Machine Learning: Core Formulas

A one-page reference of the equations behind training a neural network.

A classifier maps an input vector x to class probabilities. The raw scores (logits) z are turned into a distribution with the **softmax** function, where the probability of class i is

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Loss

Training minimizes the **cross-entropy** loss between the predicted distribution and the one-hot target y:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{c=1}^{K} y_{i,c} \cdot \log \hat{Y}_{i,c}$$

Gradient descent

Parameters θ are updated against the gradient of the loss, scaled by the learning rate η:

$$\theta_{t+1} = \theta_t - \eta \, \nabla_\theta \, \mathcal{L}(\theta_t)$$

Mean squared error, used for regression, is $\mathrm{MSE} = (1/n)\sum (y_i - \hat{y}_i)^2$.

Typical hyperparameters

Symbol	Name	Value
η	Learning rate	3×10^{-4}
B	Batch size	32
λ	Weight decay	0.01