsilon X^T g(s) + X^T W X(\beta_{\gamma-\epsilon,\tau} - \beta_{\gamma,\tau}) = 0 = X^T g(s) + X^T W X \frac{\beta_{\gamma-\epsilon,\tau} - \beta_{\gamma,\tau}}{\epsilon} \qquad \text{式(2.30)}$$

由于$(X^T W_{\gamma,\tau} X)v = -\frac{1}{\gamma} X^T W_{\gamma,\tau} r_{\gamma,\tau}$是一个过定的正常线性系统，一定有解，因此$(X^T W_{\gamma,\tau} X)v = X^T g_{\gamma,\tau}$一定有解$v_{\gamma,\tau}$。令$v = \frac{\beta_{\gamma-\epsilon,\tau} - \beta_{\gamma,\tau}}{\epsilon}$，

$$\beta_{\gamma-\epsilon,\tau} = \beta_{\gamma,\tau} + \epsilon v_{\gamma,\tau} \qquad \text{式(2.31)}$$

在式(2.7)中提到，Pinball 损失有唯一最小解β^*的充分必要条件为：

$$(\tau-1)l_K \le \sum_{t\in\bar{h}} [\frac{1}{2} - \tau - \frac{1}{2} sgn^*(y_t - x_t\beta^*)] x_t X(h)^{-1} \le \tau l_K$$

如果类似地定义：

$$s_{0,\tau}(\beta) = \begin{bmatrix} s_1(\beta) \\ \cdots \\ s_n(\beta) \end{bmatrix}, s_i(\beta) = \begin{cases} -1，\ 当 r_i < 0 \\ 0，\ 当 r_i = 0 \\ 1，\ 当 r_i > 0 \end{cases}$$

以及$A_{0,\tau} = \{j | 1 \le j \le n \wedge s_j(\beta) = 0\}$ 即 $\{j | 1 \le j \le n \wedge r_j = 0\}$，则对于式(2.2)的导数

$$\sum_{t=1}^{T} [-\tau + \frac{1}{2} - \frac{1}{2} sgn(y_t - x_t b)] x_t \qquad \text{式(2.32)}$$

当$y_t - x_t b \ne 0$，$\sum_{y_t - x_t b \ne 0} [-\tau - \frac{1}{2} + \frac{1}{2}] x_t = -\sum_{j \notin A_{0,\tau}(\beta)} g_j(s_j) x_j$；当$y_t - x_t b = 0$，$\sum_{y_t - x_t b = 0} [-\tau + \frac{1}{2}] x_t = -\sum_{j \in A_{0,\tau}(\beta)} \xi_j x_j$。因此式(2.32)可以表示为

$$D_{\rho_\tau}^{(1)}(\beta) = -\sum_{j \in A_{0,\tau}(\beta)} \xi_j x_j - \sum_{j \notin A_{0,\tau}(\beta)} g_j(s_j) x_j = 0$$

该形式与式(2.25)的形式是相同的；同时要求$(\tau-1) < \xi_j < \tau$，这与式(2.7)是一致的。由连续性，当$\gamma - \epsilon = 0$ 时，式(2.28)也成立。因此有

$$\beta_{0,\tau} = \beta_{\gamma,\tau} + \gamma v_{\gamma,\tau} \qquad \text{式(2.33)}$$

其中$\beta_{0,\tau}$是$D_{\gamma,\tau}(\beta)$的最优解，也就是$D_{\rho_\tau}(\beta)$的最优解。

三、 两个非线性模型原理

（一）XGBoost 原理

CART 决策树表达为：

$$f(x_i)=\sum_{m=1}^{M} c_m I(x_i \in R_m) \qquad \text{式（3.1）}$$

即对于第 i 个样本向量x_i，经过决策树 f，落入哪个节点 R，便输出为该节点对应的值 c（m 是对应 R,c 的编号，I表示示性函数）。此后的的主要目标为确定树 f 的具体形式，例如树的深度、每层节点如何分裂。

之后，要决定树的深度（可以通过自定义、限制子节点包含样本数或给定精度）。梯度提升树是以决策树（回归树）作为基学习器，进行集成学习。梯度提升树采用前向分布算法，首先确定初始树，$f_0(x)=0$。此后的每一轮迭代中，增加一棵新的基学习器（树）T，其参数为Θ_m。目标是每次增加的这棵新树都能够使得总体损失越来越小，即第 m 步的损失要比第 m-1 步小。如果第 m-1 步的集成模型为$f_{m-1}(x)$，则作为加法模型，

$$f_m(x)=f_{m-1}(x)+T(x;\Theta_m) \qquad \text{式（3.2）}$$

为了实现真实值y和模型估计值$f(x)$导出的总体损失L变小，应该有：

$$L(y^{(i)},f_m(x^{(i)}))<L(y^{(i)},f_{m-1}(x^{(i)})) \qquad \text{式（3.3）}$$

即

$$L(y^{(i)},f_{m-1}(x^{(i)}))-L(y^{(i)},f_m(x^{(i)}))>0 \qquad \text{式（3.3）}$$

将$L(y^{(i)},f_m(x^{(i)}))$视为关于$f_m(x^{(i)})$的函数。将$L(y^{(i)},f_m(x^{(i)}))$在$f_{m-1}(x^{(i)})$处进行一阶泰勒展开，有

$$L(y^{(i)},f_m(x^{(i)}))\approx L(y^{(i)},f_{m-1}(x^{(i)}))+\frac{\partial L(y^{(i)},f_{m-1}(x^{(i)}))}{\partial f_{m-1}(x^{(i)})}\cdot(f_m(x^{(i)})-f_{m-1}(x^{(i)})) \qquad \text{式（3.4）}$$

要满足上述不等式，即应使

$$\begin{aligned}&-\frac{\partial L(y^{(i)},f_{m-1}(x^{(i)}))}{\partial f_{m-1}(x^{(i)})}\cdot(f_m(x^{(i)})-f_{m-1}(x^{(i)}))\\ &=-\frac{\partial L(y^{(i)},f_{m-1}(x^{(i)}))}{\partial f_{m-1}(x^{(i)})}\cdot T(x^{(i)};\Theta_m)\\ &>0\end{aligned} \qquad \text{式（3.5）}$$

如果令$T(x^{(i)};\Theta_m)=-\frac{\partial L(y^{(i)},f_{m-1}(x^{(i)}))}{\partial f_{m-1}(x^{(i)})}$，则上式成为平方和形式，为非负，满足条件。这意味着，$-\frac{\partial L(y^{(i)},f_{m-1}(x^{(i)}))}{\partial f_{m-1}(x^{(i)})}$可以作为第 m 轮的基学习器$T$所要拟合的真实值，也可以看作是上一轮回归拟合的残差值。

1. XGBoost 模型表达

XGBoost 模型可以表示为：

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \qquad \text{式（3.6）}$$

即每个样本的目标估计$\hat{y}_i$由x_i经过K棵树$f_k(k = 1,2,\ldots,K)$得到的。此时需要确定 XGBoost 的目标函数。根据前向分布算法，第 t 轮的模型得到的因变量的估计值应该由前 t-1 轮的模型得到的因变量估计值和第 t 轮的新模型估计值相加构成。即，

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \qquad \text{式（3.7）}$$

2. 目标函数的构建

XGBoost 每一轮前向分布计算的目标函数由两部分构成。

第一为损失，即估计值和真实值之间的差，例如，对于样本 i，用$l(y_i, \hat{y}_i)$表示。

第二为正则项，主要用来防止过拟合。正则项分别该轮前向分布最新一棵回归树节点的的权值ω和节点总数 T 进行惩罚，具体为$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda\sum_{j=1}^{T}\omega_j^2$，$\lambda$为惩罚超参数。如果一共有 K 棵树，则总的损失函数Obj为：

$$Obj = \sum_i l(y_i, \hat{y}_i) + \sum_{k=1}^{K}\Omega(f_k) \qquad \text{式（3.8）}$$

3. 目标函数的优化

当进行到第 t 步前向分布算法时，第 t 轮的目标函数为：

$$Obj^{(t)} = \sum_i^n l(y_i, \hat{y}_i) + \Omega(f_t) \qquad \text{式（3.9）}$$

代入式（3.7），有

$$Obj^{(t)} = \sum_i^n l\left(y_i^{(t)}, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \qquad \text{式（3.10）}$$

对$l\left(y_i^{(t)}, \hat{y}_i^{(t-1)} + f_t(x_i)\right)$视为关于$f_t(x_i)$的函数，对其在$\hat{y}_i^{(t-1)}$处进行二阶泰勒展开，有

$$l\left(y_i^{(t)}, \hat{y}_i^{(t)}\right) \approx l\left(y_i, \hat{y}_i^{(t-1)}\right) + \frac{\partial l\left(y_i, \hat{y}_i^{(t-1)}\right)}{\partial \hat{y}_i^{(t-1)}} \cdot f_t(x_i) + \frac{1}{2}\frac{\partial^2 l\left(y_i, \hat{y}_i^{(t-1)}\right)}{\partial (\hat{y}_i^{(t-1)})^2} \cdot f_t^2(x_i) \qquad \text{式（3.11）}$$

式（3.11）第一项$l\left(y_i,\hat{y}_i^{(t-1)}\right)$为常数，此后不再讨论。令$g_i=\frac{\partial l\left(y_i,\hat{y}_i^{(t-1)}\right)}{\partial \hat{y}_i^{(t-1)}}$，$h_i=\frac{\partial^2 l\left(y_i,\hat{y}_i^{(t-1)}\right)}{\partial (\hat{y}_i^{(t-1)})^2}$且$\sum_i g_i=G_i$，$\sum_i h_i=H_i$。代入，有

$$\tilde{l}\left(y_i^{(t)},\hat{y}_i^{(t)}\right)=g_i\cdot f_t(x_i)+\frac{1}{2}h_i\cdot f_t^2(x_i) \qquad \text{式（3.12）}$$

假设对第 i 个样本x_i，经过 f 树结构得到$w_j=f_t(x_i)$即第 j 个子节点的值，

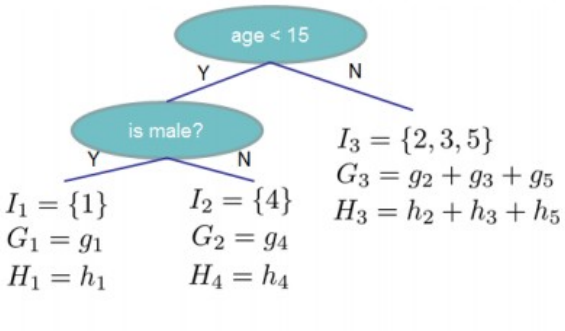

图 3-1 XGBoost 子树的结构

则第 j 个子节点内的总损失为

$$\left(\sum_{样本i落入j节点} g_i\right)\cdot w_j+\frac{1}{2}\left(\sum_{样本i落入j节点} h_i\right)\cdot w_j^2 \qquad \text{式（3.13）}$$

令$\sum_{样本i落入j节点} g_i=G_j$，$\sum_{样本i落入j节点} h_i=H_j$，本轮最新的树一共有 T 个节点，得到

$$Obj^{(t)}=\sum_{j=1}^{T}G_j\cdot w_j+\sum_{j=1}^{T}\frac{1}{2}H_j\cdot w_j^2+\Omega(f_t) \qquad \text{式（3.14）}$$

代入正则项，得到

$$Obj^{(t)}=\sum_{j=1}^{T}G_j\cdot w_j+\frac{1}{2}\sum_{j=1}^{T}(H_j+\lambda)\cdot w_j^2+\gamma T \qquad \text{式（3.15）}$$

为了最小化损失函数，需要得到最优的 w，要满足$Obj^{(t)}$对w_j求一阶导为零，得到最优解

$$w_j^*=-\frac{G_j}{H_j+\lambda} \qquad \text{式（3.16）}$$

代入上式，此时目标函数的极小值点为：

$$Obj^{(t)}=-\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j+\lambda}+\gamma T \qquad \text{式（3.17）}$$

对于每一棵回归树，都需要通过节点的不断分裂来确定树的结构。假设最初只有一个节点，则其目标损失为：

$$Obj_{\,分裂前}=-\frac{1}{2}\cdot\frac{G^2}{H+\lambda}+\gamma \qquad \text{式（3.18）}$$

节点分裂为左右两个子节点 L 和 R，我们可以得到$G=G_L+G_R$，$H=H_L+H_R$

$$Obj_{分裂后} = (-\frac{1}{2}\cdot\frac{{G_L}^2}{H_L+\lambda}+\gamma)+(-\frac{1}{2}\cdot\frac{{G_R}^2}{H_R+\lambda}+\gamma) \quad 式（3.19）$$

分裂前后的增益为：

$$Obj_{分裂前} - Obj_{分裂后} = \frac{1}{2}[\frac{{G_L}^2}{H_L+\lambda}+\frac{{G_R}^2}{H_R+\lambda}-\frac{(G_L+G_R)^2}{(H_L+H_R)+\lambda}]-\gamma \quad 式（3.20）$$

每棵树结构的确定使用精确贪心算法，即每次只关注每一个节点的分裂，而不考虑全局，其大致流程为：

1. 遍历自变量 X 的不同特征，每个特征尝试不同的划分阈值
2. 对于不同的划分方式，分别计算分裂前后的增益$Obj_{分裂前} - Obj_{分裂后}$
3. 取增益最大的划分方式，得到分裂后的新节点
4. 不断分裂节点，直到树的深度达到设定好的超参数

通过增加新的子树以及确定每棵子树的结构，以不断降低总的损失，可以训练出最终的模型。

可以看到，XGBoost 并没有限制损失函数的形式。对于任何能够得到 g，h 的函数，都可以作为损失函数，并决定其最终的目标函数。因此，取 Huber 损失$D_{\gamma,\tau}$，即可利用 XGBoost 进行分位数回归。

（二）神经网络原理

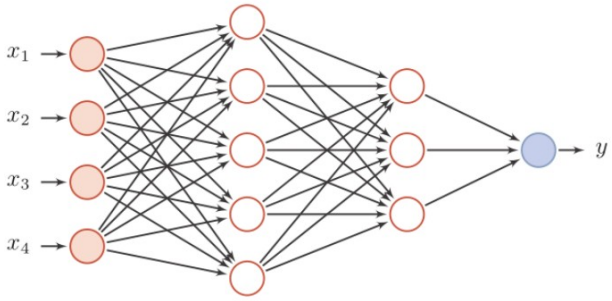

图 3-2 单个神经元示意图

1. 神经网络的结构及前向传播

神经网络中的基本单位是神经元模型。每个神经元接收一组输入，通过 W 加权求和后，再加上一个偏置项 b，表示为

$$Wx+b \quad 式（3.21）$$

此后，通过一个激活函数 f 产生输出 y：

$$y=f(Wx+b) \quad 式（3.22）$$

即单个神经元的行为。激活函数的作用是引入非线性因素，使得神经网络能够学习和表示复杂的模式。没有激活函数，无论神经网络有多少层，最终都等同于一个线性模型。常用的激活函数包括 Sigmoid、Tanh、ReLU 等。

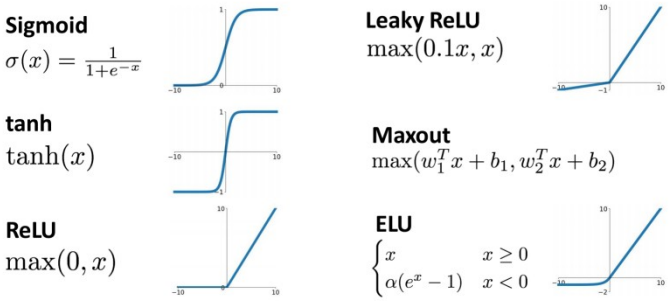

图 3-3 常见的激活函数

神经网络是由多个神经元按照特定的结构组织起来的。根据网络结构的不同，神经网络可以分为多种类型，如前馈（FNN）、循环（RNN）、卷积神经网络（CNN）等。最简单的一种是前馈神经网络，它将神经元分布在多个层次上，数据从输入层经过隐藏层处理后流向输出层，当前神经元的输出作为下一个神经元的输入，形成多层网络结构。

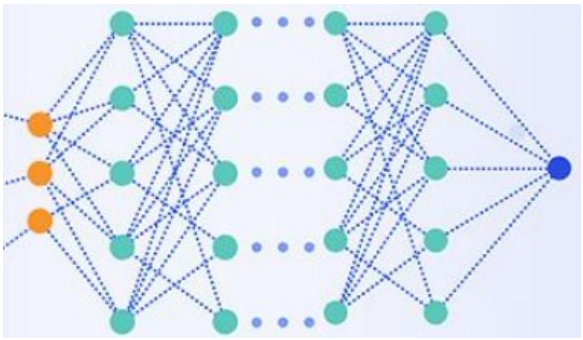

图 多层神经网络

2. 反向传播与优化

反向传播是神经网络训练过程中的关键算法。类似机器学习方法，每一轮迭代中神经网络的输出用于计算损失函数（常见的损失函数有均方误差、交叉熵损失等，不同的任务选择不同的损失函数），即通过网络输出与实际值之间的差异评估神经网络的性能。反向传播计算损失函数关于网络参数的梯度，并利用这些梯度进行梯度下降或其他优化算法更新网络的权重和偏置，从而最小化损失函数。

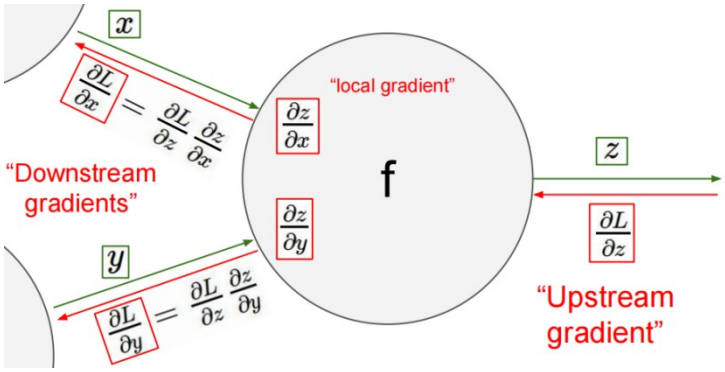

图 3-4 反向传播的示意图

本文所采用的神经网络模型为两层神经元的前馈神经网络。具体到元素，隐藏层维度为 K，X 有 n 个样本（i=1,2,...n）和 m 个特征（j=1,2,..m），g 为第一层神经元的输出，

$$g_{ji} = Relu(\sum_{j=1}^{m} x_{ji} w_{jk}^{(h)} + b_k^{(h)})$$

$$\hat{y}_i = \sum_{k=1}^{K} g_{ji} w_k^{(h)} + b^{(o)} \qquad \text{式（3.23）}$$

由式(2.18)，损失函数为

$$\sum_{i=1}^{n} \rho_\tau(h(y_i - \hat{y}_i)) \qquad \text{式（3.24）}$$

反向传播进行优化，即可得到最优解即得分位数回归解。

本文中采用 Adam 优化器。Adam 是一种常用的优化算法，结合了动量法和自适应学习率调整的思想。它可以有效地调整学习率，并在训练过程中自适应地更新参数。

其中，动量的作用是在更新参数时保持方向，并加速参数更新的速度即梯度下降的过程，有助于跳出局部最优解。

另外，根据参数的梯度的一阶矩估计和二阶矩估计来自适应地调整学习率。它通过计算梯度的指数加权移动平均值和平方梯度的指数加权移动平均值来估计梯度的一阶矩和二阶矩。

以下是 Adam 算法的公式。动量更新为：

$$v_t = \beta_1 v_{t-1} + (1 - \beta_1) \nabla J(\theta_t) \qquad \text{式（3.25）}$$

其中，v_t是时间步t的动量，β_1是动量的衰减系数，$\nabla J(\theta_t)$是参数θ_t对应的梯度。其次是自适应学习率：

$$s_t = \beta_2 s_{t-1} + (1 - \beta_2)(\nabla J(\theta_t))^2 \qquad \text{式（3.26）}$$

其中s_t是时间步t的梯度的平方的指数加权移动平均值，β_2是梯度平方的衰减系数。最后，参数的更新为：

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{s_t} + \epsilon} v_t \qquad \text{式（3.27）}$$

θ_t为待优化的参数，η是学习率，为了防止除以 0，分母上设置一个较小的常数ϵ。

四、 实验

（一）评估

为了更好地评估区间预测的的结果，本文采用以下几个指标。

Prediction Interval Coverage Probability (PICP)：预测区间覆盖概率，是用来评估预测区间的准确性的指标。它表示预测区间中包含实际观测值的概率。对于一个给定的预测区间$[\underline{y_i}, \overline{y_i}]$，如果真实值落在其中的概率达到了预先设定的置信水平，则 PICP 的值越接近于置信水平，说明该预测区间的性能越好。

$$PICP = \frac{1}{n}\sum_{i=1}^{n} c_i \text{，} \quad 其中 c_i = \begin{cases} 1, \ y_i \in [\underline{y_i}, \overline{y_i}] \\ 0, \ o.w. \end{cases} \qquad \text{式（4.1）}$$

Prediction Interval Average Width (PIAW)：预测区间平均宽度用于衡量预测区间的宽度。它表示在一系列预测中，所有预测区间的宽度的平均值。通常情况下，我们希望预测区间足够窄以提供对真实值的精确估计，但同时又要保持足够的置信水平。因此，PIAW 的目标是要尽可能小，同时满足预先设定的置信水平。实践中，为了消除不同数据集之间的影响，通常对 PIAW 除以样本量n和 y 的范围R作为标准化，计算 PINAW：

$$PINAW = \frac{1}{R \times n}\sum_{j=1}^{n} (\underline{y_i} - \overline{y_i}) \text{，} \qquad \text{式（4.2）}$$

这两个指标通常一起使用，以全面评估预测模型的性能。PICP 衡量了预测区间的准确性，而 PINAW 则衡量了预测区间的精度和宽度。理想情况下，我们希望得到一个高的 PICP（接近预先设定的置信水平），同时保持一个较小的 PINAW。

（二）模拟数据测试

利用随机变量，构造模拟数据：

$$y = x \cdot sin(x) + \varepsilon$$

为了使数据具有更好的随机性，ε采用异方差正态分布，$\varepsilon \sim N(0, V), (V-2) \sim U(0,1)$

y，x的散点图为：

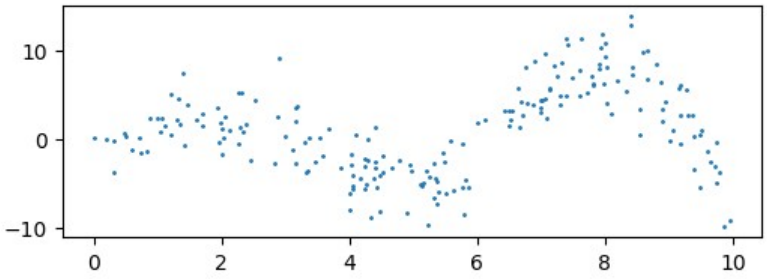

图 4-1 模拟数据

共在$x \in (0,10)$上取 200 个样本点。利用训练样本，预测(0,10)上全体 x 的 10%，50%和 90%分位数。

1. 线性回归

首先使用线性回归方法在测试数据上进行拟合。

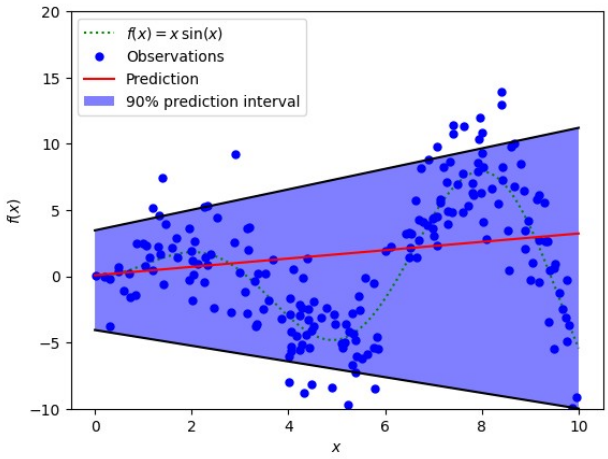

图 4-2 线性回归拟合测试数据

由于模拟数据是由非线性模型生成的，可以看到线性分位数回归无法对随机样本的模型进行拟合。接下来使用非线性模型进行拟合训练。

2. XGBoost 在模拟数据集上的表现

使用 XGBoost，对模拟数据进行拟合，并使用 CV 进行调参，交叉验证次数取 7。但利用训练数据对$x \in (0,10)$所有点进行预测，可以看到模型并没有收敛，无法准确地拟合各分位数。

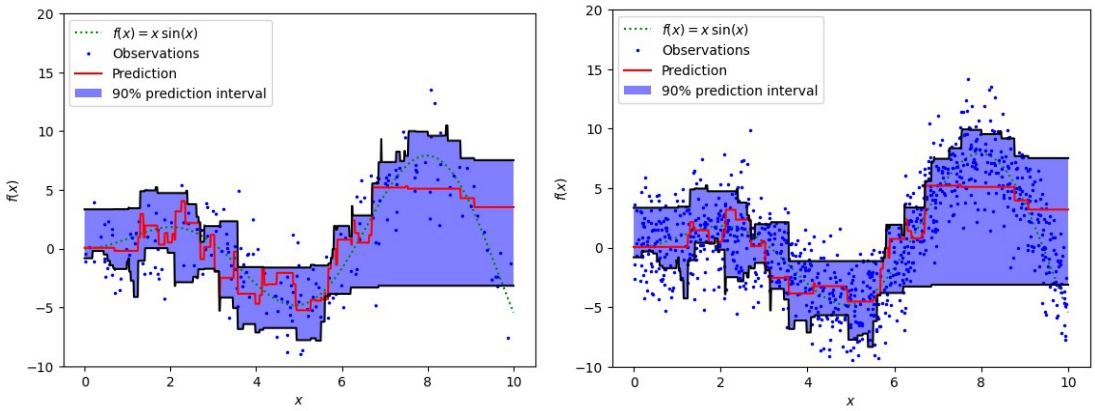

图 4-3 原始 XBoost 在模拟数据上的表现

Chen(2016)提出，二阶展开有效，是因为它近似地紧密上界了目标函数（假设步长很小）。因此，根据 Chen(2016)提出的技巧，尝试将为损失函数的二阶导数赋一个较小的值，以帮助模型更好的收敛。尝试不同的赋值结果如下。

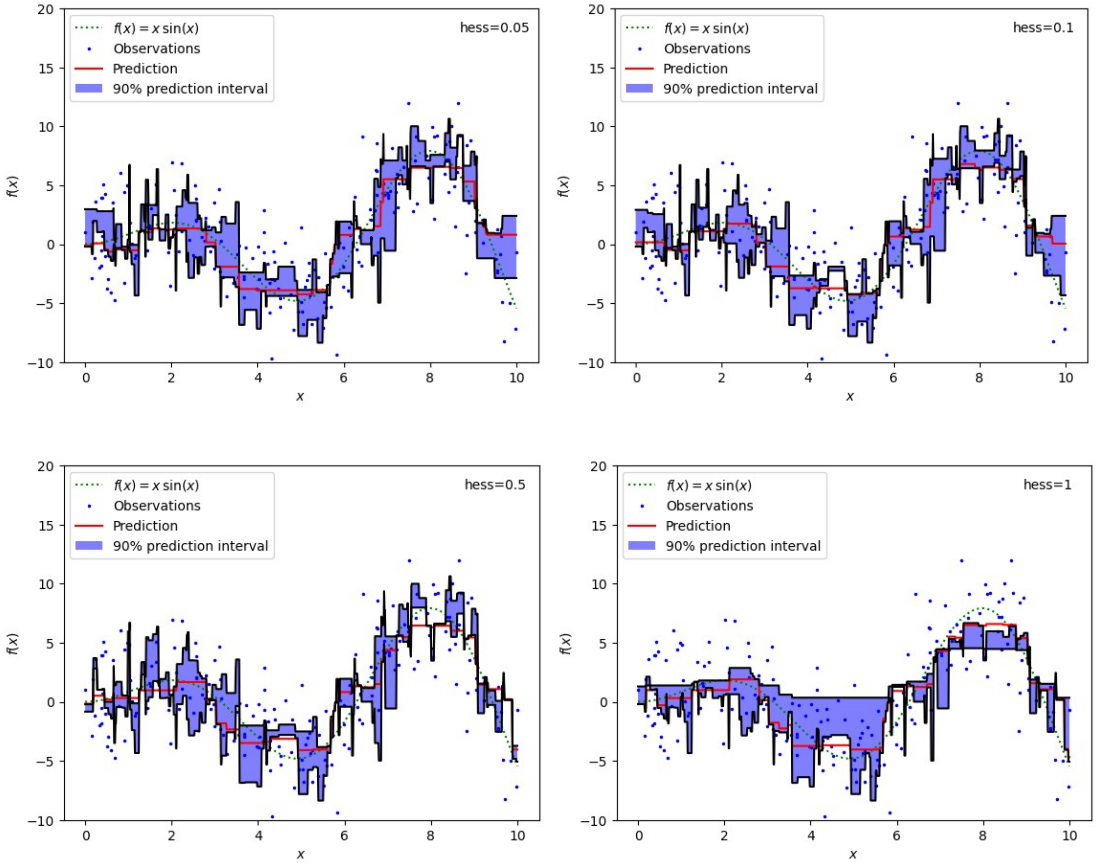

图 4-4 为 Hess 赋值并进行比较

3. 神经网络在模拟数据集上的表现

分别尝试直接使用 Pinball Loss 作为损失函数和 Huber Loss 作为损失函数，使用神经网络进行

训练。对神经网络训练和优化器参数进行如下设置：

表 4-1 神经网络超参数

参数	值
dropout	0.1
epochs	5000
隐藏层数	2
隐藏层维度	64
Adam Learning Rate	0.001
Adam Weight Decay	10^{-6}

二者在测试数据集上的表现可视化如下：

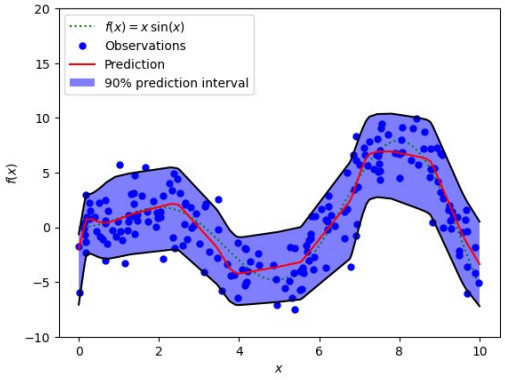

图 4-5 神经网络使用 HuberLoss 损失

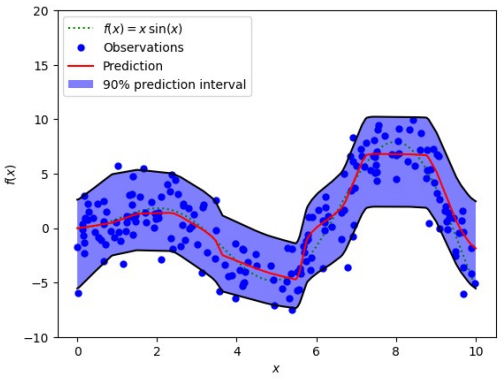

图 4-6 神经网络使用 PinballLoss 损失

可以看到，Huber Loss 表现出比较不错的效果，可以利用中位数对原模型进行拟合，并预测其分布区间（5%和 95%分位数）。

根据链式法则

$$\frac{\partial L}{\partial w}=\frac{\partial L}{\partial output}\cdot\frac{\partial output}{\partial hidden}\cdot\frac{\partial hidden}{\partial w} \qquad \text{式（4.3）}$$

理论上，Pinball Loss 在 0 处的不可导使得其无法利用反向传播进行梯度计算。但是对于 Pinball Loss，$\frac{\partial L}{\partial output}$仅在$y_i-\hat{y}_i=0$ 时不可导。在梯度下降的过程中如果没有出现这一情况，训练过程就可以正常进行。因此实践上即便不进行 Huber 平滑，也能进行分位数的预测。

（三）实际场景应用

1.连续性变量数据-风力发电

回归最早用于拟合连续型变量之间的关系。在实际应用场景中，对连续性指标进行预测或挖掘其分布是十分经典的问题。实验上，本文采用一组独特的现场气象观测数据和风能发电数据。数据集包含了从 2017 年 1 月 2 日开始的详细的每小时记录，直接从现场和风力涡轮装置收集数据。这个丰富的数据集提供了真实世界中各种天气条件和风能生产之间相互作用的见解，有助于理解气象变量与它们对风能发电的影响之间的动态关系的必要性，气象仪器测量了不同高度的温度、湿度、露点和风的特性，发电数据则记录了风力涡轮的输出。

该数据集适合于行业专家、研究人员和数据科学家，特别是那些探索可再生能源（特别是风能）者。它可以帮助开发发电的预测模型，研究环境对可再生能源的影响，并提高风力发电场的运营效率。

（1）描述性统计

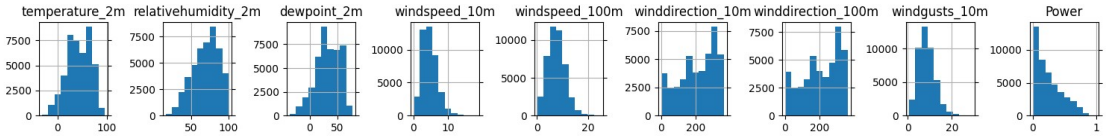

图 4-7 电力数据分布图

该数据集样本量为 43800，各变量分布如下。其中 Power 作为因变量，其余为自变量。各变量均为连续性变量。

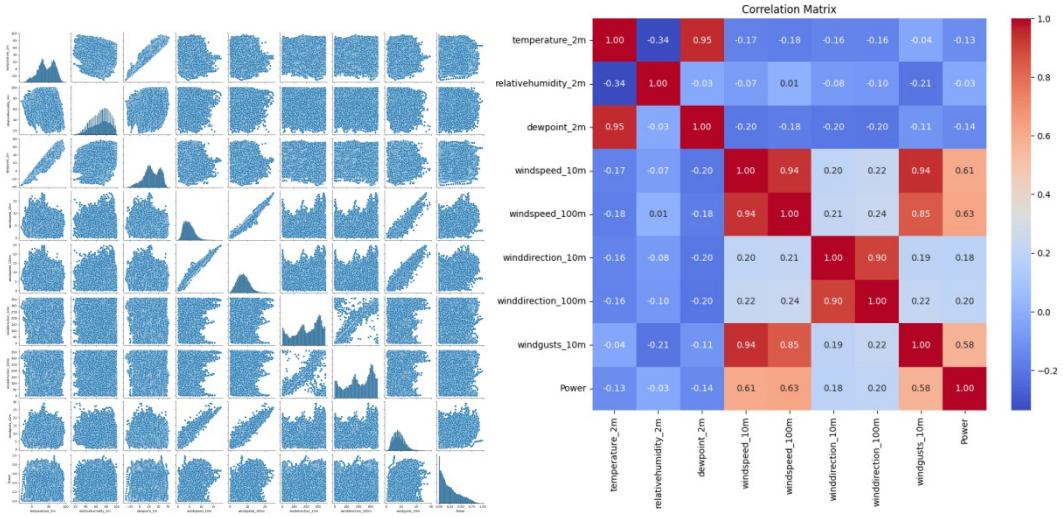

图 4-8 电力数据散点图和热力图

绘制两两变量之间的散点图和 Pearson 相关系数热力图，Power 与各自变量之间的没有显著的相关性，适于作为因变量进行回归挖掘。

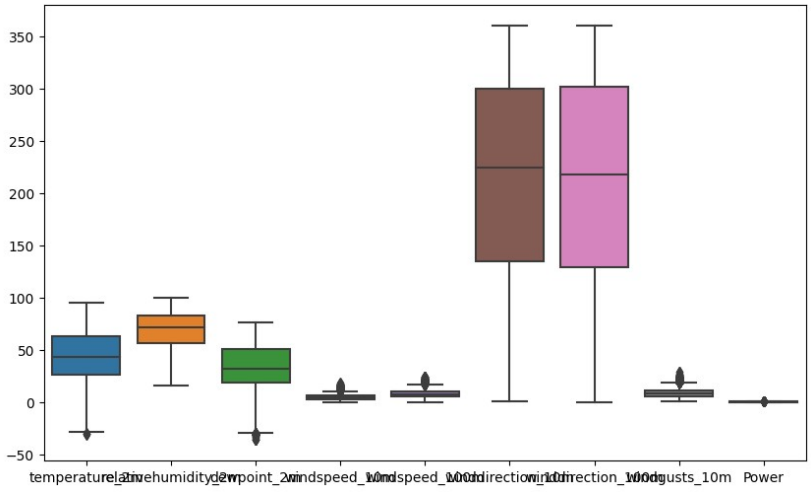

图 4-9 电力数据箱型图

根据箱线图，各变量分布相对集中，数量级具有一定差异，异常值较少，不进行进一步的缺失值处理或数据变换。

由于 Power 分布比较不均，将其进行对数变换并调整均值。

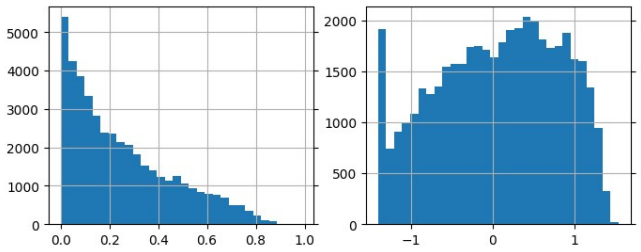

图 4-10 电力数据的数据变换

（2）模型训练及结果

训练过程

XGBoost 使用 cv 进行调参。根据模拟数据集上的表现，将二阶导设置为常数 0.1

由于数据量比较充足，神经网络训练迭代次数降低为 20，batch 大小设置为 2000；对数据进行 0.1，0.5 和 0.9 分位数预测，得到其 80%置信区间。其余参数与模拟数据集的训练保持一致。

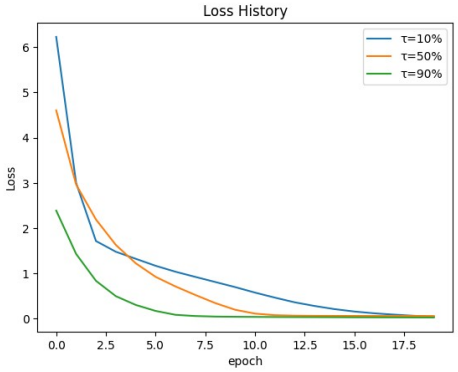

图 4-11 神经网络的训练过程 1

比较两个模型的结果，如下表。

表 4-2 电力数据拟合结果 1

指标	训练集 PICP	测试集 PICP	测试集 PIAW
		XGBoost	
数据	0.8013	0.6937	0.3889
		神经网络	
数据	0.701398	0.702968	0.414842

直接使用训练集样本进行预测，XGBoost 结果在指标上显然并不理想。尽管成功地限制了预测区间，但 PICP 与预估的目标 80%（0.9-0.1）有一定差距；神经网络的结果则理想很多，尤其测试集的 PICP 没有降低，这说明训练避免了过拟合的风险，且测试集 PIAW 较小，说明预测区间得到了较好的限制。

表 4-3 电力数据拟合结果 2

指标	$y < \hat{y}_{0.1}$ 占比	$y < \widehat{y_{0.5}}$ 占比	$y < \widehat{y_{0.9}}$ 占比
		XGBoost	
数据	14.20%	48.88%	83.34%
		神经网络	
数据	15.57%	50.25%	85.87%

进一步地，查看预测结果被真实分位数分割的比例，两个模型在对称性上的表现都比较好，中位数估计也相对准确。但显然 XGBoost 的分位限制效果更差。

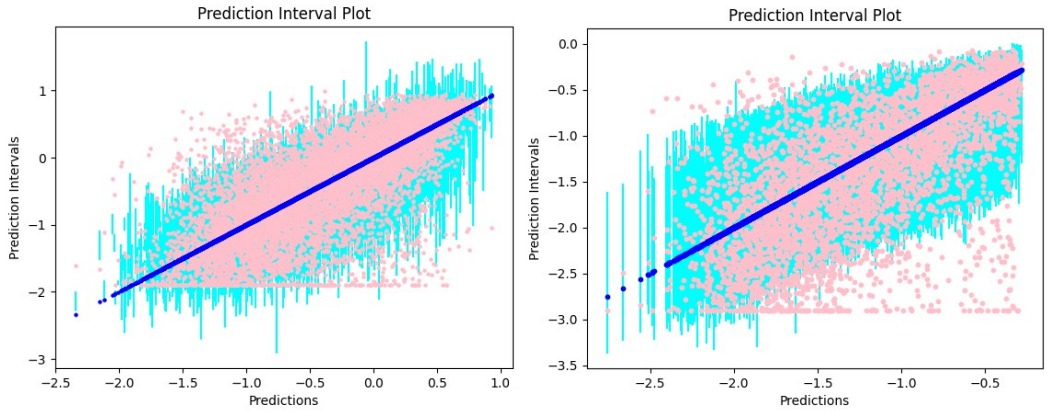

图 4-12 XGBoost（左）和神经网络（右）预测散点图 1

可视化散点图。蓝色部分是预测的 0.1-0.9 区间，蓝色点为预测 0.5 分位点，粉色点为真实值；XGBoost 对区间的预测较宽；二者受到真实值的偏态分布影响比较严重。

2.类别变量数据-碳排放

实际数据挖掘中大量场景涉及类别变量。实验上，本文采用一份综合多个调查形成的问卷数据，统计了个体的生活行为、习惯和碳排放相关资产数，并最终以碳排放量作为输出。通过该数据集可以探究个体不同特征与其碳排放总量之间的关联。

（1）描述性统计

该数据集中，含有十一个单类别变量（每个样本只属于一类）：

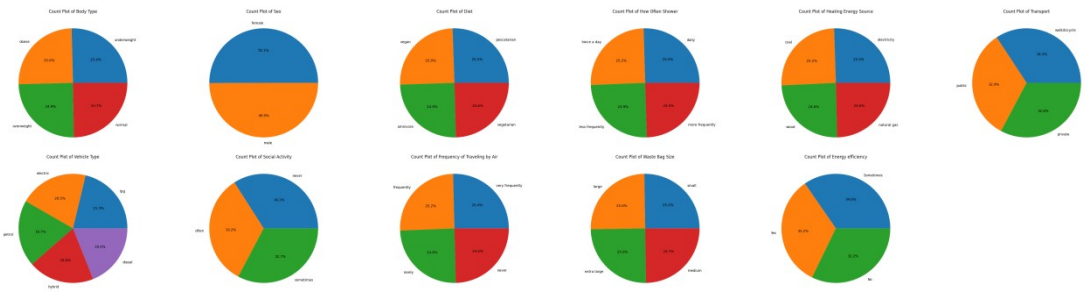

图 4-13 碳排放数据饼图

每个变量下各类别对应的因变量分布和均值如下：

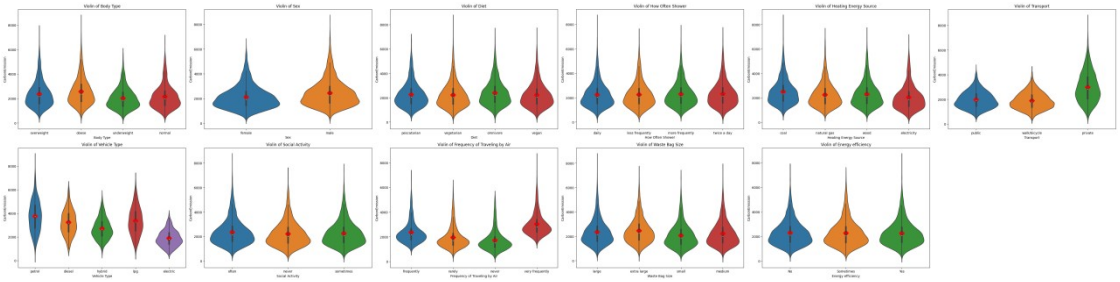

图 4-14 碳排放数据提琴图

此外，该数据集还具有两个多类别变量：

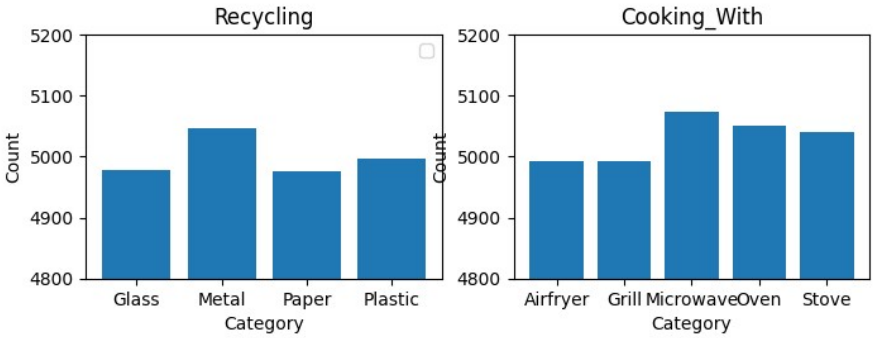

图 4-15 碳排放数据柱状图

以及七个连续型变量，其中 CarbonEmission 是因变量。

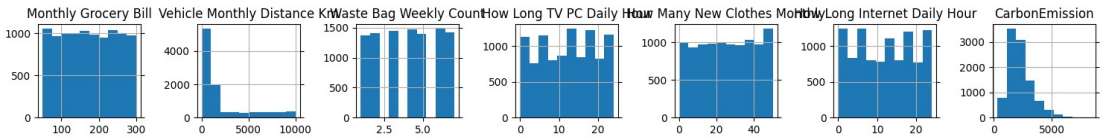

图 4-16 碳排放数据分布图

连续变量间的 Pearson 相关系数热力图和散点图如下：

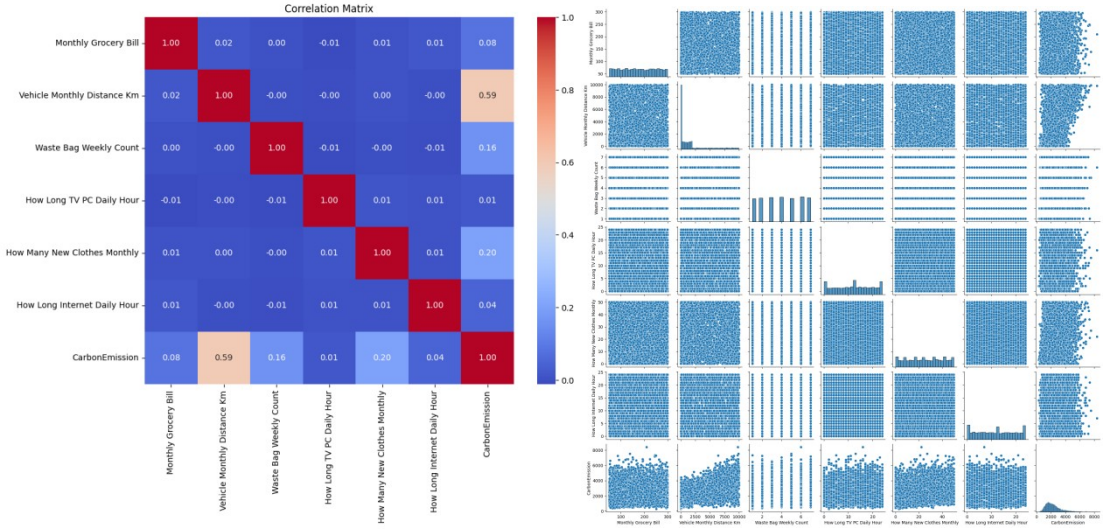

图 4-17 碳排放数据热力图和散点图

综合类别变量小提琴图和连续变量热力图，没有某个自变量对因变量有决定性作用，适于利用模型进行训练；此外，因变量 CarbonEmission 呈偏态分布，对其进行对数变换。为了便于模型识别，将类别变量转换为 one-hot 编码。

（2）模型训练及结果

XGBoost 使用 cv 进行调参。根据模拟数据集上的表现，将二阶导设置为常数 0.1。

由于数据量比较充足，神经网络训练迭代次数调整为 200，batch 大小设置为 256；对数据进行 0.1，0.5 和 0.9 分位数预测，得到其 80%置信区间。其余参数与模拟数据集的训练保持一致。

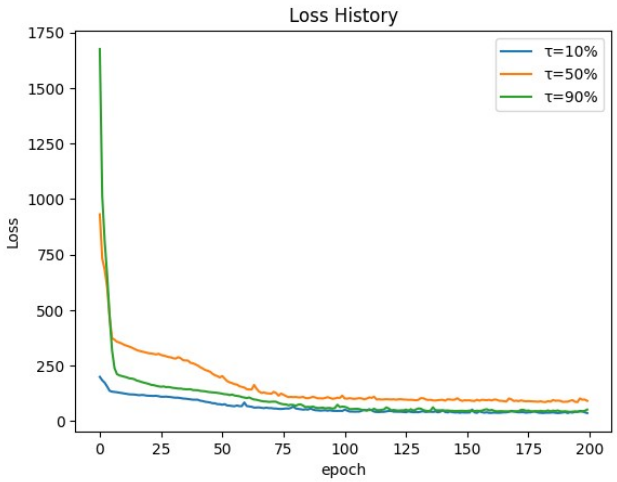

图 4-18 神经网络的训练过程 2

比较两个模型的结果，如下表。

表 4-4 碳排放数据拟合结果 1

指标	训练集 PICP	测试集 PICP	测试集 PIAW
	XGBoost		
数据	0.527625	0.4785	0.03925
	神经网络		
数据	0.8235	0.8035	0.0628

直接使用训练集样本进行预测，两个模型的 PICP 与预估的目标 80%（0.9-0.1）差距都比较大。其中，XGBoost 的指标全部偏小，估计比较保守；而神经网络指标偏大，估计过散。

表 4-5 碳排放数据拟合结果 2

指标	$y < \widehat{y_{0.1}}$ 占比	$y < \widehat{y_{0.5}}$ 占比	$y < \widehat{y_{0.9}}$ 占比
	XGBoost		
数据	0.2815	0.4915	0.749
	神经网络		
数据	0.097	0.387	0.9005

进一步地，查看预测结果被真实分位数分割的比例，XGBoost 表现出较好的对称性，上下分位数估计都是有限地超过和低于真实样本分位数；但神经网络则没有表现出对称性，几乎没有真实值低于估计的 0.1 和 0.5 分位值，说明分位数的估计过大。

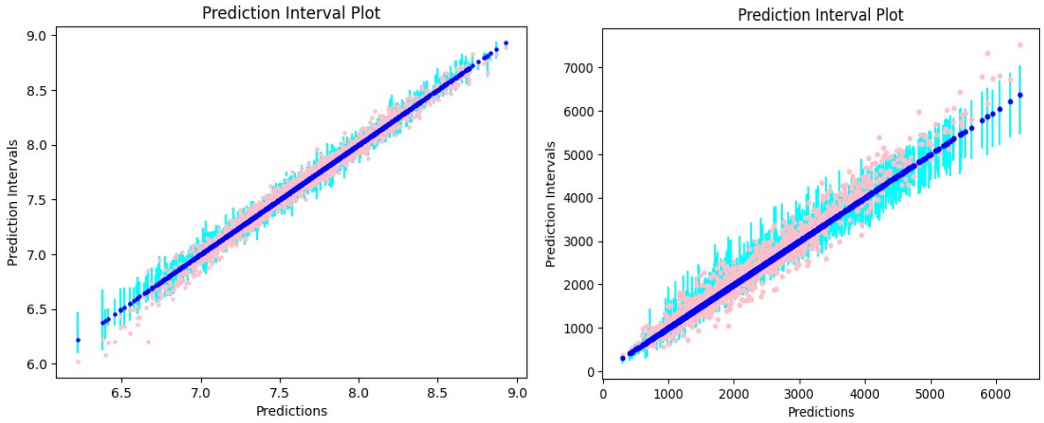

图 4-19 XGBoost（左）和神经网络（右）预测散点图 2

进行可视化，XGBoost 的上下分位数估计表现出较好的对称性，而神经网络的区间覆盖率更加准确。

第五章 结论

在探索分位数回归在非线性模型中的应用过程中，本文从分位数的定义及其在线性背景下的参数估计原理入手，进一步探讨了非线性模型，特别是 XGBoost 和神经网络，与分位数回归结合的方法。通过对 Pinball Loss 与 Huber Loss 的比较与应用，本文不仅解决了 Pinball Loss 在某些非线性模型中由于不可导性质导致无法拟合的问题，还通过实际数据集的实验验证了这些方法的有效性。

研究发现，分位数回归的线性背景理论基础可以被推广至非线性情况。此外 Huber Loss 作为 Pinball Loss 的近似，能够有效地应用于非线性模型中的分位数回归，这为分位数回归的进一步应用提供了新的途径。实验部分通过模拟数据测试和实际场景应用，如风力发电和碳排放的数据集，展示了在非线性模型中应用分位数回归的潜力。特别是，通过与线性模型的比较，展现了非线性模型在处理复杂数据时的优越性。

在风力发电数据集上，虽然 XGBoost 在预测区间的控制上不如神经网络，但通过调整模型参数和优化算法，两种模型均能较好地预测不同分位数的值但神经网络的表现更为突出，显示了非线性模型在预测连续性变量时的有效性。碳排放数据集的分析则比较了类别变量数据中两个模型的表现，尽管从评价指标上看二者都有待提高，XGBoost 的决策树模型根基使其更好的刻画了对称性和更高的效率，神经网络的区间覆盖率则更加准确。

总体而言，本文的研究不仅深化了分位数回归理论的理解，还拓宽了其在非线性模型中的应用范围。未来的研究可以进一步探讨不同非线性模型结合分位数回归的性能，以及如何优化模型参数和损失函数以适应更广泛的应用场景。此外，分位数回归在处理数据分布偏斜或含有异常值的情况下，需要对数据进行进一步处理。

参考文献

李航. 统计学习方法[M]. 清华大学出版社, 2012.

邱锡鹏. 神经网络与深度学习[M]. 机械工业出版社, 2020.

Cannon A J. Quantile regression neural networks: Implementation in R and application to precipitation downscaling[J]. Computers & geosciences, 2011, 37(9): 1277-1284.

Chen C. A finite smoothing algorithm for quantile regression[J]. Journal of Computational and Graphical Statistics, 2007, 16(1): 136-164.

Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.

Davino C, Furno M, Vistocco D. Quantile regression: theory and applications[M]. John Wiley & Sons, 2013.

Duman M. Individual Carbon Footprint Calculation[DB/OL].https://www.kaggle.com/datasets/dumanmesut/individual-carbon-footprint-calculation/data.2023

Jantre S R, Bhattacharya S, Maiti T. Quantile regression neural networks: a bayesian approach[J]. Journal of Statistical Theory and Practice, 2021, 15(3): 68.

Koenker R, Bassett Jr G. Regression quantiles[J]. Econometrica: journal of the Econometric Society, 1978: 33-50.

Koenker R, Park B J. An interior point algorithm for nonlinear quantile regression[J]. Journal of Econometrics, 1996, 71(1-2): 265-283.

März A. XGBoostLSS--An extension of XGBoost to probabilistic forecasting[J]. arXiv preprint arXiv:1907.03178, 2019.

Rahim M.Wind Power Generation Data - Forecasting[DB/OL].https://www.kaggle.com/datasets/mubashirrahim/wind-power-generation-data-forecasting.2024

Smelyakov K, Klochko O, Dudar Z. Building Quantile Regression Models for Predicting Traffic Flow[C]//COLINS (1). 2023: 117-132.

Taylor J W. A quantile regression neural network approach to estimating the conditional density of multiperiod returns[J]. Journal of forecasting, 2000, 19(4): 299-311.

Tyurin A S, Saraev P V. Construction of quantile regression using natural gradient descent[J]. Applied Mathematics and Control Sciences, 2023 (2): 43-52.

Xu Q, Deng K, Jiang C, et al. Composite quantile regression neural network with applications[J]. Expert Systems with Applications, 2017, 76: 129-139.

Yin X, Fallah-Shorshani M, McConnell R, et al. Quantile Extreme Gradient Boosting for Uncertainty Quantification[J]. arXiv preprint arXiv:2304.11732, 2023.