Deep Learning

AlejandroMllo

This document serves as a very brief summary of the topics covered in each chapter of the book Deep Learning [1].

Disclaimer: This document is completely extracted from [1], the author does not attribute any ownership over the material.

1 Introduction

- Deep Learning (DL) learns complicated concepts by building them out of simpler ones. DL is the study of models composed of either learned functions or learned concepts.
- Data Representation is key when finding patterns.
- Many AI tasks can be solved by designing the right set of features to extract.
- **Representation Learning:** discovers the mapping from representation to output and the representation itself.
  - Makes use of an *Autoencoder* which is a combination of:
    - Encoder: Converts the input data into a different representation.
    - Decoder: Converts the new representation back into the original format.
- Factors: sources of influence in the model.
- **Depth**, in a DL model, enables the computer to learn a multistep computer program (not all the information in a layer's activations necessarily encodes factors that explain the input). There are two ways to measure it:
  - Based on the number of sequential instructions, or
  - Based on the depth of the graph describing how concepts are related to each other (used by deep probabilistic models).
- DL is not an attempt to simulate the brain. It borrows ideas from different fields (one of which is neuroscience).

2 Linear Algebra

- Scalar: a single number. Denoted by lowercase variable names and italics.
- Vectors: ordered array of numbers. Denoted by lowercase and bold typeface. $\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$
- Matrices: ordered 2-D $m \times n$ array of numbers. Denoted by uppercase and bold typeface.

$$\mathbf{A} = \begin{bmatrix} A_{1,1} & \dots & A_{1,n} \\ \vdots & \dots & \vdots \\ A_{1,m} & \dots & A_{m,n} \end{bmatrix}$$

- Tensors: array of numbers arranged on a regular grid with a variable number of axes.
- Transpose: mirror image of a matrix/vector. $(A^\top)_{i,j} = A_{j,i}$.
- It is possible to add $m \times n$ matrices together. $\mathbf{C} = \mathbf{A} + \mathbf{B}$, where $C_{i,j} = A_{i,j} + B_{i,j}$.
- To add a scalar to a matrix or multiply a matrix by a scalar, perform that operation on each element of the matrix.

- Matrix Multiplication: $\mathbf{C} = \mathbf{AB}$ is defined as:

$$C_{i,j} = \sum_k A_{i,k} B_{k,j}$$

where $\mathbf{A}$ is of shape $m \times n$, $\mathbf{B}$ is of shape $n \times p$ and $\mathbf{C}$ is of shape $m \times p$.
Matrix product has some useful properties not covered in detail in the book.

- System of linear equations: $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{x}$ is a vector of unknown variables to be solved.
- Identity Matrix ($\boldsymbol{I}_n$): $n \times n$ matrix whose entries along the main diagonal are 1, while all the other entries are zero.
- Matrix Inverse ($\mathbf{A}^{-1}$): $\mathbf{A}^{-1}\mathbf{A} = \boldsymbol{I}_n$. For a matrix to have an inverse, it must be $n \times n$ and all its columns be linearly independent, it means that no column is a linear combination of another column.
- Norm (L^p): function that measures the size of vectors.

$$\|\mathbf{x}\|_p = \left(\sum_i |x_i|^p \right)^{\frac{1}{p}}$$

where $p \in \mathbb{R}$, $p \geq 1$.

  - The squared L^2 norm is commonly used and can be calculated as $\mathbf{x}^\top \mathbf{x}$.
  - L^∞ norm (max norm): absolute value of the element with the largest magnitude in the vector.

- Frobenius Norm: used to measure the size of a matrix (analogous to the L^2 norm of a vector).

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} A_{i,j}^2}$$

- Symmetric Matrix: $\mathbf{A} = \mathbf{A}^\top$.
- Unit Vector: $\|\mathbf{x}\|_2 = 1$ (unit norm).
- Vectors $\mathbf{x}$ and $\mathbf{y}$ are orthogonal if $\mathbf{x}^\top \mathbf{y} = 0$. If both this vectors have unit norm, then they are called orthonormal.
- Orthogonal Matrix: square matrix whose rows are mutually orthonormal and whose columns are mutually orthonormal.
- An eigenvector of a square matrix $\mathbf{A}$ is a nonzero vector $\mathbf{v}$ such that multiplication by $\mathbf{A}$ alters only the scale of $\mathbf{v}$: $\mathbf{Av} = \lambda \mathbf{v}$. The scalar λ is known as the eigenvalue corresponding to this eigenvector. The eigendecomposition of $\mathbf{A}$ is given by: $\mathbf{A} = \mathbf{V}\mathrm{diag}(\lambda)\mathbf{V}^{-1}$, where $\mathbf{V}$ is the matrix whose columns are each eigenvector of $\mathbf{A}$ and $\mathrm{diag}(\lambda)$ is the diagonal matrix of eigenvalues such that the eigenvalue at $\lambda_{i,i}$ is the one associated with the eigenvector (column) i of $\mathbf{V}$. Eigendecomposition is not defined for every matrix.
- Singular Value Decomposition (SVD): $\mathbf{A} = \mathbf{UD}\boldsymbol{V}^\top$, where $\mathbf{A}$ is a $m \times n$ matrix. $\mathbf{U}$ ($m \times m$) is composed by the eigenvectors of $\boldsymbol{AA}^\top$; $\mathbf{V}$ ($n \times n$) is composed by the eigenvectors of $\boldsymbol{A}^\top \boldsymbol{A}$; and $\mathbf{D}$ ($m \times n$) is a diagonal matrix composed by the eigenvalues of $\boldsymbol{AA}^\top$ or $\boldsymbol{A}^\top \boldsymbol{A}$.
- Moore-Penrose pseudoinverse: The pseudoinverse of $\mathbf{A}$ is defined as a matrix $\boldsymbol{A}^+ = \boldsymbol{V}\boldsymbol{D}^+\boldsymbol{U}^\top$, where $\mathbf{U}$, $\mathbf{D}$ and $\mathbf{V}$ are the SVD of $\mathbf{A}$, and the pseudoinverse $\boldsymbol{D}^+$ of a diagonal matrix $\mathbf{D}$ is obtained by taking the reciprocal of its nonzero elements then making the transpose of the resulting elements.
- The Trace operator gives the sum of the diagonal entries of a matrix: $\mathrm{Tr}(\mathbf{A}) = \sum_i \boldsymbol{A}_{i,i}$. It has some useful identities to manipulate expressions.
- Determinant ($\det(\mathbf{A})$): function that maps matrices to real scalars. It is equal to the product of all the eigenvalues of a matrix.
- Principal Components Analysis (PCA): ML algorithm that can be derived using only knowledge of basic Linear Algebra. The reader can find a description of the implementation at section 2.12.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.