Appendix A

Notation

This appendix details the notation used in this book. This mostly adheres to standard conventions in computer science, but deep learning is applicable to many different areas, so it is explained in full. In addition, there are several notational conventions that are unique to this book, including notation for functions and the systematic distinction between parameters and variables.

Scalars, vectors, matrices, and tensors

Scalars are denoted by either small or capital letters a, A, α. Column vectors (i.e., 1D arrays of numbers) are denoted by small bold letters $\mathbf{a}, \boldsymbol{\phi}$, and row vectors as the transpose of column vectors $\mathbf{a}^T, \boldsymbol{\phi}^T$. Matrices and tensors (i.e., 2D and ND arrays of numbers, respectively) are both represented by bold capital letters $\mathbf{B}, \boldsymbol{\Phi}$.

Variables and parameters

Variables (usually the inputs and outputs of functions or intermediate calculations) are always denoted by Roman letters $a, \mathbf{b}, \mathbf{C}$. Parameters (which are internal to functions or probability distributions) are always denoted by Greek letters $\alpha, \boldsymbol{\beta}, \boldsymbol{\Gamma}$. Generic, unspecified parameters are denoted by $\boldsymbol{\phi}$. This distinction is retained throughout the book except for the policy in reinforcement learning, which is denoted by π according to the usual convention.

Sets

Sets are denoted by curly brackets, so $\{0, 1, 2\}$ denotes the numbers 0, 1, and 2. The notation $\{0, 1, 2, \ldots\}$ denotes the set of non-negative integers. Sometimes, we want to specify a set of variables and $\{\mathbf{x}_i\}_{i=1}^{I}$ denotes the I variables $\mathbf{x}_1, \ldots \mathbf{x}_I$. When it's not necessary to specify how many items are in the set, this is shortened to $\{\mathbf{x}_i\}$. The notation $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{I}$ denotes the set of I pairs $\mathbf{x}_i, \mathbf{y}_i$. The convention for naming sets is to use calligraphic letters. Notably, $\mathcal{B}_t$ is used to denote the set of indices in a batch at iteration t during training. The number of elements in a set $\mathcal{S}$ is denoted by $|\mathcal{S}|$.

The set $\mathbb{R}$ denotes the set of real numbers. The set $\mathbb{R}^+$ denotes the set of non-negative real numbers. The notation $\mathbb{R}^D$ denotes the set of D-dimensional vectors containing real

numbers. The notation $\mathbb{R}^{D_1 \times D_2}$ denotes the set of matrices of dimension $D_1 \times D_2$. The notation $\mathbb{R}^{D_1 \times D_2 \times D_3}$ denotes the set of tensors of size $D_1 \times D_2 \times D_3$ and so on.

The notation $[a, b]$ denotes the real numbers from a to b, including a and b themselves. When the square brackets are replaced by round brackets, this means that the adjacent value is not included in the set. For example, the set $(-\pi, \pi]$ denotes the real numbers from $-\pi$ to π, but excluding $-\pi$.

Membership of sets is denoted by the symbol $\in$, so $x \in \mathbb{R}^+$ means that the variable x is a non-negative real number, and the notation $\boldsymbol{\Sigma} \in \mathbb{R}^{D \times D}$ denotes that $\boldsymbol{\Sigma}$ is a matrix of size $D \times D$. Sometimes, we want to work through each element of a set systematically, and the notation $\forall \ \{1, \ldots, K\}$ means "for all" the integers from 1 to K.

Functions

Functions are expressed as a name, followed by square brackets that contain the arguments of the function. For example, $\log[x]$ returns the logarithm of the variable x. When the function returns a vector, it is written in bold and starts with a small letter. For example, the function $\mathbf{y} = \mathbf{mlp}[\mathbf{x}, \boldsymbol{\phi}]$ returns a vector $\mathbf{y}$ and has vector arguments $\mathbf{x}$ and $\boldsymbol{\phi}$. When a function returns a matrix or tensor, it is written in bold and starts with a capital letter. For example, the function $\mathbf{Y} = \mathbf{Sa}[\mathbf{X}, \boldsymbol{\phi}]$ returns a matrix $\mathbf{Y}$ and has arguments $\mathbf{X}$ and $\boldsymbol{\phi}$. When we want to leave the arguments of a function deliberately ambiguous, we use the bullet symbol (e.g., $\mathbf{mlp}[\bullet, \boldsymbol{\phi}]$).

Minimizing and maximizing

Some special functions are used repeatedly throughout the text:

- The function $\min_x[\mathrm{f}[x]]$ returns the minimum value of the function $\mathrm{f}[x]$ over all possible values of the variable x. This notation is often used without specifying the details of how this minimum might be found.
- The function $\operatorname{argmin}_x[\mathrm{f}[x]]$ returns the value of x that minimizes $\mathrm{f}[x]$, so if $y = \operatorname{argmin}_x[\mathrm{f}[x]]$, then $\min_x[\mathrm{f}[x]] = \mathrm{f}[y]$.
- The functions $\max_x[\mathrm{f}[x]]$ and $\operatorname{argmax}_x[\mathrm{f}[x]]$ perform the equivalent operations for maximizing functions.

Probability distributions

Probability distributions should be written as $Pr(x = a)$, denoting that the random variable x takes the value of a. However, this notation is cumbersome. Hence, we usually simplify this and just write $Pr(x)$, where x denotes either the random variable or the value it takes according to the sense of the equation. The conditional probability of y given x is written as $Pr(y|x)$. The joint probability of y and x is written as $Pr(y, x)$. These two forms can be combined, so $Pr(\mathbf{y}|\mathbf{x}, \boldsymbol{\phi})$ denotes the probability of the variable $\mathbf{y}$, given that we know $\mathbf{x}$ and $\boldsymbol{\phi}$. Similarly, $Pr(\mathbf{y}, \mathbf{x}|\boldsymbol{\phi})$ denotes the probability of variables $\mathbf{y}$ and $\mathbf{x}$ given that we know $\boldsymbol{\phi}$. When we need two probability distributions over the same variable, we write $Pr(x)$ for the first distribution and $q(x)$ for the second. More information about probability distributions can be found in appendix C.

Asymptotic notation

Asymptotic notation is used to compare the amount of work done by different algorithms as the size D of the input increases. This can be done in various ways, but this book only uses *big-O* notation, which represents an upper bound on the growth of computation in an algorithm. A function $\mathrm{f}[n]$ is $\mathcal{O}[\mathrm{g}[n]]$ if there exists a constant $c > 0$ and integer n_0 such that $\mathrm{f}[n] < c \cdot \mathrm{g}[n]$ for all $n > n_0$.

This notation provides a bound on the worst-case running time of an algorithm. For example, when we say that inversion of a $D \times D$ matrix is $\mathcal{O}[D^3]$, we mean that the computation will increase no faster than some constant times D^3 once D is large enough. This gives us an idea of how feasible it is to invert matrices of different sizes. If $D = 10^3$, then it may take of the order of 10^9 operations to invert it.

Miscellaneous

A small dot in a mathematical equation is intended to improve ease of reading and has no real meaning (or just implies multiplication). For example, $\alpha \cdot \mathrm{f}[x]$ is the same as $\alpha \mathrm{f}[x]$ but is easier to read. To avoid ambiguity, dot products are written as $\mathbf{a}^T\mathbf{b}$ (see appendix B.3.4). A left arrow symbol $\leftarrow$ denotes assignment, so $x \leftarrow x + 2$ means that we are adding two to the current value of x.