Problem 2: Linear Regression with Huber Loss

Setup

Residuals: $r_i(w) = y_i - w^\top x_i$, with $w \in \mathbb{R}^d$, $x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$.

Huber loss (with parameter $\delta > 0$):

$$\ell_\delta(r) = \begin{cases} \frac{1}{2}r^2 & |r| \le \delta, \\ \delta\left(|r| - \frac{1}{2}\delta\right) & |r| > \delta. \end{cases}$$

Objective:

$$L_\delta(w) = \sum_{i=1}^{n} \ell_\delta(r_i(w)).$$

Part 1: Gradient derivation (6 pts)

We need $\nabla_w L_\delta(w)$. The loss is a sum over samples, so the gradient is the sum of the gradients of $\ell_\delta(r_i(w))$. By the chain rule, the contribution from sample i is

$$\nabla_w \left[\ell_\delta(r_i(w))\right] = \frac{\mathrm{d}\ell_\delta}{\mathrm{d}r}(r_i) \cdot \nabla_w r_i(w).$$

The residual is $r_i(w) = y_i - w^\top x_i$, so

$$\nabla_w r_i(w) = -x_i.$$

The derivative of the Huber loss with respect to r is piecewise: for $|r| \le \delta$ we have $\ell_\delta(r) = \frac{1}{2}r^2$, so $\frac{\mathrm{d}\ell_\delta}{\mathrm{d}r} = r$; for $|r| > \delta$ we have $\ell_\delta(r) = \delta(|r| - \frac{1}{2}\delta)$, so $\frac{\mathrm{d}\ell_\delta}{\mathrm{d}r} = \delta \cdot \operatorname{sign}(r)$. Thus

$$\frac{\mathrm{d}\ell_\delta}{\mathrm{d}r}(r) = \begin{cases} r & |r| \le \delta, \\ \delta \operatorname{sign}(r) & |r| > \delta. \end{cases}$$

Substituting into the chain rule and summing over i,

$$\boxed{\nabla_w L_\delta(w) = -\sum_{i=1}^{n} \frac{\mathrm{d}\ell_\delta}{\mathrm{d}r}(r_i(w))\, x_i,}$$

with the piecewise rule above applied to each $r_i(w)$. In explicit piecewise form:

$$\nabla_w L_\delta(w) = -\sum_{i\colon |r_i(w)| \le \delta} r_i(w)\, x_i \; - \sum_{i\colon |r_i(w)| > \delta} \delta \operatorname{sign}(r_i(w))\, x_i.$$

Part 2: Optimal w: closed form or not? (4 pts)

Minimizing $L_\delta(w)$ does **not** admit a single closed-form solution like ordinary least squares (OLS).

In OLS we minimize $\frac{1}{2}\sum_i (y_i - w^\top x_i)^2$. Setting the gradient to zero gives the normal equations $X^\top X w = X^\top y$, which are linear in w, so we get an explicit solution $w^* = (X^\top X)^{-1} X^\top y$ (when $X^\top X$ is invertible). The key is that the objective is quadratic in w, so $\nabla_w L$ is linear in w.

For Huber loss, the gradient we derived is

$$\nabla_w L_\delta(w) = -\sum_i \psi(r_i(w))\, x_i, \qquad \psi(r) = \begin{cases} r & |r| \le \delta, \\ \delta\, \mathrm{sign}(r) & |r| > \delta. \end{cases}$$

Setting $\nabla_w L_\delta = 0$ therefore gives

$$\sum_i \psi(y_i - w^\top x_i)\, x_i = 0.$$

The function ψ is nonlinear (piecewise linear with a kink). The residuals $r_i(w) = y_i - w^\top x_i$ depend on w, and which branch of ψ applies to each i depends on w. So we get a **nonlinear** system in w: we cannot rearrange this into a single linear system $Aw = b$ with a fixed matrix A. That is what breaks the normal-equations approach.

In practice we minimize $L_\delta(w)$ numerically. One standard approach is **iteratively reweighted least squares (IRLS)**: at each iteration we treat the current residuals as fixed, approximate the Huber objective by a weighted least-squares problem (with weights that depend on the current r_i), solve that for the next w, and repeat. Another standard approach is to use a general-purpose optimizer such as **gradient descent** or **L-BFGS** on $L_\delta(w)$, using the gradient we derived.

Part 3: Add L2 regularization (3 pts)

Define

$$\tilde{L}(w) = \sum_{i=1}^{n} \ell_\delta(r_i(w)) + \frac{\lambda}{2}\|w\|_2^2.$$

The gradient of $\frac{\lambda}{2}\|w\|_2^2$ is λw. So

$$\boxed{\nabla_w \tilde{L}(w) = -\sum_{i=1}^{n} \frac{\mathrm{d}\ell_\delta}{\mathrm{d}r}(r_i(w))\, x_i \;+\; \lambda w,}$$

with the same piecewise definition of $\frac{\mathrm{d}\ell_\delta}{\mathrm{d}r}$ as in Part 1.

How λ changes the solution: Larger λ penalizes large $\|w\|$, so the solution is shrunk toward zero and has smaller norm. That reduces overfitting in high-dimensional settings (fewer samples than dimensions or many correlated features) by favoring simpler models. Under outliers, the quadratic part of Huber already limits their influence; adding $\lambda\|w\|^2$ further stabilizes the solution by preventing the fit from being pulled too strongly by any subset of points, so the regularized minimizer tends to be more robust than an unregularized one when the data are noisy or contain outliers.

Part 4: ML pipeline reflection (2 pts)

In the 3-step recipe (define objective → optimize → evaluate), Huber changes both optimization and evaluation compared to squared loss.

Optimization: The objective is no longer quadratic in w, so we cannot solve for w in closed form. We rely on iterative methods (e.g. gradient descent or IRLS). The gradient is still straightforward to compute and is continuous (subgradient at $|r| = \delta$ can be defined consistently), so optimization is well behaved.

Evaluation robustness: Squared loss heavily penalizes large residuals, so a few outliers can dominate the loss and distort both the learned w and the reported loss value. Huber loss behaves like squared loss for small $|r|$ and like $|r|$ for large $|r|$, so outliers have bounded influence on the objective and on the fit. So (i) optimization requires an iterative procedure but remains tractable, and (ii) evaluation (and the fitted model) are more robust to outliers than with squared loss alone.