Part I

Preliminary Considerations

1
Introduction

Recent advances in deep learning have spearheaded impressive automated capabilities in visual and natural language processing and resulted in machine learning becoming a very popular discipline. While a decade or so ago only a handful of individuals – well read in matters of artificial intelligence (AI) and machine learning – perceived the inherent contributions that machine learning could bring to their field of expertise, today many people believe that machine learning will solve all of their problems: for better or for worse, machine learning has become ubiquitous. In such a climate, evaluating machine learning's performance is of paramount importance, especially when large-scale deployments of tools based on its foundations are under consideration.

Performance evaluation in machine learning and, more generally, artificial intelligence has always been an important aspect of the field. Back in 1950, Turing famously designed a test he called the "imitation game," now known as the "Turing test," to evaluate the "intelligence" of the machine. In truth, any computer system must be subjected to testing and evaluation, and the software engineering discipline, born in the mid to late 1980s, created strict guidelines on how to do so. The issue with AI and machine learning, however, is that their testing and evaluation are not easy feats. While in traditional computer systems the range of outcomes is narrow and often quantifiable, in AI and machine learning, it is neither. The goal here is to get the machine to behave "intelligently." Yet, what is intelligence? First, not every human being behaves the same way, and second, humans must function in unpredictable settings. These and many other considerations are what make the evaluation of AI and machine learning systems ill-defined. In fact, with the release of ChatGPT in 2022, the debate is on, once again, as it probably was when ELIZA, one of the first computer programs able to converse in English with people, was released in the 1960s. Though the book does not go into the more philosophical side of the problem, one thing is clear: in order to mimic humans adequately, the computer program needs to be able to *generalize* to unknown situations, and, as a result, it has to include nondeterministic "behavior." It is this lack of determinism, or the system's *inductive* behavior, that makes testing in the context of machine learning so complex, and this is one aspect of the problem into which our book takes a deep dive.

Another aspect relates to the fact that machine learning is now mature enough to be embedded in products designed to hit the market. Its evaluation is thus leaving the realm of philosophical questioning and becoming a practical matter. This shift from an inconse-

quential practice necessary to publish research papers, but not robust enough to truly assess the practical value of the resulting product, is a crucial one. Another essential function of this book is to bridge the gap between these two evaluation paradigms. It presents many different evaluation methods with the goal of covering the various situations that may arise when deploying a product: Is the product competitive? Robust? Safe? Fair? What is its expected benefit–cost ratio? While we cannot envision all the situations that could arise, the book gives the readers many tools that will allow them to determine what evaluation practice will serve their specific situation best. This is particularly transparent in the case of evaluation metrics discussed in Chapters 4, 5, 8, and 9, where different applications (e.g., medical, information retrieval, security) and learning paradigms (e.g., supervised, anomaly detection, time-series analysis, unsupervised) call for different types of measurements with overlaps occasionally occurring.

1.1 Motivation for This Book

The deep learning revolution is what motivated us most to write this book. Until deep learning started bearing fruit a few years ago, the technology was not mature enough to be considered for widespread use in sensitive applications such as self-driving cars, automatic screening of job applicants, and so on. As a result, we reasoned, the evaluation methods previously considered may not be robust enough for the types of applications machine learning is now encountering. This is not to say that machine learning tools were not evaluated and used in practical applications before, but instead, it is to emphasize that their uses are becoming more widespread now, and their domains of application more and more sensitive. This, we feel, requires further scrutiny. Moreover, while machine learning evaluation was a topic of conversation in the research community and within a circle of practitioners tightly linked to that research community, at this point, the need to understand how to evaluate machine learning algorithms adequately has spread to a much larger audience. It encompasses many scientific circles beyond the traditional machine learning community (e.g., chemists, biologists, physicists, environmental scientists, medical researchers), the business community with its various sectors, and even social studies and the arts.

The purpose of this book is to present a concise, yet complete, intuitive, yet formal, presentation of machine learning evaluation. The book has a predecessor, cowritten by one of the current coauthors, but although the new book builds upon the old one, it departs radically from it by increasing its coverage, updating its suggested methodologies, and, generally speaking, proposing a more robust approach. Not insignificantly, the new book also expands the reach of the discussion to the broader community.

An additional motivation for this book, which also ties together the reasons previously mentioned, is that although the situation has improved, evaluation of machine learning algorithms is often seen as an annoying and non-rewarding task. After all, creating new algorithms is a lot more exciting than testing them! More often than not, researchers or practitioners feel that they *need* to perform the task of evaluation to satisfy crusty conference or journal reviewers or bosses, but that the task is well below their skill levels. The view in this book, actually, is that, first, the evaluation of machine learning algorithms is

a fascinating field of study in and of itself, and, second, the result of this evaluation has become of extreme practical importance now that the world embraces machine learning and embeds it in its products.

Indeed, the field has, at last, matured to the point where the safety of the technology must be considered in ways similar to the way in which the safety of other technological or medical advances have had to be assessed for many years. While such evaluation might have been seen as overkill, and perhaps rightly so, in the earlier years of machine learning, now that self-driving cars use the technology as well as medical diagnostic or hiring systems, the issue cannot be ignored any longer, and approaches similar to phased medical trials or other industrial-strength evaluation need to be put in place. In fact, the time might have come for teams of evaluators, independent from the developers, to test AI products. Perhaps, even, the equivalent of the CDC or FDA for AI needs to be put in place to approve or prevent AI-based products from reaching the market in order to protect the consumer from physical or psychological harm and ensure the fairness of the product.[1] The purpose of this book is to lay out the various tools available to conduct a rigorous performance evaluation for a variety of machine learning paradigms including those used in supervised learning, unsupervised learning, image processing applications, large language models, and so on. The tools presented should also help evaluate different aspects of practical uses that may differ from one area of application to the next. To make the task of evaluation easier, the book refers the reader to existing evaluation tools, or provides new ones, where they are lacking, on what is currently the most prevalent machine learning development platform, scikit-learn.

There are a variety of reasons why we believe that the time has come to write this book. As mentioned previously, in the 13 years since the publication of its predecessor, there has been an explosion of people using machine learning tools, and trying to make sense of how to evaluate their performance. This new audience is not as homogenous as it was 13 years ago since, in addition to computer scientists, it includes many statisticians, data scientists, and practitioners of various disciplines ranging from the core sciences to the social sciences, including the medical sciences, education, journalism, and even arts disciplines such as literature, visual arts, and music.

In addition to the explosion and diversity of new users, there has also been a data explosion, bringing into focus different types of data (e.g., images, text), much larger amounts (sometimes in the order of terabytes), and a true desire to leverage the data toward robust industrial products. This last change makes the issue of evaluation essential and raises questions that were not considered with the same urgency in the past, such as privacy, bias, and explainability matters, as well as domain and task-dependent considerations. These, along with the model's correctness, need to be properly assessed before a product can be deployed or commercialized. By the same token, new machine learning tasks have emerged or, at least, become more prevalent. This includes computer vision tasks such as image

[1] A practical advantage of this suggestion, by the way, would be to free developers from having to perform as rigorous an evaluation of their ideas since these could be pitted professionally at a later stage. More to the point, however, such a division of labor would lead to a less biased evaluation process since the evaluators would have no stake in the products they are testing.

segmentation, unsupervised learning, and data stream analysis. Each of these new tasks requires new analysis tools.

In summary, the different types of data and their amounts, the new tasks that have emerged, as well as the stricter evaluation imperatives that are – or should be – in effect, may, in certain cases, require the use of evaluation tools different from those discussed previously, and ready access to these tools is imperative.

1.2 Contents and Organization of the Book

Machine learning evaluation typically refers to the evaluation of classification. The scope of this book extends beyond the task of classification, although classification remains a predominant aspect of machine learning and is covered thoroughly.

The presentation of the book is designed to appeal to both machine learning researchers and practitioners. Specifically, we provide both an informal discussion and access to simple-to-use tools with clear guidelines on when to use them (when possible), as well as more formal, theoretical explanations to support some of these practical considerations. To reflect the current trend in the field, all of the code provided is written in Python and uses the scikit-learn package (though some of the tools exist, in a previous version, in R).

The book is organized into four parts. Part I reviews essential statistical and machine learning concepts that are needed to provide context for the remainder of the book. This includes random variables, distributions, confidence intervals, and hypothesis testing on the statistical side; as well as the concepts of loss function, risk, empirical and structural risk minimization, regularization, the bias–variance tradeoff, clustering, dimensionality reduction, latent variable modeling, and generative learning on the machine learning side. In addition, Part I presents the de facto way in which machine learning evaluation is conducted, reviewing, along the way, well-known concepts such as the confusion matrix, micro- and macro- averaging, as well as well-known classification, regression, and clustering metrics; error estimation methods such as the holdout and k-fold cross-validation, and basic statistical tests such as the t-test and the sign test. Many of the evaluation methods covered in this part are well known to most practitioners of machine learning. In addition to reviewing them, we explain why they are, oftentimes, not sufficient. This prompts a discussion of why it is necessary to go beyond the material covered in the first part of the book, thus motivating the need for its further three parts.

Part II discusses machine learning evaluation in the important classification setting. In particular, it discusses the metrics that have been proposed for that setting, paying particular attention to the issues of class imbalances, costs, uncertainty, and calibration; the error-estimation/resampling approaches that have been proposed and their relationship to the bias and variance of the error estimates; and the different approaches to statistical analysis, including null hypothesis statistical testing, confidence intervals, effect size, and power analysis, as well as newer Bayesian analysis approaches that have recently been proposed in the machine learning literature.

Part III then turns to machine learning tasks other than classification. Evaluation methods for many tasks are presented, including those for classical paradigms such as regression anal-

ysis, time-series analysis, outlier detection, and reinforcement learning, and also newer tasks such as positive-unlabeled classification, ordinal classification, multi-labeled classification, image segmentation, text generation, data stream mining, and lifelong learning. A full chapter is then devoted to the important unsupervised learning paradigm, and evaluation methods for tasks including clustering and hierarchical clustering, dimensionality reduction, latent variable models, and generative models are discussed.

Finally, Part IV turns to practical considerations related to evaluation and deployment. First, machine learning evaluation is presented in a software engineering light whose goal is to herald the future of machine learning's use in industrial applications. Topics include data, algorithms, and platform imperfections, online testing, along with a description of current industry practice, and suggestions for improvements. The next chapter turns to the question of how to practice machine learning in a responsible manner. In particular, it dives into the issues of data and algorithmic bias, fairness, explainability, privacy, and security among others, and advocates the need for human-centered machine learning.

The book concludes with a discussion of how the performance evaluation components discussed throughout the book unify into an overall framework for in-laboratory evaluation. This is followed by a discussion of how to move from a laboratory setting to a deployment setting based on the material covered in Part IV of the book. Associated with this deployment, we emphasize the potential social consequences of machine learning technology, together with their causes, and suggest that these potential social effects should be considered as part of the evaluation framework.

The book comes accompanied by the Github site https://github.com/zoisboukouvalas/MachineLearningEvaluation_TowardsReliableResponsibleAI, which was written by American University graduate students Liam Spoletini and Sunday Okechukwu. The site provides Python code that illustrates how to implement the different evaluation methods discussed in the book. This, therefore, provides a quick way for machine learning designers and practitioners to apply evaluation techniques to their applications.

2

Statistics Overview

This chapter aims to introduce the basic elements of statistics necessary to understand the more advanced concepts and procedures that will be introduced in later chapters. Needless to say, rather than trying to be exhaustive, we will discuss the most relevant concepts. Furthermore, this overview will have more of a functional than an analytical bias, our goal being to encourage better practice in machine learning. In certain cases, this chapter will provide a brief introduction to a topic that will then be developed in more detail in later chapters.

The chapter is divided into four sections. In Section 2.1, we define the notion of random variables and their associated quantities. Section 2.2 then introduces the concept of probability distributions and discusses one of the extremely important results in statistics theory, the central limit theorem. Section 2.3 discusses the notion of confidence intervals. Finally, Section 2.4 briefly covers the basics behind hypothesis testing and discusses the concepts of type I and type II errors and the power of a test. If the reader is already acquainted with these concepts, they can confidently omit this chapter.

2.1 Random Variables

A random variable is a function that associates a unique numerical value with every outcome of an experiment. That is, a random variable can be seen as a measurable function that maps the points from a probability space to a measurable space. Here, by probability space we mean the space in which the actual experiments are done and the outcomes achieved. This need not be a measurable space, that is, the outcomes of an experiment need not be numeric. Consider the most standard example of a coin toss. The outcomes of a coin toss can be a "head" or a "tail." However, we often need to map such outcomes to numbers, that is, measurable space. Such a quantification allows us to study their behavior. We can precisely achieve this by using the notion of a random variable. Naturally, the range of values that a random variable can take would also depend on the nature of the experiment that it models. For a fixed set of outcomes of an experiment, such as the coin toss, a random variable results in discrete values. Such a random number is known as a discrete random variable. By contrast, a continuous random variable can model experiments with infinite possible outcomes.

The probabilities of the values that a random variable can take are also modeled accordingly. For a random variable x, these probabilities are modeled using a probability

distribution, denoted $P(x)$ when x is discrete and using a probability density function, denoted by $p(x)$, when x is continuous. As such, a probability distribution associates a probability with each of the possible values that a discrete random variable can take. It can thus be seen as a list of these probability values.

In the case of a continuous random variable, which can take an infinite number of values, we need a function that can yield the probability of the variable taking on values in a given interval. That is, we need an integrable function. The PDF fulfills these requirements.

In order to look at this closely, we first take a look at the cumulative distribution function (CDF). With every random variable, there is an associated CDF that provides the probability of the variable taking a value less than or equal to a value x_i for every x_i. That is, the CDF $p_{\text{cdf}}(x)$ is

$$p_{\text{cdf}}(x) = P(x \leq x_i), x \in \mathbb{R}.$$

Given a CDF, we can define the PDF $p(\cdot)$ associated with a continuous random variable x. The PDF $p(x)$ is the derivative of the CDF with respect to x:

$$p(x) = \frac{d}{dx} p_{\text{cdf}}(x).$$

If x_a and x_b are two of the possible values of x, then it follows that

$$\int_{x_a}^{x_b} p(x)dx = p_{\text{cdf}}(x_b) - p_{\text{cdf}}(x_a) = P(x_a < x < x_b),$$

where $\int(\cdot)$ denotes the integral operator. Hence, $p(x)$ can be a PDF of x if and only if for all $x \in \mathbb{R}$,

$$\int_{-\infty}^{\infty} p(x)dx = 1$$

and

$$p(x) > 0, x \in \mathbb{R}.$$

The expected value of a random variable x denotes its central value and is generally used as a summary value of the distribution of the random variable. The expected value generally denotes the average value of the random variable. For a discrete random variable x taking m possible values $x_i, i \in \{1, \ldots, m\}$, the expected value can be obtained as

$$\mathbf{E}[x] = \sum_{i=1}^{m} x_i P(x_i),$$

where $P(\cdot)$ denotes the probability distribution with $P(x_i)$ denoting the probability of x taking on the value x_i. Similarly, in the case when x is a continuous random variable with $p(x)$ as the associated PDF, the expected value is obtained as

$$\mathbf{E}[x] = \int_{-\infty}^{\infty} xp(x)dx.$$

In most practical scenarios, however, the associated probability distributions or PDFs are unknown. What is available is a set of values that the random variables take. In such cases

we can consider, when the size of this set is acceptably large, this sample as representative of the true distribution. Under this assumption, the sample mean can then be used to estimate the expected value of the random variable. Hence, if S_x is the set of values taken by the variable x then the sample mean can be calculated as

$$\overline{x} = \frac{1}{|S_x|} \sum_{i=1}^{|S_x|} x_i,$$

where $|S_x|$ denotes the size of the set S_x.

The expected value of a random variable summarizes its central value. However, it does not provide any indication about the distribution of the underlying variable by itself. That is, two random variables with the same expected value can have entirely different underlying distributions. A better sense of a distribution can be obtained by considering the statistics of variance in conjunction with the expected value of the variable.

The variance is a measure of the spread of the values of the random variable around its central value. More precisely, the variance of a random variable (probability distribution or sample) measures the degree of the statistical dispersion (the spread of values). The variance of a random variable is always nonnegative. Hence, the larger the variance, the more scattered the values of the random variable with respect to its central value. The variance of a random variable x is calculated as

$$\text{Var}(x) = \sigma^2(x) = \mathbf{E}[x - \mathbf{E}[x]]^2 = \mathbf{E}[x^2] - \mathbf{E}[x]^2.$$

In the continuous case, this means that

$$\sigma_2(x) = \int_{-\infty}^{\infty} (x - \mathbf{E}[x])^2 p(x) dx,$$

where $\mathbf{E}[x]$ denotes the expected value of the continuous random variable x and $p(x)$ denotes the associated PDF. Similarly, for the discrete case,

$$\sigma^2(x) = \sum_{i=1}^{m} P(x_i)(x_i - \mathbf{E}[x])^2,$$

where, as before, $P(\cdot)$ denotes the probability distribution associated with the discrete random variable x.

Given a sample of the values taken by x, we can calculate the sample variance by replacing the expected value of x by the sample mean:

$$\text{Var}_S(x) = \sigma_S^2 = \frac{1}{|S_x| - 1} \sum_{i=1}^{|S_x|} (x_i - \overline{x})^2.$$

Note that the denominator of the above equation is $|S_x| - 1$ instead of $|S_x|$.[1] The above estimator is known as the unbiased estimator of the variance of a sample. For large $|S_x|$, the difference between $|S_x|$ and $|S_x| - 1$ is rendered insignificant. The advantage of using

[1] This is known as Bessel's correction.

$|S_x| - 1$ is that in this case it can be shown that the expected value of the variance $\mathbf{E}[\sigma^2]$ is equal to the true variance of the sampled random variable.

The variance of a random variable is an important statistical indicator of the dispersion of the data. However, the unit of the variance measurement is not the same as the mean, as is clear from above. In some scenarios, it can be more helpful if a statistic is available that is comparable to the expected value directly. The standard deviation of a random variable fills this gap. The standard deviation of a random variable is simply the square root of the variance. When estimated from a population or sample of values, it is known as the sample standard deviation. It is generally denoted by $\sigma(x)$. This also makes it clear that using $\sigma^2(x)$ for variance denotes that the unit of the measured variance is the square of the expected value statistic. We calculate $\sigma(x)$ as

$$\sigma(x) = \sqrt{\text{Var}(x)}.$$

Similarly, the sample standard deviation can be obtained by considering the square root of the sample variance. One point should be noted. Even when using an unbiased estimator of the *sample* variance (with $|S_x|-1$ in the denominator instead of $|S_x|$), the resulting estimator is still *not* an unbiased estimator of the *sample* standard deviation.[2] Furthermore, it underestimates the true sample standard deviation. A biased estimator of the sample variance can also be used without significant deterioration. An unbiased estimator of the sample standard deviation is not known except when the variable obeys a normal distribution.

Another significant use of the standard deviation will be seen in terms of providing confidence to some statistical measurements. One of the main such uses involves the use of the standard deviation to provide confidence intervals, or margin of error, around a measurement (mean) from samples.

2.1.1 Performance Measures as Random Variables

The insights into the random variables and the related statistics we just presented are quite significant in evaluation. For instance, the performance measure of a classifier on any given dataset can be modeled as a random variable, and much of the subsequent analysis follows, enabling us to understand the behavior of the performance measure in both absolute terms and in terms relative to other performance measures or even the same performance measure across different learning settings. Various learning strategies have varying degrees of assumptions on the underlying distribution of the data. Given a classifier f resulting from applying a learning algorithm A to some training data S_{train}, we can test f on previously unseen examples from test data. Learning from inductive inference does make the underlying assumption, here, that the data for both the training and the test set comes from the same distribution. The examples are assumed to be sampled in an independently and identically distributed (i.i.d.) manner. The most general assumption that can be made is that the data (and possibly their labels) are assumed to be generated from some arbitrary

[2] This can be seen by applying Jensen's inequality to the standard deviation, which is a concave function unlike its square, the variance. We do not discuss these issues in detail since they are beyond the scope of this book.