Entropy of Gaussian

$$H(x) = -\int_{-\infty}^{\infty} p(x)\log[p(x)]\,dx$$

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

$$\log p(x) = \log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) - \left(\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$\Rightarrow H(x) = -\int_{-\infty}^{\infty} p(x)\left[\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) - \left(\frac{(x-\mu)^2}{2\sigma^2}\right)\right]dx$$

$$= -\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right)\int_{-\infty}^{\infty} p(x)\,dx + \frac{1}{2\sigma^2}\int_{-\infty}^{\infty}(x-\mu)^2 p(x)\,dx$$

$$= -\left[-\frac{1}{2}\log(2\pi) + (-1)\log\sigma\right] + \frac{1}{2\sigma^2}\int_{-\infty}^{\infty}(x-\mu)^2 p(x)\,dx$$

$$= \frac{1}{2}\log(2\pi) + \log\sigma + \frac{1}{2\sigma^2}\int_{-\infty}^{\infty}(x-\mu)^2 p(x)\,dx$$

$$\int_{-\infty}^{\infty}(x-\mu)^2 p(x)\,dx = E[(x-\mu)^2] = \sigma^2$$

$$= \frac{1}{2}\log(2\pi) + \log\sigma + \frac{\sigma^2}{2\sigma^2}$$

$$= \frac{1}{2}\log(2\pi) + \log\sigma + \frac{1}{2}$$

$$= \frac{1}{2}\log(2\pi) + \log\sigma + \frac{1}{2}\log e$$

$$\Rightarrow H(x) = \frac{1}{2}\left(\log(2\pi) + 2\log\sigma + \log e\right)$$

$$\boxed{H(x) = \frac{1}{2}\log(2\pi e\sigma^2)}$$

Regression Loss

1. Mean Squared Error

$$MSE = \left(\frac{1}{n}\right)\sum_{i=0}^{n}(\hat{y}_i - y_i)^2$$

Sensitive to outliers

2. Mean Absolute Error

$$MAE = \frac{1}{n}\sum_{i=0}^{n}|\hat{y}_i - y_i|$$

Not sensitive to outliers, not differentiable at zero

3. Huber loss → MLE + MAE.

δ: Huber's threshold. linear for large values, quadratic for smaller ones

$$L = \begin{cases} \frac{1}{2}(\hat{y}-y_i)^2 & \text{if } (\hat{y}-y_i) \le \delta \\ \delta|\hat{y}-y_i| - \frac{1}{2}\delta^2 & \text{if } (\hat{y}-y_i) > \delta \end{cases}$$

Small δ → MAE

large δ → MSE

Problem: Threshold effect; not smooth

4. log-cosh loss

Smoother version of Huber's loss

$$L = \sum_{i=1}^{n}\log(\cosh(\hat{y}-y_i))$$

→ No critical threshold

→ Smooth diff function that handles outliers better than MSE

$$\cosh(x) = \frac{e^x + e^{-x}}{2}$$

Binary Classification Loss

1. Avg. Zero-One Error

$$L_{0-1}(\hat{y}_i, y_i) = \begin{cases} 0 & \hat{y}_i = y_i \\ 1 & \hat{y}_i \neq y_i \end{cases}$$

$$L_{0-1}^{avg} = \frac{1}{n}\sum_{i=1}^{n} L_{0-1}(\hat{y}_i, y)$$

→ non-differentiable, non-continuous, no gradients

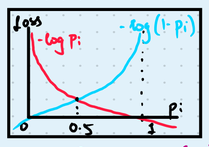
→ no margin info → treats all misclassifications equally

→ can help determine optimum decision threshold

2. Log loss / Binary cross entropy / Logistic

$$L_i = -y_i \log p_i - (1-y_i)\log(1-p_i)$$

where $p_i = P(y_i = 1)$

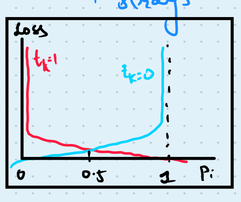


→ lower = better

→ penalises harshly when predicted probability strays

3. Brier Score

$$BS = \frac{1}{N}\sum_{i=1}^{n}(t_k - o_k)^2$$



Much gentler penalty

Multiclass Classification Loss

1. Cross Entropy loss

$$L_{CE} = -\sum_{i=1}^{n} y_i \log(\hat{p}_i)$$

2. Softmax

$$L_i = \frac{e^{y_i}}{\sum_{j=1}^{n} e^{y_i}}$$

Converts model output into probabilities

3. Categorical cross entropy (many classes)

$$L_{cat} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{p}_{ij})$$

cost func: iterates over samples

loss func: Iterates over all classes

sparse → converts integer to one-hot

N → number of samples

$\vec{y}$ = {1 for some target i, 0 elsewhere}

$\vec{p}$ = probability estimates for every class

Multiclass vs multilabel

- many classes, one correct → CCE
- many classes, many correct → BCE

Hinge Loss

Hinge opens in one direction → loss increases only if certain criterion not satisfied → does not distinguish level of satisfaction

$$L = \sum_{i=0, i\neq c}^{N-1}\max(0, y_i - y_c + m)$$

if $y_i - y_c > m$ then loss changes

if $y_i - \underbrace{y_c}_{GT} < m$: no change to loss, 0

typically $m = 1$

Squared hinge loss

$$L = \frac{1}{N}\sum_{i=0}^{N-1}\max(0, 1 - y_i\cdot\hat{y}_i)^2$$

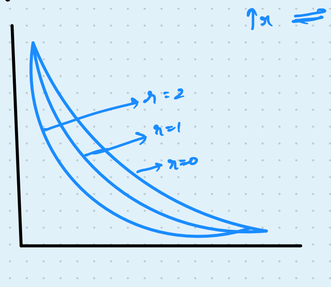
Focal Loss: Imbalanced Datasets

$$L = -(1-y_t)^{\gamma}\log y_t \qquad y_t = P(\text{class } 1)$$

where $y_t = \begin{cases} y & \text{if GT is class 1} \\ 1-y & \text{if GT is class 0} \end{cases}$ (confidence in class)

Effect of γ

$\gamma \uparrow$ ⇒ $\uparrow$ steepness ⇒ harsher penalty



Contrastive Loss

Open-set task → model needs to generalise classes → pull similar instances together, push dissimilar away

self-supervised / pretext task → supervised loss function

$$L(W(Y,\vec{X_1},\vec{X_2})^i) = (1-Y)L_S(D_W^i) + YL_D(D_W^i)$$

L_S: loss for similarity; L_D: loss func for dissimilarity

$\|G_W(\vec{X_1}) - G_W(\vec{X_2})\|$: mapping func Euclidean

$$L(W,Y,\vec{X_1},\vec{X_2}) = (1-Y)\frac{1}{2}(D_W)^2 + Y\frac{1}{2}\left[\max(0, m - D_W)\right]^2$$

Triplet loss

$$L = \max(d(a,p) - d(a,n) + m, 0)$$

tolerates intra-class variance

less greedy → does not change ve cluster if no -ve interferes

Contrastive loss may reach local minima earlier because it only looks at margin for dissimilar pairs

Focal Loss → imbalanced learning

For ith sample:

$$L(y^{(i)}) = \begin{cases} -\log y^{(i)} & \text{if } gt = 1 \\ -\log(1-y^{(i)}) & \text{if } gt = 0 \end{cases}$$

weight results by difference from gt

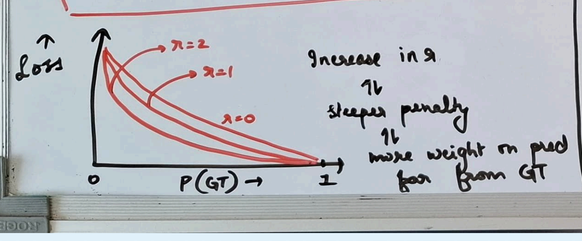
GT=1 → 1-y

GT=0 → 0-y = y

$$L(y) = \begin{cases} -(1-y)^{\gamma}\log y & \text{if } gt = 1 \\ -y^{\gamma}\log(1-y) & \text{if } gt = 0 \end{cases}$$

$$L(y) = -(1-y_t)^{\gamma}\log y_t$$

where $y_t = \begin{cases} y & \text{if } gt = 1 \\ 1-y & \text{if } gt = 0 \end{cases}$



increase in γ ⇒ steeper penalty ⇒ more weight on pred far from GT

Contrastive Loss → self-supervised learning

pull similar samples together, push dissimilar ones away

$$L(W,Y,\vec{X_1},\vec{X_2}) = (1-Y)L_S(D_W) + YL_D(D_W) \qquad Y = \begin{cases} 0 & \text{if similar} \\ 1 & \text{if not} \end{cases}$$

Y=0 gives similar L_S; Y=1 gives dissimilar L_D

L_S: loss function for similar pair $\vec{X_1}, \vec{X_2}$

as $D_W \uparrow$, $L_S \uparrow$

L_D: " " " dissimilar pair

as $D_W \uparrow$, $L_D \downarrow$ but only until margin m

after certain distance m, no point pushing further apart

D_W: distance between $\vec{X_1}, \vec{X_2}$; any metric

$$D_W(X_1, X_2) = \|G(X_1) - G(X_2)\|_2$$

where $G(\cdot)$ → mapping function to transformed input space where distance has meaning

Say we use Euclidean distance

$$L_S(D_W) = \frac{1}{2}D_W^2$$

$$L_D(D_W) = \frac{1}{2}\left[\max(0, m - D_W)\right]^2$$

once D_W becomes greater than m, no penalty

$$L[W,Y,X_1,X_2] = (1-Y)\left[\frac{1}{2}D_W^2\right] + Y\left[\frac{1}{2}(\max(0, m-D_W))^2\right]$$

Triplet Loss

Similar points closer together than dissimilar only by some margin

$$L = \max(0, d(a,p) - d(a,n) + m)$$

a: same label as anchor; n: diff label

→ Uses three points: anchor, positive & negative

→ From above:

Once distance from a to p differs from distance from a to n by the margin m or more, NO PENALTY.

→ Allows it to maintain some intra-class variance

→ Considers margin for both +ve and -ve; contrastive considers only for -ve

Contrastive for +ve: L_S

$L_S = \frac{1}{2}D_W^2$

minimising L_S pushes D_W to zero

triplet only pushes to margin m

→ less greedy; more robust and allows it to learn more discriminative features