Machine Learning Group

Ovidiu Calin, October 6, 2017

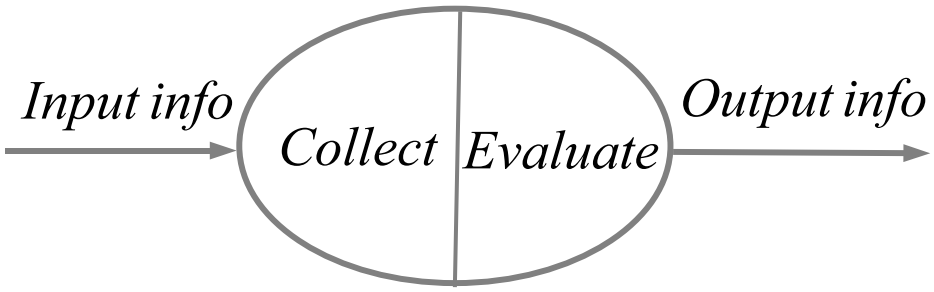
Error Functions

In the learning process, all neural networks are subject to minimize a certain objective function, which is also known under the equivalent names of:

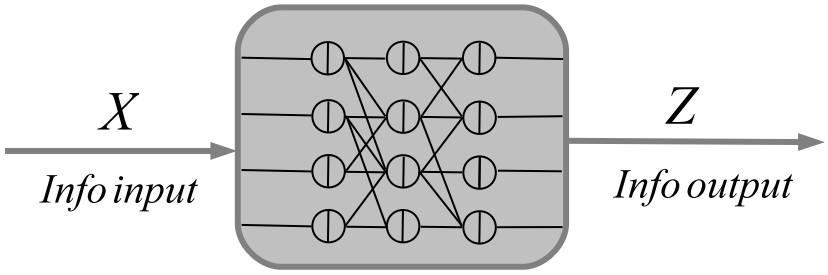
- *cost function*
- *loss function*
- *error function*.

In the following we shall describe some of the most familiar cost functions used in neural networks.

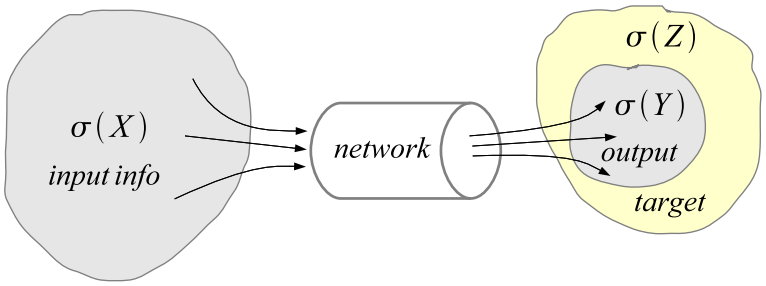
Neuron as an information processing unit.



Information flows in and out of a feed-forward neural network.



Input, output and desired information fields. The error function measures the discrepancy between the target and output information.



The supremum error function

Continuous bounded inputs: $x \in [0, 1]$

Goal: learn a given continuous function $\phi : [0, 1] \to \mathbb{R}$.

Let f_w be the input-output mapping of the network

The cost function is

$$C(w) = \sup_{x \in [0,1]} |f_w(x) - \phi(x)|.$$

For all practical purposes, when the target function is known at n points

$$z_1 = \phi(x_1), z_2 = \phi(x_2), \ldots, z_n = \phi(x_n)$$

the aforementioned cost function becomes

$$C(w) = \max_{1 \leq i \leq n} |f_w(x_i) - z_i|.$$

The L^2- error function

Input of the network: $x \in [0,1]$

Target function $\phi : [0,1] \to \mathbb{R}$ square integrable.

If f_w is the input-output mapping, the cost function measures the distance in the L^2-norm between the output and target

$$C(w) = \int_0^1 \big(f_w(x) - \phi(x)\big)^2 \, dx.$$

If the target function is known at n points

$$z_1 = \phi(x_1), z_2 = \phi(x_2), \ldots, z_n = \phi(x_n),$$

then the cost function is the square of the Euclidean distance in $\mathbb{R}^n$ between $f_w(\mathbf{x})$ and $\mathbf{z}$

$$C(w) = \sum_{i=1}^{n} (f_w(x_i) - z_i)^2 = \|f_w(\mathbf{x}) - \mathbf{z}\|^2,$$

where $\mathbf{x}^T = (x_1, \ldots, x_n)$ and $\mathbf{z}^T = (z_1, \ldots, z_n)$.

For $\mathbf{x}, \mathbf{z} \in \mathbb{R}^n$ fixed, $C(w)$ is a smooth function of w which can be minimized by the gradient descent method.

Actually, the mapping $w \to f_w(\mathbf{x})$ is a hypersurface in $\mathbb{R}^n$ and $C(w)$ is the distance between $\mathbf{z}$ and a point on this surface. Its minimum is obtained as the length of the orthogonal projection of $\mathbf{z}$ onto the surface.

$$w^* = \arg\min_w C(w) = \arg\min_w \| f_w(\mathbf{x}) - \mathbf{z} \|.$$

The fact that the vector $\mathbf{z} - f_{w^*}(\mathbf{x})$ is perpendicular to the hypersurface is equivalent to the fact that the vector is perpendicular to the tangent plane, which is generated by $\partial_{w_k} f_{w^*}(\mathbf{x})$. Writing the vanishing inner products, we obtain the *normal equation*

$$\sum_{i=1}^{n} (z_i - f_{w^*}(x_i)) \partial_{w_k} f_{w^*}(x_i) = 0.$$

In the case of an Adeline neuron the mapping

$$w \to f_{w^*}(\mathbf{x})$$

is a hyperplane, and the argument w^* exists and it is unique.

Furthermore, the normal equations can be solved explicitly in this case. For other neurons this is not possible and hence the gradient descent method need to be used.

Mean square error function Consider a neural network whose input is a random variable X, and its output is the random variable $Y = f_{w,b}(X)$, where $f_{w,b}$ denotes the input-output mapping of the network, which depends on the parameter w and bias b. Assume the network is used to approximate the target random variable Z. The error function in this case measures a proximity between the output and target random variables Y and Z.

A good candidate is given by the expectation of their squared difference

$$C(w,b) = \mathbb{E}[(Y - Z)^2] = \mathbb{E}[(f_{w,b}(X) - Z)^2]. \quad (0.1)$$

We search for the pair (w, b) which achieves the minimum of the cost function, i.e. we look for

$$(w^*, b^*) = \arg\min C(w,b).$$

This is supposed to be obtained by one of the minimization algorithms (such as the steepest descent method).

Reasons of popularity

1. All square integrable random variables on a probability space forms a Hilbert space with the inner product $\langle X, Y \rangle = \mathbb{E}[XY]$.

Norm $\|X\|^2 = \mathbb{E}[X^2]$

Distance $d(X, Y) = \|X - Y\|$.

Cost function (0.1) represents the square of a distance, $C(w, b) = d(Y, Z)^2$.

Minimizing the cost is equivalent to finding the parameters (w, b) that minimize the distance between the output Y and target Z.

2. The relation with the conditional expectation.

The neural network transforms the input random variable X into an output random variable $Y = f_{w,b}(X)$, which is parameterized by w and b. The information generated by the random variable $f_{w,b}(X)$ is denoted by $\mathcal{E}_{w,b} = \mathfrak{S}(f_{w,b}(X))$, and is the smallest body of information that determines $f_{w,b}(X)$.

All these sigma-fields generate the exit information, $\mathcal{E} = \bigvee_{w,b} \mathcal{E}_{w,b}$, which is the sigma-field generated by the union $\bigcup_{w,b} \mathcal{E}_{w,b}$.

In general, the target random variable, Z, is not determined by the information $\mathcal{E}$. The problem now can be stated as in the following:

Given the exit information $\mathcal{E}$, find the best prediction of Z based on the information $\mathcal{E}$.

This is a random variable, denoted by $Y = \mathbb{E}[Z|\mathcal{E}]$, called *the conditional expectation* of Z given $\mathcal{E}$.

The best predictor, Y, is determined by $\mathcal{E}$ and is situated at the smallest possible distance from Z

$$d(Y, Z) \leq d(U, Z)$$

for any $\mathcal{E}$-measurable random variable U. This is equivalent to

$$\mathbb{E}[(Y - Z)^2] \leq \mathbb{E}[(U - Z)^2], \tag{0.2}$$

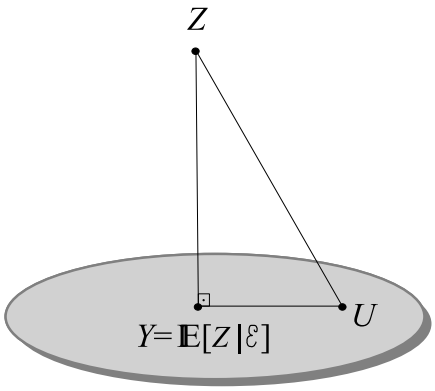
which means that

$$Y = \arg\min_{U} \mathbb{E}[(U - Z)^2].$$

If $U = f_{w,b}(X)$, then the right side of (0.2) is the cost function $C(w, b)$. Then $Y = f_{w^*,b^*}(X)$ minimizes the cost function, and hence

$$(w^*, b^*) = \arg\min_{w,b} \mathbb{E}[(f_{w,b}(X) - Z)^2] = \arg\min_{w,b} C(w, b).$$

Y is the orthogonal projection of Z onto the space of $\mathcal{E}$-measurable functions. The orthogonality is considered in the sense of the Hilbert space of square integrable functions.



3. Extendability to the case when the random variables are known by measurements. Consider n measurements of random variables (X, Z) given by

$$(x_1, z_1), (x_2, z_2), \ldots, (x_n, z_n)$$

Then the cost function is defined by the following average

$$\tilde{C}(w, b) = \frac{1}{n} \sum_{j=1}^{n} \big(f_{w,b}(x_j) - z_j\big)^2.$$

This can be considered as an empirical mean of the square difference of Y and Z.

The case when the input variable X and the target variable Z are independent.

The neuronal network is trained on pairs of independent variables.

What is the best estimator, Y, in this case?

Since X and Z are independent, then also $f_\theta(X)$ and Z will be independent for all values of parameter θ. Then the exit information, $\mathcal{E}$, which is generated by the random variable $f_\theta(X)$ will be independent of Z. The best estimator is

$$Y = \mathbb{E}[Z|\mathcal{E}] = \mathbb{E}[Z],$$

where we used that an independent condition drops out from the conditional expectation. The best estimator is a number, which is the mean of the target variable.

The impostor witch

An experienced witch pretends to read the future of a gullible customer using the coffee traces on her mug. Then the coffee traces represent the input X and the customer future is the random variable Z. The witch-predicted future is the best estimator, Y, which is the mean of all possible futures, and does not depend on the coffee traces.

Proposition: Any differentiable function $f : (0, \infty) \to \mathbb{R}$ satisfying $f(xy) = f(x) + f(y), \quad \forall x, y \in (0, \infty)$ is of the form $f(x) = c \ln x$, with c real constant.

Proof: Let $y = 1 + \epsilon$, with $\epsilon > 0$ small. Then

$$f(x + x\epsilon) = f(x) + f(1 + \epsilon). \tag{0.3}$$

Using the continuity of f, taking the limit

$$\lim_{\epsilon \to 0} f(x + x\epsilon) = f(x) + \lim_{\epsilon \to 0} f(1 + \epsilon)$$

yields $f(1) = 0$. This implies

$$\lim_{\epsilon \to 0} \frac{f(1 + \epsilon)}{\epsilon} = \lim_{\epsilon \to 0} \frac{f(1 + \epsilon) - f(1)}{\epsilon} = f'(1).$$

Equation (0.3) can be written equivalently as

$$\frac{f(x+x\epsilon)-f(x)}{x\epsilon}=\frac{f(1+\epsilon)}{x\epsilon}.$$

Taking $\epsilon \to 0$ and get a differential equation

$$f'(x)=\frac{1}{x}f'(1).$$

Let $c = f'(1)$. Integrating in $f'(x) = \frac{c}{x}$ yields the solution $f(x) = c\ln x + K$, with K constant. Substituting in the initial functional equation, we obtain $K = 0$. ■

Cross entropy-the one-dimensional case.

Let p and q be two densities on $\mathbb{R}$. The negative likelihood function, $-\ell_q(x) = -\ln q(x)$, measures the information given by $q(x)$.

This is compatible with the following properties:

(i) non-negativity: $-\ell_q(x) \geq 0$;

(ii) we have $-\ell_{q_1 q_2}(x) = -\ell_{q_1}(x) - \ell_{q_2}(x)$ for any two independent distributions q_1 and q_2;

(iii) the information increases for rare events, with $\lim_{q(x)\to 0}(-\ell_q(x)) = \infty$.

The *cross entropy* of p with respect to q is defined by the expectation with respect to p of the negative likelihood function as

$$S(p,q) = \mathbb{E}^p[-\ell_q] = -\int_{\mathbb{R}} p(x) \ln q(x)\, dx.$$

This represents the information given by $q(x)$ assessed from the point of view of distribution $p(x)$, which is supposed to be given.

Obviously, $S(p,q) \geq 0$.

Proposition: We have $S(p, q) \geq H(p)$, where $H(p)$ is the Shannon entropy

$$H(p) = -\mathbb{E}^p[\ell_p] = -\int_{\mathbb{R}} p(x) \ln p(x)\, dx.$$

Proof: Evaluate the difference, using the inequality $\ln u \leq u - 1$ for $u > 0$:

$$\begin{aligned} S(p,q) - H(p) &= -\int_{\mathbb{R}} p(x) \ln q(x)\, dx + \int_{\mathbb{R}} p(x) \ln p(x)\, dx \\ &= -\int_{\mathbb{R}} p(x) \ln \frac{q(x)}{p(x)}\, dx \geq \int_{\mathbb{R}} p(x) \Big(\frac{q(x)}{p(x)} - 1 \Big)\, dx \\ &= \int q(x)\, dx - \int p(x)\, dx = 1 - 1 = 0. \end{aligned}$$

Hence, $S(p,q) - H(p) \geq 0$, with $S(p,q) = H(p)$ if and only if $p = q$. ■

Conclusion: For a given density p, the minimum of $q \to S(p,q)$ occurs for $q = p$; this minimum is equal to the Shannon entropy of the given density p.

Remark: in the case of a continuous density the entropy $H(p)$ can be zero or negative, while in the case of a discrete distribution it is always positive.

The difference between the cross entropy and the Shannon entropy is the *Kullback-Leibler divergence*

$$D_{KL}(p||q) = S(p,q) - H(p).$$

Equivalently:

$$D_{KL}(p||q) = -\int_{\mathbb{R}} p(x) \ln \frac{q(x)}{p(x)}\, dx.$$

By the previous result, $D_{KL}(p||q) \geq 0$. However, this is not a distance, since it is neither symmetric, nor it satisfies the triangle inequality.

Both the cross entropy and the Kulback-Leibler divergence can be considered as cost functions for a neuronal network (especially when we have a classification problem).

This topic will be discussed in the following slides.

Setting up notations:
X is the input random variable for a given NN
$Y = f_\theta(X, \xi)$ is the output, where $\theta = (w, b)$
ξ is the noise in the network.
Z is the target variable.
$p_\theta(y|x)$ is the *conditional model density function* (conditional density of Y, given the input X)
$p_{X,Z}(x, z)$ is the *training distribution* (the joint density of (X, Z))

Goal: Match the output Y to target Z using probability density functions.

The value

$$\theta^* = \arg\min_{\theta} S(p_{X,Z}, p_\theta(Z|X))$$

is the minimum for the cost function

$$C(\theta) = S(p_{X,Z}, p_\theta(Z|X)).$$

In the best case scenario, the aforementioned minimum equals the Shannon entropy of the training distribution, $H(p_{X,Z})$.

Let $p_{_X}(x)$ be the density of the input variable X.

$$\begin{aligned}
C(\theta) &= S(p_{_{X,Z}}, p_\theta(Z|X)) = -\iint p_{_{X,Z}}(x,z)\ln p_\theta(z|x)\,dxdz \\
&= -\iint p_{_{X,Z}}(x,z)\ln\Big(\frac{p_\theta(x,z)}{p_{_X}(x)}\Big)\,dxdz \\
&= -\iint p_{_{X,Z}}(x,z)\ln p_\theta(x,z)\,dxdz + \iint p_{_{X,Z}}(x,z)\ln p_{_X}(x)\,dxdz \\
&= S(p_{_{X,Z}}, p_\theta(X,Z)) + \int\Big(\int p_{_{X,Z}}(x,z)\,dz\Big)\ln p_{_X}(x)\,dx \\
&= S(p_{_{X,Z}}, p_\theta(X,Z)) + \int p_{_X}(x)\ln p_{_X}(x)\,dx \\
&= S(p_{_{X,Z}}, p_\theta(X,Z)) - H(p_{_X}),
\end{aligned}$$

where $H(p_{_X})$ is the *input entropy*, i.e. the Shannon entropy of the input variable X.

Since $H(p_X)$ is independent of the model parameter θ, the new cost function

$$\bar{C}(\theta) = S(p_{X,Z}, p_\theta(X, Z)),$$

which is the cross entropy of the training density with the model density, reaches its minimum for the same parameter θ as $C(\theta)$

$$\theta^* = \arg\min_\theta C(\theta) = \arg\min_\theta \bar{C}(\theta).$$

Conclusion: Given a training density, $p_{X,Z}$, and either a model density, $p_\theta(X, Y)$, or a conditional model density, $p_\theta(Y|X)$, we search for the parameter value θ for which either the cost $\bar{C}(\theta)$ or $C(\theta)$, respectively, is minimum.

In practical applications, the random variables (X, Z) are known through n measurements

$$(x_1, z_1), (x_2, z_2), \cdots, (x_n, z_n).$$

Assume that the joint density of the pair (X, Z) is given by its empirical training distribution $\widehat{p}_{X,Z}(x, z)$. The new cost function is the cross entropy between the empirical training distribution defined by the training set and probability distribution defined by the model.

Approximate the expectation with an average

$$\begin{aligned}\tilde{C}(\theta) &= S(\hat{p}_{X,Z}, p_\theta(Z|X)) = \mathbb{E}^{\hat{p}_{X,Z}}[-\ln p_\theta(Z|X)] \\ &= -\frac{1}{n}\sum_{j=1}^{n} \ln p_\theta(z_j|x_j).\end{aligned}$$

Similarly

$$\begin{aligned}\widehat{C}(\theta) &= S(\hat{p}_{X,Z}, p_\theta(X,Z)) = \mathbb{E}^{\hat{p}_{X,Z}}[-\ln p_\theta(X,Z)] \\ &= -\frac{1}{n}\sum_{j=1}^{n} \ln p_\theta(x_j, z_j).\end{aligned}$$

Kulback-Leibler divergence The cost function

$$C(\theta) = D_{KL}(p_{X,Z}||p_\theta(X,Z))$$

is given by the Kulback-Leibler divergence of the training density with the model density. Since Shannon entropy, $H(p_{X,Z})$, is independent of parameter θ, we have

$$\theta^* = \arg\min_\theta D_{KL}(p_{X,Z}||p_\theta(X,Z)) = \arg\min_\theta S(p_{X,Z}, p_\theta(X,Z)).$$

In the best case scenario, when the training and the model distributions coincide, the previous minimum is equal to zero.

In the case when a training set is provided

$$(x_1, z_1), (x_2, z_2), \cdots, (x_n, z_n),$$

the cost function is written using the empirical density, $\hat{p}_{X,Z}$, as

$$\begin{aligned} C(\theta) &= \mathbb{E}^{\hat{p}_{X,Z}}\Big[-\ln \frac{p_\theta(X, Z)}{\hat{p}_{X,Z}}\Big] \\ &= -\frac{1}{n}\sum_{j=1}^{n}\Big(\ln p_\theta(x_i, z_i) - \ln \hat{p}(x_i, z_i)\Big). \end{aligned}$$

Maximum Likelihood The minimum of the aforementioned empirical cost function

$$\widehat{C}(\theta) = \mathbb{E}^{\hat{p}_{X,Z}}[-\ln p_\theta(X, Z)] = -\frac{1}{n}\sum_{j=1}^{n} \ln p_\theta(x_j, z_j),$$

which is given by $\theta^* = \arg\min_\theta \widehat{C}(\theta)$, has the following distinguished statistical property: it is the *maximum likelihood estimator* of θ, given n independent measurements

$$(x_1, z_1), (x_2, z_2), \cdots, (x_n, z_n).$$

This follows from the next computation, which uses properties of logarithms

$$\begin{aligned}
\theta^* &= \arg\min_\theta \widehat{C}(\theta) = \arg\max_\theta \frac{1}{n}\sum_{j=1}^{n} \ln p_\theta(x_j, z_j) \\
&= \arg\max_\theta \sum_{j=1}^{n} \ln p_\theta(x_j, z_j) = \arg\max_\theta \ln\Big(\prod_{j=1}^{n} p_\theta(x_j, z_j)\Big) \\
&= \arg\max_\theta \prod_{j=1}^{n} p_\theta(x_j, z_j) = \arg\max_\theta p_\theta(X = \mathbf{x}, Z = \mathbf{z}) \\
&= \theta_{ML}.
\end{aligned}$$

Other several ways to form cost functions are given below.

L^1-distance Assuming the densities are integrable, the distance between them is measured by $D_1(p_\theta, p_{X,Z}) = \iint |p_{X,Z}(x,z) - p_\theta(x,z)|\, dxdz$. The minimum of D_1 is zero and it is attained for identical distributions.

L^2-distance Assuming the densities are square integrable, the distance between them is measured by $D_2(p_\theta, p_{X,Z}) = \iint \left(p_{X,Z}(x,z) - p_\theta(x,z)\right)^2 dxdz$. For identical distributions this distance vanishes.

Hellinger distance Another variant to measure the distance is

$$H^2(p_\theta, p_{X,Z}) = 2 \iint \left[\sqrt{p_\theta(x,z)} - \sqrt{p_{X,Z}(x,z)}\right]^2 dxdz.$$

Jeffrey distance This is

$$J(p_\theta, p_{X,Z}) = \frac{1}{2} \iint \big(p_\theta(x,z) - p_{X,Z}(x,z)\big)\big(\ln p_\theta(x,z) - \ln p_{X,Z}(x,z)\big)\, dxdz$$