MO434 - Deep Learning
Fundamentals of (Deep) Neural Networks I

Alexandre Xavier Falcão

Institute of Computing - UNICAMP

afalcao@ic.unicamp.br

Agenda

- Data types, preprocessing, and data representation.
- A simple neural network for classification and its geometrical interpretation.
- A simple example of classification.
- A simple example of regression.

Data

Data consist of all digital information acquired by sensors and/or human input to solve a computer (ML) problem.

Data

Data consist of all digital information acquired by sensors and/or human input to solve a computer (ML) problem.

- Data may appear as numbers and texts in a table, a set of images and their annotations, a collection of texts in a given language (corpus), time series of stock prices, brain activity signals (EEGs), speech signals, etc.

Data

Data consist of all digital information acquired by sensors and/or human input to solve a computer (ML) problem.

- Data may appear as numbers and texts in a table, a set of images and their annotations, a collection of texts in a given language (corpus), time series of stock prices, brain activity signals (EEGs), speech signals, etc.

- Data may be classified as either structured or unstructured. ▸ EXAMPLES

Data

Data consist of all digital information acquired by sensors and/or human input to solve a computer (ML) problem.

- Data may appear as numbers and texts in a table, a set of images and their annotations, a collection of texts in a given language (corpus), time series of stock prices, brain activity signals (EEGs), speech signals, etc.
- Data may be classified as either structured or unstructured. ▸ EXAMPLES
- The main steps for data analysis are preprocessing, data representation, and (supervised, unsupervised, semi-supervised) classification/regression.

Preprocessing

Preprocessing is an application-dependent task. For text analysis, it is usually required to

- remove stopping words, e.g., for, the, but, after, yet, so;
- separate the remaining words (tokenization);
- abreviate words and/or transform them into a standard form (stemmization and lemmatization);
- segment the text into parts of speech (tagging), e.g., nouns, adjectives and verbs;
- recognize entity names (NER), e.g., institution, person and country; and
- convert the words into numbers, such that similar words are mapped to close numbers.

Preprocessing

For image analysis, the images might need to be cropped at the center, rescaled and normalized. Common normalization functions applied to each pixel p of each image channel c are:

$$\begin{aligned} c'(p) &= \frac{c(p) - c_{\min}}{c_{\max} - c_{\min}}, \\ c'(p) &= \frac{c(p) - \mu_c}{\sigma_c}, \end{aligned}$$

where $c_{\min}$, $c_{\max}$, μ_c, and σ_c are the minimum, maximum, mean, and standard deviation of pixel values in that channel for images of a given application.

Preprocessing

For image analysis, the images might need to be cropped at the center, rescaled and normalized. Common normalization functions applied to each pixel p of each image channel c are:

$$\begin{aligned} c'(p) &= \frac{c(p) - c_{\min}}{c_{\max} - c_{\min}}, \\ c'(p) &= \frac{c(p) - \mu_c}{\sigma_c}, \end{aligned}$$

where $c_{\min}$, $c_{\max}$, μ_c, and σ_c are the minimum, maximum, mean, and standard deviation of pixel values in that channel for images of a given application.

Ex: Pretrained models on ImageNet transform rgb images with 8 bits per channel by $r(p) \leftarrow \frac{r(p)}{255}$, $g(p) \leftarrow \frac{g(p)}{255}$, $b(p) \leftarrow \frac{b(p)}{255}$, and then $r(p) \leftarrow \frac{r(p) - 0.485}{0.229}$, $g(p) \leftarrow \frac{g(p) - 0.456}{0.224}$, and $b(p) \leftarrow \frac{b(p) - 0.406}{0.225}$.

Data representation

Once the n observations (numbers or texts) for each sample in a dataset are defined and converted into numbers (features),

Data representation

Once the n observations (numbers or texts) for each sample in a dataset are defined and converted into numbers (features),

- the dataset becomes a set $X = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$ with feature vectors $\mathbf{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}\right)$ associated to the N samples.

Data representation

Once the n observations (numbers or texts) for each sample in a dataset are defined and converted into numbers (features),

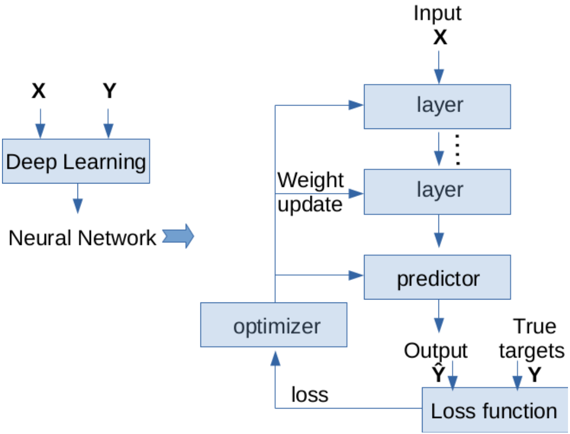
- the dataset becomes a set $X = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$ with feature vectors $\mathbf{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}\right)$ associated to the N samples.
- For classification and regression, the prediction $\hat{\mathbf{y}}^{(i)} = F(\mathbf{x}^{(i)})$ must be as close as possible to its target $\mathbf{y}^{(i)} \in Y = \{\mathbf{y}^{(i)}\}_{i=1}^{N}$.

Data representation

Once the n observations (numbers or texts) for each sample in a dataset are defined and converted into numbers (features),

- the dataset becomes a set $X = \{\mathbf{x}^{(i)}\}_{i=1}^{N}$ with feature vectors $\mathbf{x}^{(i)} = \left(x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}\right)$ associated to the N samples.
- For classification and regression, the prediction $\hat{\mathbf{y}}^{(i)} = F(\mathbf{x}^{(i)})$ must be as close as possible to its target $\mathbf{y}^{(i)} \in Y = \{\mathbf{y}^{(i)}\}_{i=1}^{N}$.
- A neural network F must then be trained to minimize a suitable loss function.

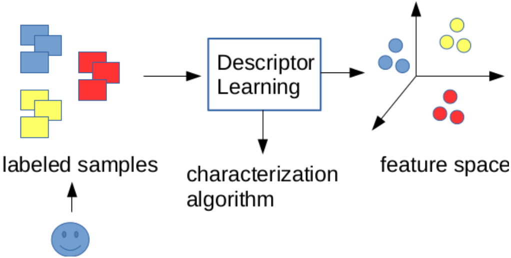
Training a neural network F



This process must use part of the dataset (X, Y) called training set, leaving the rest for validation and test.

Deep neural network for classification

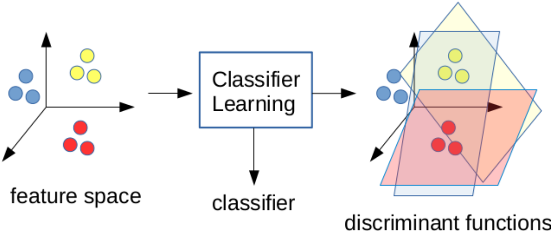
A deep neural network is a sequence of dense/convolutional layers (descriptor/encoder) which transforms data representation to a more suitable one for classification/regression (classifier/decoder).



For classification, at the last hidden layer, it is expected that samples from distinct classes be linearly separable.

Deep neural network for classification

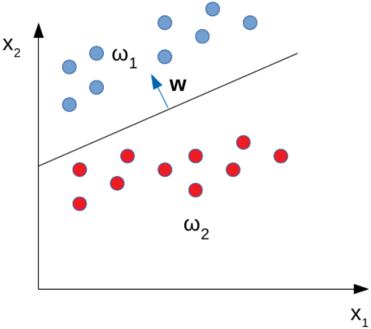
A deep neural network is a sequence of dense/convolutional layers (descriptor/encoder) which transforms data representation to a more suitable one for classification/regression (classifier/decoder).



For classification, at the last hidden layer, it is expected that samples from distinct classes be linearly separable.

A simple example of data transformation for classification

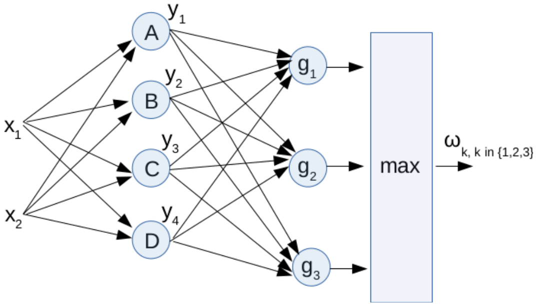
For a simple example with 2D feature vectors $\mathbf{x} = (x_1, x_2)$ and two classes $\mathbf{y} = (y_1, y_2)$, with $y_k = f(\langle \mathbf{x}, \mathbf{w} \rangle + w_0)$, $f(v) = 0$, for $v < 0$, or $f(v) = v$, $v \geq 0$, and the decision is $\omega_k = \arg\max_{k=1,2}\{y_k\}$.



Each vector $\mathbf{w} \in \Re^2$ (neuron weights) and the bias $w_0 \in \Re$ defines a line at some position of the 2D feature space. A single layer with one neuron would be enough, but we use two.

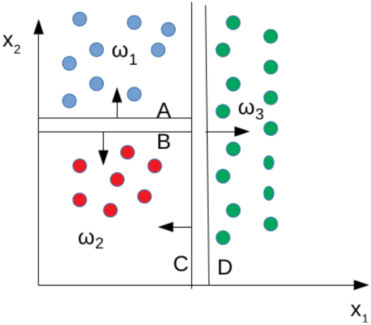
A simple example of data transformation for classification

Whenever the classes are nonlinearly separable, hidden layers may be added with the number of neurons at the decision layer equal to the number of classes.



A simple example of data transformation for classification

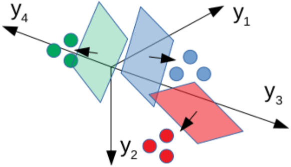
A hidden layer with four neurons and a decision layer with three neurons are enough for the simple example below.



For each class, the decision layer will identify which neurons activate together when the example comes from that class.

A simple example of data transformation for classification

The output of the hidden layer will be a four-dimensional feature space.



The decision layer will use one hyperplane per class to separate them in that four-dimensional feature space. ▸ CLASSIFICATION

This link helps to understand how a neural network separates classes in 2-dimensional feature spaces: `https://playground.tensorflow.org/`

A simple example of regression

For regression, a simple example is the estimation of $y = x_1^2 + x_2^2$ from training examples $(\mathbf{x}^{(i)}, y^{(i)})$, $i = 1, 2, \ldots, N$. ▸ REGRESSION

A simple example of regression

For regression, a simple example is the estimation of $y = x_1^2 + x_2^2$ from training examples $(\mathbf{x}^{(i)}, y^{(i)})$, $i = 1, 2, \ldots, N$. ▸ REGRESSION

- The basic difference between these NNs for classification and regression rely on their loss functions and activation functions at the decision layer.

A simple example of regression

For regression, a simple example is the estimation of $y = x_1^2 + x_2^2$ from training examples $(\mathbf{x}^{(i)}, y^{(i)})$, $i = 1, 2, \ldots, N$. ▸ REGRESSION

- The basic difference between these NNs for classification and regression rely on their loss functions and activation functions at the decision layer.
- Regression may estimate one or more properties y_k from one or multiple input features x_j.

A simple example of regression

For regression, a simple example is the estimation of $y = x_1^2 + x_2^2$ from training examples $(\mathbf{x}^{(i)}, y^{(i)})$, $i = 1, 2, \ldots, N$. ▸ REGRESSION

- The basic difference between these NNs for classification and regression rely on their loss functions and activation functions at the decision layer.
- Regression may estimate one or more properties y_k from one or multiple input features x_j.
- Activation, loss, and training by backpropagation for classification and regression will be discussed in the next lecture.