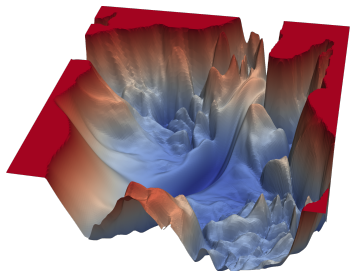


Introduction to Machine Learning

Advanced Risk Minimization Properties of Loss Functions



Learning goals

- Statistical properties
- Robustness
- Numerical properties
- Some fundamental terminology

THE ROLE OF LOSS FUNCTIONS

Why should we care about the choice of the loss function $L(y, f(\mathbf{x}))$?

- **Statistical** properties: choice of loss implies statistical assumptions about the distribution of $y \mid \mathbf{x} = \mathbf{x}$ (see *maximum likelihood estimation vs. empirical risk minimization*).
- **Robustness** properties: some loss functions are more robust towards outliers than others.
- **Numerical** properties: the computational complexity of

$$\arg \min_{\theta \in \Theta} \mathcal{R}_{\text{emp}}(\theta)$$

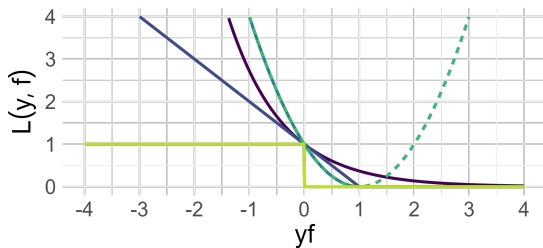
is influenced by the choice of the loss function.



SOME BASIC TERMINOLOGY

Classification losses are usually expressed in terms of the **margin**:

$$\nu := y \cdot f(\mathbf{x}).$$

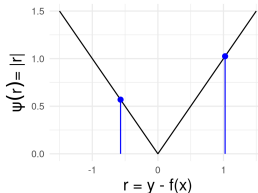
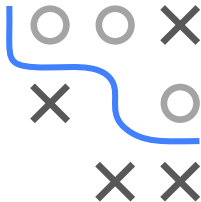


- Exponential
- Hinge
- Squared hinge
- Squared (scores)
- 0-1

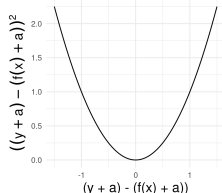


SOME BASIC TERMINOLOGY

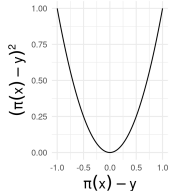
- Regression losses often only depend on the **residuals** $r := y - f(\mathbf{x})$.
- Losses are called **symmetric** if $L(y, f(\mathbf{x})) = L(f(\mathbf{x}), y)$.
- A loss is **translation-invariant** if $L(y + a, f(\mathbf{x}) + a) = L(y, f(\mathbf{x}))$, $a \in \mathbb{R}$.
- A loss is called **distance-based** if
 - it can be written in terms of the residual, i.e., $L(y, f(\mathbf{x})) = \psi(r)$ for some $\psi : \mathbb{R} \rightarrow \mathbb{R}$, and
 - $\psi(r) = 0 \Leftrightarrow r = 0$.



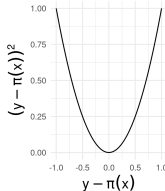
Distance-based: L1



Translation-invariant: L2



Symmetric: Brier score



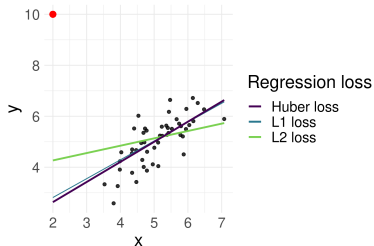
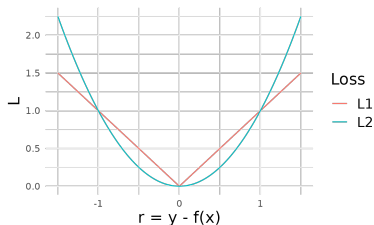
ROBUSTNESS

Outliers (in y) have large residuals $r = y - f(\mathbf{x})$. Some losses are more affected by large residuals than others. If loss goes up superlinearly (e.g. L2) it is not robust, linear (L1) or even sublinear losses are more robust.

$y - \hat{f}(\mathbf{x})$	L1	L2	Huber ($\epsilon = 5$)
1	1	1	0.5
5	5	25	12.5
10	10	100	37.5
50	50	2500	237.5

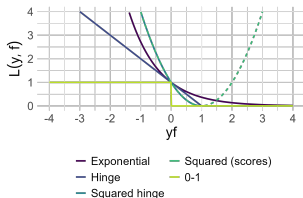
As a consequence, a model is less influenced by outliers than by “inliers” if the loss is **robust**.

Outliers e.g. strongly influence L2.



NUMERICAL PROPERTIES: SMOOTHNESS

- **Smoothness** of a function is a property measured by the number of continuous derivatives.
- Derivative-based optimization requires smoothness of the risk $\mathcal{R}_{\text{emp}}(\theta)$
 - If loss is unsmooth, we might have to use derivative-free optimization (or worse, in case of 0-1)
 - Smoothness of $\mathcal{R}_{\text{emp}}(\theta)$ not only depends on L , but also requires smoothness of $f(\mathbf{x})$!



Squared loss, exponential loss and squared hinge loss are continuously differentiable.
Hinge loss is continuous but not differentiable.
0-1 loss is not even continuous.

NUMERICAL PROPERTIES: CONVEXITY

- A function $\mathcal{R}_{\text{emp}}(\theta)$ is convex if

$$\mathcal{R}_{\text{emp}}\left(t \cdot \boldsymbol{\theta} + (1-t) \cdot \tilde{\boldsymbol{\theta}}\right) \leq t \cdot \mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) + (1-t) \cdot \mathcal{R}_{\text{emp}}(\tilde{\boldsymbol{\theta}})$$

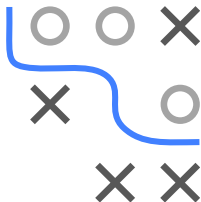
$$\forall t \in [0, 1], \theta, \tilde{\theta} \in \Theta$$

(strictly convex if the above holds with strict inequality).

- In optimization, convex problems have a number of convenient properties. E.g., all local minima are global.

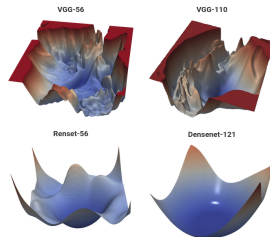
→ strictly convex function has at most **one** global min (uniqueness).

- For $\mathcal{R}_{\text{emp}} \in \mathcal{C}^2$, \mathcal{R}_{emp} is convex iff Hessian $\nabla^2 \mathcal{R}_{\text{emp}}(\theta)$ is psd.



NUMERICAL PROPERTIES: CONVEXITY

- Convexity of $\mathcal{R}_{\text{emp}}(\theta)$ depends both on convexity of $L(\cdot)$ (given in most cases) and $f(\mathbf{x} \mid \theta)$ (often problematic).
- If we model our data using an exponential family distribution, we always get convex losses
 - For $f(\mathbf{x} \mid \theta)$ linear in θ , linear/logistic/softmax/poisson/. . . regression are convex problems (all GLMs)!



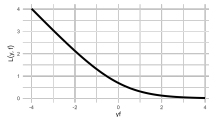
Li et al., 2018: *Visualizing the Loss Landscape of Neural Nets*. The problem on the bottom right is convex, the others are not (note that very high-dimensional surfaces are coerced into 3D here).

NUMERICAL PROPERTIES: CONVERGENCE

In case of **complete separation**, optimization might even fail entirely, e.g.:

- Margin-based loss that is strictly monotonically decreasing in $y \cdot f$, e.g., **Bernoulli loss**:

$$L(y, f(\mathbf{x})) = \log(1 + \exp(-yf(\mathbf{x})))$$



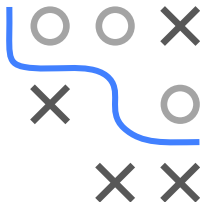
- f linear in θ , e.g., **logistic regression** with $f(\mathbf{x} | \theta) = \theta^T \mathbf{x}$
- Data perfectly separable by our learner, so we can find θ :

$$y^{(i)} f(\mathbf{x}^{(i)} | \theta) = y^{(i)} \theta^T \mathbf{x}^{(i)} > 0 \quad \forall \mathbf{x}^{(i)}$$

- Can now construct a strictly better θ

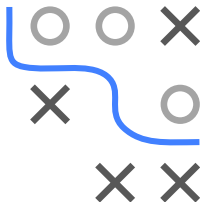
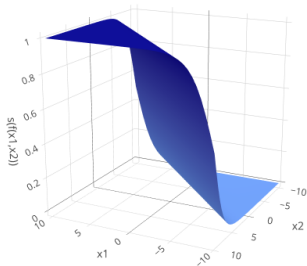
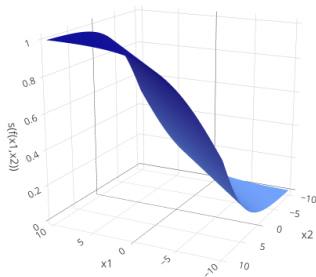
$$\mathcal{R}_{\text{emp}}(2 \cdot \theta) = \sum_{i=1}^n L(2y^{(i)} \theta^T \mathbf{x}^{(i)}) < \mathcal{R}_{\text{emp}}(\theta)$$

- As $\|\theta\|$ increases, sum strictly decreases, as argument of L is strictly larger
- We can iterate that, so there is no local (or global) optimum, and no numerical procedure can converge



NUMERICAL PROPERTIES: CONVERGENCE / 2

- Geometrically, this translates to an ever steeper slope of the logistic/softmax function, i.e., increasingly sharp discrimination:



- In practice, data are seldomly linearly separable and misclassified examples act as counterweights to increasing parameter values.
- Besides, we can use **regularization** to encourage convergence to robust solutions.