Statistics and machine learning for high-energy physics

Harrison B. Prosper

Department of Physics, Florida State University, Tallahassee, FL 32306, USA

These lectures introduce some of the main ideas of frequentist and Bayesian statistics as well as supervised machine learning with a focus on the probabilistic interpretation of the latter. The ideas are illustrated using simple examples from particle physics.

1 Introduction

These lectures cover some of the key concepts and practices of statistics as well as the basic ideas of supervised machine learning. We aim to provide just enough detail to make the lectures self-contained. In discussing supervised machine learning, the focus is on foundational ideas rather than the nuts and bolts so that you gain an understanding of the probabilistic nature of machine learning. Given the striking abilities of computational models such the transformer, which powers systems like ChatGPT, it may not be immediately obvious where probability enters. But, as we shall see, systems like ChatGPT are "merely" highly sophisticated probabilistic machines.

This article should be cited as: Statistics and machine learning for high-energy physics, Harrison Prosper, DOI: 10.23730/CYRSP-2025-002.151, in: Proceedings of the 2023 CERN Latin-American School of High-Energy Physics, CERN Yellow Reports: School Proceedings, CERN-2025-002, DOI: 10.23730/CYRSP-2025-002, p. 151.

Statistics, like physics, is based on a set of mathematical rules. However, unlike physics, the rules of statistics are not informed by Nature and, consequently, we cannot appeal to Nature to adjudicate disagreements about whether a proposed statistical rule is valid or not. The primary cause of the disagreements among professional statisticians, which have lingered for more than two centuries, can be traced to the differing views about the interpretation of probability. In these lectures, we consider the two most important interpretations: *relative frequency* and *degree of belief*. The former interpretation is the basis of the *frequentist* approach to statistics, while the latter underpins the *Bayesian* approach. These interpretations are discussed later in this section.

The point of mentioning the disagreements is to alert you of the fact that in statistics there is no such thing as "the answer"; rather there are "answers", which often agree closely but sometimes do not. Therefore, in the practice of statistics a degree of pragmatism is necessary to avoid fruitless arguments about statistical practice that are ultimately about intellectual taste rather than mathematical correctness.

The lecture notes are organized as follows. The rest of the Introduction introduces some basic terminology. Section 2 covers the frequentist approach to statistics, while Section 3 introduces the Bayesian approach. Section 4 introduces supervised machine learning. Good introductions to statistical analysis for physicists may be found in the books: [1–4], while [5, 6] give excellent historical perspectives.

1.1 Samples

The result of an experiment is a sample of N data $X = x_1, x_2, \cdots, x_N$, which can be characterized with quantities called statistics[1]. A **statistic** is number that can be computed from the sample and may depend on one or more parameters. Here are a few well-known statistics that can be computed from the data alone:

the **sample moments**
$$x_r = \frac{1}{N}\sum_{i=1}^{N} x_i^r, \tag{1.1}$$

the **sample average**
$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{1.2}$$

and the **sample variance**
$$s^2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - \bar{x})^2. \tag{1.3}$$

The sample moments give detailed information about the sample, while the sample average and variance are measures of the center and spread of the data. Statistics that characterize the data, and are solely functions of the data, are called **descriptive statistics**. In these lectures, we shall encounter statistics that provide more sophisticated information about samples.

[1] Statisticians tend to use upper case letters to denote random variables and lower case letters to denote actual values. We do not follow this convention.

1.2 Populations

An infinitely large sample is an abstraction called a **population**, or an ensemble. A population can be summarized with numbers such as those listed below.

$$\begin{aligned}
&\text{Ensemble average} && E[x] \\
&\text{Mean} && \mu \\
&\text{Error} && \epsilon = x - \mu \\
&\text{Bias} && b = E[x] - \mu \\
&\text{Variance} && V = E[(x - E[x])^2] \\
&\text{Standard deviation} && \sigma = \sqrt{V} \\
&\text{Mean square error} && \text{MSE} = E[(x - \mu)^2] \\
&\text{Root MSE} && \text{RMS} = \sqrt{\text{MSE}}
\end{aligned} \tag{1.4}$$

(The symbol $E[*]$ means **ensemble average**, that is, the average over the population of the quantity within the brackets.) While it is important to keep in mind the logical distinction between a sample and its associated population, we frequently approximate populations with samples. Indeed, approximate populations are the basis of a statistical method called the bootstrap [7], in which various quantities can be approximated by treating a sample as if it were a population. In technical fields, from finance to high-energy physics, large simulated samples are often used to assess, for example, the effect of systematic uncertainties on final results or to confirm that an analysis method performs as claimed. In a simulated “population" some quantities can be computed exactly, for example the *error* associated with each element of the “population" can be computed because x is known and μ, a parameter of the simulation, is known by construction. Quantities such as bias, however, which require computing $E[x]$ remain approximate.

While it may not be possible to calculate a population quantity exactly, it is often possible to relate one population quantity to another, which can sometimes provide useful insight. For example, the mean square error (MSE), whose square root is called the root mean square (RMS)[2], can be written as

$$\text{MSE} = V + b^2. \tag{1.5}$$

Exercise 1: Show this

This is an instructive result. Suppose, for example, that μ is the true Higgs boson mass and x is a measurement of it. If the MSE is used as a measure of the accuracy of the mass measurements, then the result in Eq. (1.5) shows that correcting a measurement of the mass for bias makes sense only if, on the average, the bias-corrected results yield a smaller MSE than that of the uncorrected result. Making a bias correction may not always be the sensible thing to do if the goal is to arrive at mass measurements, which, on average, are as close to the true value of the mass as possible in the MSE sense. Using simulations to study and understand the characteristics of a population is both useful and educational. It is good practice to do many simple simulations (sometimes called *toy* experiments) to develop an intuition about

[2] The RMS and standard deviation are sometimes used interchangeably. The two quantities are identical only if the bias is zero.

statistical quantities and the behavior of statistical procedures as well as to decide whether a particular manipulation of a measurement—e.g., a bias correction—is useful.

Another example of the insight gained from studying a population is the calculation of the bias in the variance of a sample. When we speak of "bias in a measurement x", for example, a measurement of the Higgs boson mass, we should remember that this phrasing is shorthand for a precise but more cumbersome phrase. There is very likely an *error* in x, which in a real experiment is unknown[3] But *bias* is not directly a property of x. It is a property of the population to which x is presumed to belong. However, it would quickly become annoyingly pedantic to avoid the shorthand "bias in x", so it is reasonable to use this shorthand provided that we remember it is a proxy for something more precise. The ensemble average of the sample variance, Eq. (1.3), is given by

$$\begin{aligned} E[s^2] &= E[\overline{x^2}] - E[\bar{x}^2], \\ &= V - \frac{V}{N}, \end{aligned}$$

$$\boxed{\textbf{Exercise 2:}\text{ Show this}}$$

and has a bias of $b = -V/N$. The result shows that the bias can be calculated exactly only if the variance V is known exactly.

1.3 Statistical inference

One goal of a theory of statistical inference is to use a sample to infer something about the associated population. We may wish to estimate (that is, measure) a parameter associated with the population, for example, the mean Higgs boson signal in the proton-proton to 4-lepton channel. But to make this estimate meaningful, it is necessary to quantify its accuracy. Then we may wish to assess the degree to which we can claim the signal is real and not an apparent signal caused by a fluctuation of the background. We shall consider each of these tasks using the two most commonly used theories of inference, **frequentist** and **Bayesian**. In both theories, the foundational concept is **probability**, albeit interpreted in two different ways:

- **degree of belief** in, or assigned to, a proposition, e.g.,
  - *proposition*: it will rain in San Esteban tomorrow
  - *probability*: $p = 5 \times 10^{-2}$
- **relative frequency** of given outcomes in a large (strictly, infinite) set of trials, e.g.,
  - *trial*: a proton-proton collision at the Large Hadron Collider (LHC)
  - *outcome*: creation of a Higgs boson
  - *probability*: $p = 5 \times 10^{-10}$

Since each theory of inference uses a different interpretation of probability it is not surprising that the interpretation of their results differ even when both theories give numerically identical results. When data are plentiful, these interpretations usually do not affect how the results are subsequently used. Problems

[3] If the error were known, we could correct the measurement and get a perfect measurement!

arise when sample sizes are small. This is when the results of the two approaches can differ substantially and when intellectual taste becomes the main arbiter of which approach is considered the more reasonable.

The next two sections cover the application of frequentist and Bayesian theories of statistical analysis in particle physics using a simple real-world example, while the last section provides an introduction to supervised machine learning.

2 Frequentist analysis

In 2014, the CMS Collaboration published its measurement of the properties of the Higgs boson in the 4-lepton final states [8]. We shall analyze the summary results of this analysis, namely, $N = 25$ observed 4-lepton events with a background estimate of $B \pm \delta B = 9.4 \pm 0.5$ events. The goal is to make statements about the mean Higgs boson event count s—that is, the signal, where $d = s + b$ is the mean event count and b is the mean background count. Although these data are very simple, they are sufficient to illustrate the essential ideas of frequentist analysis.

Whether the data are to be analyzed using a frequentist or Bayesian approach, the starting point is the same: constructing an accurate probability, or statistical, model of the mechanism that generated the data. The terms probability model and statistical model shall be used interchangeably.

2.1 The statistical model

Given the observed count $N = 25$ events, a particle physicist would immediately model the data generation mechanism with a Poisson distribution,

$$\text{Poisson}(n, d) = \frac{e^{-d} d^n}{n!}.$$

If the data comprises M statistically independent counts $N_m, m = 1, \cdots, M$ the model generalizes to a product of Poisson distributions[4]. By statistically independent we mean that $E[n_i n_j] = E[n_i]E[n_j]$ for $i, j \in [1, \cdots, M]$, where the expectations are over the populations of counts $n \in \mathbb{N}$ defined by the Poisson probability mass function (pmf). If the random variables are from a continuous set, the statistical model is called a probability density function (pdf)[5]. But why is a Poisson pmf the appropriate model for a counting experiment? We can make this plausible with the following set of arguments.

2.1.1 Bernoulli trial

A Bernoulli trial, named after the Swiss mathematician Jacob Bernoulli (1654 – 1705), is an experiment with only two possible outcomes: S, a success or F, a failure. Each collision between protons at the LHC is a Bernoulli trial in which either a Higgs boson is created (S) or is not created (F). Here is a sequence of collisions results

$$F \quad F \quad S \quad F \quad F \quad F \quad F \quad S \quad F \quad \cdots$$

[4] Statistical analyses based on multiple counts are sometimes referred to as a *shape* analysis.

[5] In general, probability models can be made up of both pmfs and pdfs.

What is the probability of this sequence of results? No meaningful answer can be given. Unless, that is, we are prepared to make assumptions, such as the following.

1. Let p be the probability of a success.
2. Let p be the same for every collision (trial).
3. Let S and F be *exhaustive* (the only possible outcomes) and *mutually exclusive* (one outcome precludes the occurrence of the other).

Assumption 3 implies that the probability of F is $1-p$. Therefore, for a given sequence O of n proton-proton collisions, the probability $P(k|n,p,O)$ of exactly k successes and exactly $n-k$ failures is

$$P(k|n,p,O) = p^k(1-p)^{n-k}. \tag{2.6}$$

The specific sequence O of successes and failures is unknown at the LHC. So how are we to proceed? The rules of probability provide a general prescription: when a probability contains quantities that are either irrelevant or unknown, they can be eliminated from the problem by summing over all possible values of the unknown of irrelevant quantities, here the possible orders, O, of successes and failures. This prescription is called **marginalization** and is one of the most important rules in probability calculations. Applied to the problem at hand this rule yields,

$$P(k|n,p) = \sum_O P(k|n,p,O) = \sum_O p^k(1-p)^{n-k}. \tag{2.7}$$

Notice that every term in Eq. (2.7) is identical and there are $\binom{n}{k}$ of them. Therefore,

$$P(k|n,p) = \binom{n}{k} p^k(1-p)^{n-k}, \tag{2.8}$$

that is, we arrive at the **binomial distribution**, binomial(k, n, p). The alert reader may have noticed the sleight of hand in this derivation. In Eq. (2.7), we have assumed that every sequence O is equally probable. If a is the mean number of successes in n trials, then

$$\begin{aligned} a &= \sum_{k=0}^{n} k \,\text{binomial(k, n, p)}, \\ &= pn. \end{aligned} \tag{2.9}$$

$$\boxed{\textbf{Exercise 4:} \text{ Show this}}$$

For the Higgs boson outcomes, $p \sim 10^{-10}$ and $n \gg 10^{12}$. Therefore, it is reasonable to consider the limit $p \to 0$ and $n \to \infty$, while keeping a constant. In this limit

$$\begin{aligned} \text{Binomial(k, n, p)} &\to e^{-a}a^k/k!, \\ &\equiv \text{Poisson}(k,a). \end{aligned} \tag{2.10}$$

$$\boxed{\textbf{Exercise 5:} \text{ Show this}}$$

One conclusion that can be drawn from the above is that a Poisson distribution is, indeed, an appropriate model when the probability of individual events is extremely small and, crucially, when the probability of two or more events occurring in a very short time interval is negligible compared with the probability of zero or one event occurring in the given interval. In fact, the Poisson distribution can be derived from a stochastic model in which that assumption is made explicit. We conclude that it is reasonable to take

$$p(n|s,b) = \text{Poisson}(n, s+b) = \frac{(s+b)^n e^{-(s+b)}}{n!}, \tag{2.11}$$

as the probability to obtain a count n given mean event count $s+b$.

But notice how we keep hedging with imprecise words like “reasonable”. Why do this? Consider the common, and more interesting example, where we have M counts, which we can think of as constituting a histogram. We could write the following statistical model

$$p(\boldsymbol{n}|\boldsymbol{s},\boldsymbol{b}) = \prod_{m=1}^{M} \frac{(s_m+b_m)^{n_m} e^{-(s_m+b_m)}}{n_m!}, \tag{2.12}$$

for the histogram, which seems eminently sensible in view of our heuristic derivation of the Poisson pmf and our assumption that the counts are statistically independent. Suppose, however, that the total count $n = \sum_{m=1}^{M} n_m$ is taken to be constant. In that case, it would make sense to use a multinomial model,

$$p(\boldsymbol{n}|\boldsymbol{s},\boldsymbol{b}) = n! \prod_{m=1}^{M} \frac{p_m^{n_m}}{n_m!}, \quad p_m = \frac{s_m + b_m}{\sum_{m=1}^{M} s_m + \sum_{m=1}^{M} b_m}, \tag{2.13}$$

rather than a multi-Poisson model. But why assume that the total count is fixed? This question raises the thorny issue of which statistical model is appropriate for a given problem. This issue, which we seldom consider explicitly, is called the **reference class problem** [9]. The reference class is the population with respect to which probabilities are to be computed or assigned: should we consider a population in which the total count is fixed or should we consider one conditioned on an unconstrained total count? This ambiguity was well-known to the great British statistician Sir Ronald Fisher (1889 – 1962), who, late in life, noted that

> “None of the populations used to determine probability levels in tests of significance have objective reality, all being products of the statistician’s imagination”

The point is this: given data D there are many populations into which the data can be conceptually embedded. Since populations are abstractions, which exist only in the sense that π exists, there is no *operational* way to determine to which population the data D “belongs”. The reference class is defined by the assumptions that underlie the statistical model. The problem arises when there exists several plausible alternative assumptions we could adopt: change the assumptions and the reference class changes and invariably also the statistical model. For example, the $N = 25$ 4-lepton events observed by the CMS Collaboration could have been acquired after running for a fixed observation period or until N events were observed or until a certain integrated luminosity was reached. This ambiguity is an example of why a pragmatic disposition with respect to statistics is helpful and why following established practice, including conventions, is often necessary to make progress.

After this small philosophical detour, let's get back to Earth and build a statistical model for the background data. In principle, the model should encode in detail how the background estimate was obtained. But to keep things simple we shall assume that the background estimate was obtained from a Monte Carlo simulation, which yields $m = M$ simulated background events. The mean count in the simulation is kb, where k is a known scale factor that relates the mean count in the simulation to that in the signal region of the experiment that yielded $n = N$ events. The statistical model for the background is, therefore,

$$p(m|kb) = \text{Poisson}(m, kb). \tag{2.14}$$

Since the counts n and m are statistically independent, the full statistical model is

$$p(n, m|s, b) = \text{Poisson}(n, s + b)\text{Poisson}(m, kb). \tag{2.15}$$

2.2 The likelihood function

The **likelihood function** is the statistical model into which data have been inserted. The data comprise the counts N and M in the signal and background regions, respectively. Since the data are constants, the likelihood, $p(N, M|s, b)$, is a function of the parameters s and b only. Sometimes $p(N, M|s, b)$ is written as $L(s, b)$ to emphasize this point.

Unfortunately, we are given $B \pm \delta B$, not M and k. But with a plausible Ansatz, progress can be made. Let's assume that B and δB are M and $\sqrt{M}$ scaled down by the factor k, that is,

$$B = M/k, \tag{2.16}$$

$$\delta B = \sqrt{M}/k. \tag{2.17}$$

Since by assumption M is the result of a Monte Carlo simulation of the background the scale factor k is the ratio of the integrated luminosity associated with the simulation to that associated with the observed event sample of size N. If the background were computed from a signal-free sample that is otherwise like the signal sample, then k would be the scaling between the backgrounds in the two data samples. Inverting the equations in Eqs. (2.16) and (2.17) yields

$$M = (B/\delta B)^2 = 353.4, \tag{2.18}$$

$$k = B/\delta B^2 = 37.6. \tag{2.19}$$

Therefore, the likelihood for the count M is

$$(kb)^M e^{-kb}/\Gamma(M + 1), \tag{2.20}$$

written to allow for non-integral values of M. Writing $D = N, M$, the full likelihood is

$$p(D|s, b) = \frac{(s + b)^N e^{-(s+b)}}{N!} \frac{(kb)^M e^{-kb}}{\Gamma(M + 1)}. \tag{2.21}$$

In a more sophisticated version of this analysis, a probability model for the scale factor k would be included that to account for the uncertainty in k.

It is important to appreciate the fact that Eq. (2.21) cannot be said to be *the* answer to the question: what is the likelihood function for the data $N, B \pm \delta B$. A deep dive into how the background estimate $B \pm \delta B$ was arrived at by the CMS Collaboration would yield a much more sophisticated statistical model. Indeed, the models in use today at the LHC can be enormously complicated, which effectively precludes their reconstruction by readers without access to the detailed knowledge known only to the collaborations. This is one reason why all collaborations are strongly encouraged [10] to make the publication of statistical models and likelihoods a cultural habit of high-energy physicists.

Now that we have the statistical model and the associated likelihood function, we are equipped to answer several questions, including the following.

1. How is a parameter to be estimated, that is, measured?
2. How is its accuracy to be quantified?
3. How can an hypothesis be tested?
4. How is the statistical significance of the result to be quantified?

As alluded to, we should not expect unique answers to these questions, but we hope to have answers founded on plausible and even cogent assumptions and arguments.

2.3 The frequentist principle

The goal of a frequentist analysis is to construct statements with the *a priori* guarantee that a fraction $f \geq p$ of them are true. This stipulation is called the **frequentist principle** (FP) and was championed by the Polish statistician Jerzy Neyman [11] and dominates statistical thinking in many scientific fields including high-energy physics. The fraction f is called the **coverage probability**, or coverage for short, and p is called the **confidence level** (CL). An ensemble of statements that obey the frequentist principle is said to *cover*.

Points to Note

1. The FP applies to real ensembles [6], not just the virtual ones simulated on a computer. Moreover, the ensembles can contain statements about different quantities. *Example*: all published measurements x, since the discovery of the electron in 1897, yielding statements of the form $\theta \in [l(x), u(x)]$, where θ is a parameter of interest, that is, the parameter to be measured.
2. Coverage is an *objective* characteristic of samples of statements. However, to verify whether a sample of statements covers, we need to know which statements are true and which ones are false. Unfortunately, for real experiments we are not privy to this information; therefore, there is no *operational* way to compute the coverage in actual samples. In a simulation, however, we can compute the coverage because we can identify the true statements. High-fidelity simulations of all

[6] Strictly speaking, we mean real *samples* because, as we have defined it, an ensemble is a synonym for a population, which by definition contains infinitely many elements, and, as noted earlier, is therefore and abstraction.

published results may give us confidence that the actual coverage of published statements is as the simulation reports, but that does not prove that it is so.

Example

Consider an ensemble of different experiments, each with a different mean count θ, and each yielding a count N. Each experiment makes a single statement of the form

$$N + \sqrt{N} > \theta,$$

which is either true or false. As noted above, if these were real experiments, we would not be able to determine which statements are true and which are false because we would not know the values of θ and, therefore, we would not be able to determine the coverage. But in a simulation we know which statements are true and which are false. Suppose that in a simulation each mean count θ is randomly sampled from a uniform distribution (`uniform`(0, 10)), with range [0, 10]. Since the mean counts are known, we can compute the coverage probability f.

Exercise 7: Compute the coverage of these statements; repeat the exercise using `uniform`(0, 1000)

The next section discusses the important concept of the confidence interval, which is the classic exemplar of the frequentist principle.

2.4 Confidence intervals

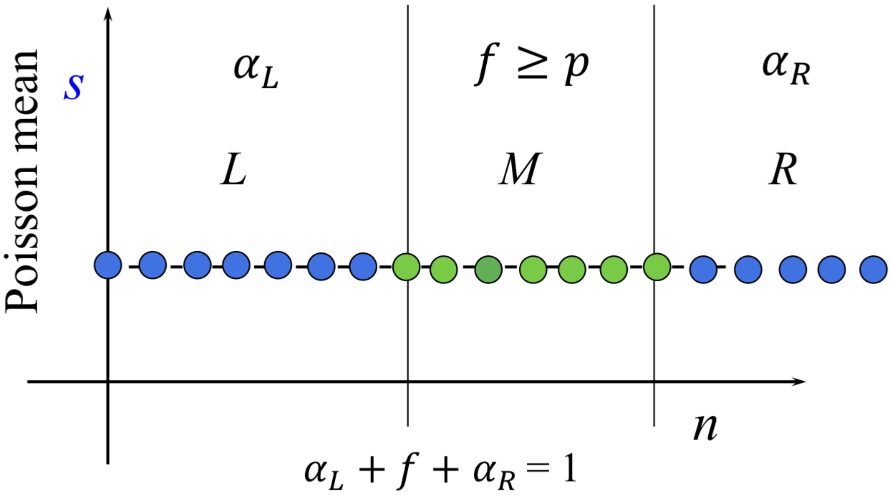

Fig. 1: Plotted is the tensor product of the parameter space, with parameter s, and the space of observations with potential observations n. For a given value of s, the observation space is partitioned into three disjoint intervals, labeled L, M, and R, such that the probability to observe a count n in M is $f \geq p$, where p = is the desired confidence level.

In 1937, Neyman [11] introduced the concept of the **confidence interval**, a way to quantify uncertainty that respects the frequentist principle. Confidence intervals are a concept best explained through an example. Consider an experiment that observes $n = N$ events with mean signal count s and no background. A confidence interval $[l(N),\, u(N)]$, with confidence level CL $= p$, permits a statement of the form

$$s \in [l(N),\, u(N)], \tag{2.22}$$

with the *a priori* guarantee that a fraction $f \geq p$ of them will be true. Neyman repeatedly emphasized that the statements need not be about the same quantity or arise from the same kind of experiment. What matters is that they are constructed using a method that satisfies the frequentist principle. For simplicity, we consider experiments of the same kind, but which differ by their mean signal count s.

Figure 1 shows the tensor product of the parameter space $\{s\}$ and the space of potential observations $\{N\}$ as well as the potential observations, represented by the dots, of an experiment with mean count s. The two vertical lines divide the space of observations into the three regions labeled L, M, and R. The region M is chosen so that the probability to obtain a count in that region is $f \geq p$, where p is the desired confidence level (CL). The probabilities to obtain a count in region L or region R are α_L and α_R, respectively. Since the three regions span the space of observations, $\alpha_L + f + \alpha_R = 1$.

The choice of the confidence level p does not uniquely specify the region M. Different methods have been suggested to define M. The first method was devised by Neyman [11], which we shall consider shortly. Another method was suggested by Feldman and Cousins [12]. The Feldman-Cousins method will serve to illustrate a general method to construct confidence intervals that satisfy the frequentist principle, at least for statistical models with a single unknown parameter.

Feldman-Cousins Method

In the Feldman-Cousins method, every potential count n is associated with a pair of numbers: a weight $p(n|s) \,/\, p(n|\hat{s})$, where $\hat{s}(n) = n$ is the maximum likelihood estimator of s, together with the probability $p(n|s)$ to obtain that count. (An **estimator** is a procedure, often just a function but it could be an entire analysis program, which when data are entered into it furnishes an estimate. To lighten the prose, we will typically not distinguish between estimators and estimates, though by doing so we are making what philosophers refer to a category mistake.) The counts are placed in *descending* order of their weights. Starting with the first count in the ordered list, a set of counts $(n_{(1)}, n_{(2)}, \cdots)$ is accumulated one by one until their summed probabilities $f = \sum_{(i)} p(n_{(i)}|s) \geq p$. The symbol (i) denotes the ordinal value of a count in the ordered list. The set of counts $(n_{(1)}, n_{(2)}, \cdots)$ defines an interval in the space of observations whose lowest (leftmost) and highest (rightmost) counts n_L and n_R are given by $n_L = \min(n_{(1)}, n_{(2)}, \cdots)$ and $n_R = \max(n_{(1)}, n_{(2)}, \cdots)$, respectively. This construction (for this single parameter problem) guarantees that the probability to obtain a count within region M is $f \geq p$ [7].

There is, however, a potential pitfall with any algorithm to define M. The region M can only be defined if the mean count associated with the experiment is known. But if we knew that we wouldn't need to do the experiment! It may well be true that we know s within a simulation, but it is not so in real experiment. Therefore, any algorithm for defining the region M must be repeated for every value of s

[7] We write $f \geq p$ rather than $f = p$ because, in general, for a discrete distribution it is not possible to satisfy the equality except at specific values of s.

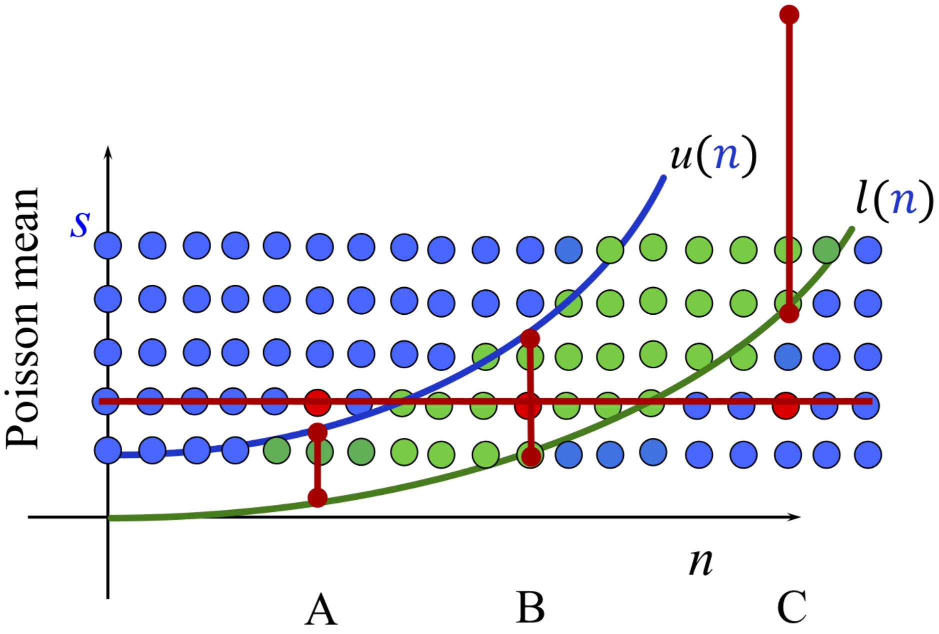

Fig. 2: The algorithm for defining region M (see Fig. 1), must be repeated for every value of s that is possible *a priori*. For the experiment whose mean s is represented by the thick horizontal line, the figure shows three possible outcomes, labeled A, B, and C, and their associated confidence intervals $[l(n), u(n)]$. Only outcomes, such as B, which lie within the region M of the experiment will yield intervals that bracket s. The probability to obtain such an interval is $f \geq p$, by construction.

that is possible *a priori*, as illustrated in Fig. 2. Repeating the Feldman-Cousins algorithm for different (closely-spaced) values of s produces regions M_s, indexed by the mean count s. The concatenation of these regions defines two curves labeled $l(n)$ and $u(n)$ in Fig. 2. For a given n, these curves define the confidence intervals $[l(n), u(n)]$, that is, sets of parameters that depend on n. Over an ensemble of experiments—and irrespective of their associated mean count s, the fraction of statements of the form $s \in [l(n),\ u(n)]$ that are true is $f \geq p$, by construction.

To see this, consider again Fig. 2. It shows three possible outcomes for the experiment defined by the thick horizontal line together with three possible confidence intervals (the vertical lines terminated with dots). If an observation lands in the region M for that experiment, the interval $[l(n), u(n)]$ will bracket the mean count s, as shown in the figure. If a count lands in region L, then the upper limit $u(n)$ will lie below s and, consequently, the interval $[l(n), u(n)]$ will exclude s. If n lands in region R, then the lower limit $l(n)$ will lie above s and the interval will exclude s. Therefore, the interval $[l(n), u(n)]$ will include s only if n lies in M, for which the relative frequency is $f \geq p$. A procedure for constructing confidence intervals in this manner is called a **Neyman construction**.

Neyman Method

The algorithm described above requires that a region M be constructed for each value of s. A more straightforward algorithm was given by Neyman in his 1937 paper and is illustrated in Fig. 3. For every

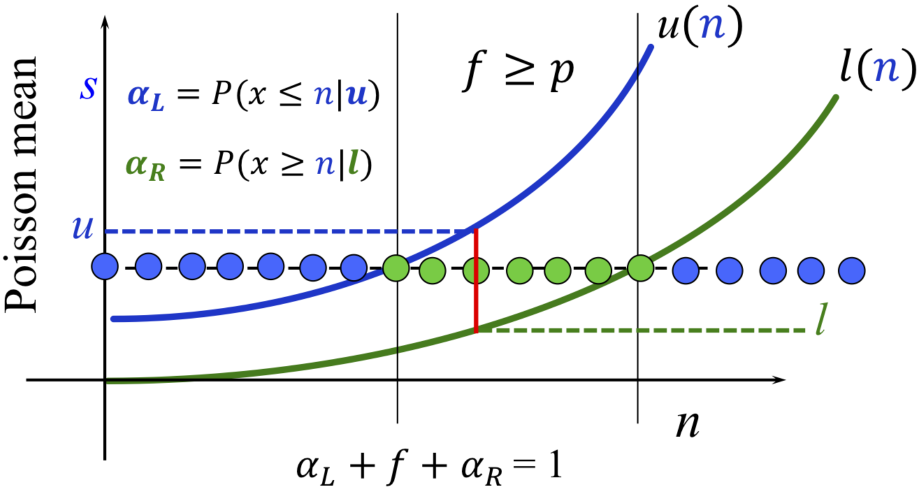
Poisson mean
s
u
$\alpha_L = P(x \leq n|u)$
$\alpha_R = P(x \geq n|l)$
$f \geq p$
$u(n)$
$l(n)$
l
n
$\alpha_L + f + \alpha_R = 1$

Fig. 3: The Neyman method. For every n, an interval $[l(n), u(n)]$ is computed by solving the equations in the plot. See text for details.

n, the upper and lower limits are found by solving

$$P(x \leq n|u) = \alpha_L, \tag{2.23}$$

$$P(x \geq n|l) = \alpha_R. \tag{2.24}$$

Equation (2.23) yields the curve $u(n)$ for which the probability to obtain a count $x \leq n$, for a given s, is α_L, while Eq. (2.24) yields a curve $l(n)$ for which the probability to obtain a count $x \geq n$, for a given s, is α_R. Therefore, every horizontal line will necessarily partition the space of observations into three regions L, M, and R as described above. The curves $l(n)$ and $u(n)$ can also be made using the procedure described above for the Feldman-Cousins method, but Neyman's solution based on Eqs. (2.23) and (2.24) is computationally more efficient.

Figure 4 shows the coverage probability over the parameter space for the Neyman intervals, in which we have chosen $\alpha_L = \alpha_R = (1 - p)/2$. This choice, the one made by Neyman, define **central confidence intervals**. As advertised, these intervals satisfy the frequentist principle. Also shown is the coverage for intervals of the form $[N - \sqrt{N}, N + \sqrt{N}]$ and $[N - \sqrt{N}, N + \sqrt{N} + \exp(-N)]$. These intervals are *approximate* confidence intervals in that they do not satisfy the frequentist principle exactly. Notice, however, that for $s > 2.5$ the coverage of these intervals bounces around the $p = 0.683$ line. Therefore, over a large sample of experiments, with a distribution of Poisson means, it is plausible that the **marginal coverage** would turn out to be close to the desired confidence level using the simpler intervals. Marginal coverage, that is, coverage averaged over the parameter space, is a weaker condition than is required by the frequentist principle, which demands **conditional coverage**, that is, coverage point-by-point in the parameter space. In practice, it is seldom possible to achieve this exactly.

A notable feature of Fig. 4 is the jaggedness of the coverage probabilities over the parameter space. The jaggedness is caused by the discreteness of the Poisson distribution. For a discrete distribution, coverage equal to the desired confidence level is possible only at specific values of s. Therefore, if we insist on the frequentist principle, $f \geq p$, the price to be paid is *over-coverage* over the whole parameter space except over a set of measure zero.

We have yet to mention the most important probability model in statistics, namely, the Gaussian,

$$\text{Gaussian}(x, \mu, \sigma) = \frac{e^{-\frac{1}{2}(x-\mu)^2/\sigma^2}}{\sigma\sqrt{2\pi}}, \tag{2.25}$$

or normal, probability density, where μ is the mean of the density and σ its standard deviation. The random variable x lies in the set $\mathbb{R}$. If Neyman's prescription for computing confidence intervals, Eqs. (2.23) and (2.24), is applied to the Gaussian, where now n is to be regarded as a continuous quantity, the one-standard-deviation interval $[x - \sigma, x + \sigma]$ satisfies the condition $\alpha_L = \alpha_R = (1 - p)/2$ with $p = 0.683$. That is, fixed-width intervals of this form are confidence intervals with confidence level 68.3%. It is this fact about the Gaussian that is the origin of the convention to quote 68.3% confidence intervals when reporting a measurement even for non-Gaussian statistical models. The Gaussian is important because, stated loosely, every sensible probability distribution becomes Gaussian as the sample size increases without limit. Another important and closely related probability density is that of the sum z of k quantities of the form $(x - \mu)^2/\sigma^2$ with each random variable x in the sum sampled from a Gaussian. The density of z is given by

$$\text{Chisquared}(z, k) = \frac{1}{2^{k/2}\Gamma(k/2)} z^{k/2-1} e^{-z/2}. \tag{2.26}$$

The integer k is called the degrees of freedom. Observe that for $k = 1$ the solutions of the equation $z = (x - \mu)^2/\sigma^2 = 1$ are $l(x) = x - \sigma$ and $u(x) = x + \sigma$, that is, the lower and upper bounds of the 68.3% CL intervals. We'll return to this important fact later.

2.5 The profile likelihood

The likelihood function,

$$p(D|s, b) = \frac{(s + b)^N e^{-(s+b)}}{N!} \frac{(kb)^M e^{-kb}}{\Gamma(M + 1)}, \tag{2.27}$$

contains *two* parameters, the mean signal count s and mean background count b. The Neyman construction can be extended to any number of parameters. Therefore, in principle, it is always possible to construct regions in the full parameter space of a statistical model called **confidence sets** that satisfy the frequentist principle exactly. (A confidence interval is just a 1-dimensional confidence set.) However, in this problem the **parameter of interest** is the mean signal s. The mean background count is needed to define the probability model, but is otherwise not of current interest. The parameter b is an example of a **nuisance parameter**. If we wish to make inferences about the parameters of interest irrespective of the true values of the nuisance parameters, we must rid the problem of *all* nuisance parameters; we need eliminate b from the problem. A very general and widely accepted method for doing so is to convert the likelihood function into a function called the **profile likelihood**. But before discussing this, we briefly describe the most common frequentist method to arrive at estimates of parameters, a method we mentioned above without comment.

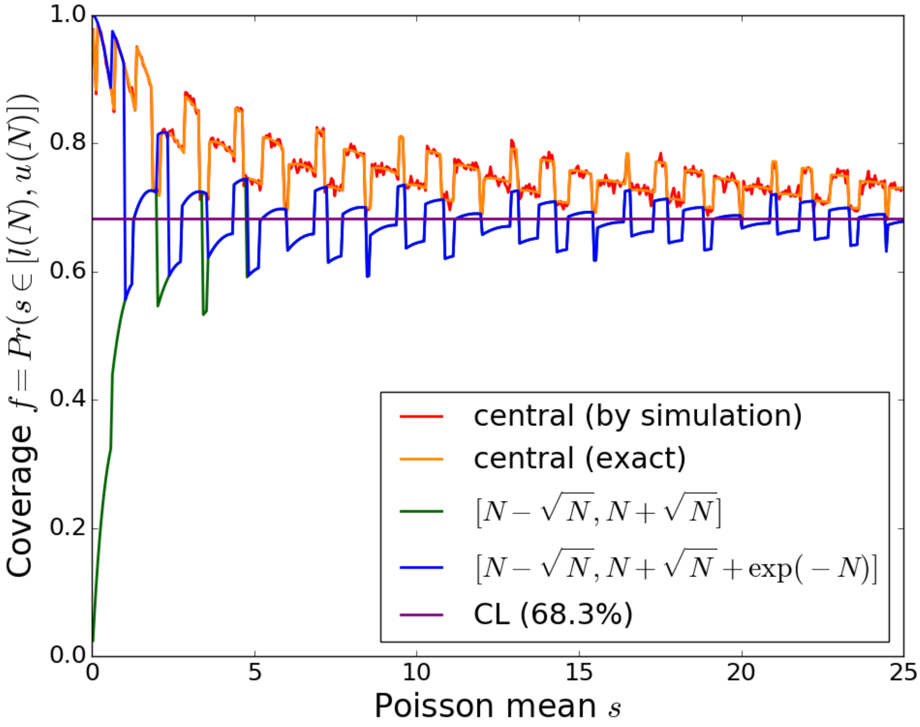

Fig. 4: Coverage probability f as a function of the Poisson mean s. As expected, the central intervals satisfy the frequentist principle, namely, $f \geq p$, where $p = 0.683$ is the confidence level. The coverage for two other sets of intervals are shown for which the frequentist principle is not satisfied exactly.

Given the likelihood function $L(s, b) \equiv p(D|s, b)$, its parameters can be estimated by maximizing $L(s, b)$ or equivalently maximizing $\ln L(s, b)$ with respect to s and b,

$$\begin{aligned} \frac{\partial \ln p(D|s, b)}{\partial s} &= 0 \quad \text{leading to } \hat{s} = N - B, \\ \frac{\partial \ln p(D|s, b)}{\partial b} &= 0 \quad \text{leading to } \hat{b} = B, \end{aligned}$$

as expected, recalling that $B = M/k$. Estimates found this way (first done by the Prince of Mathematicians Karl Frederick Gauss and systematically developed by Sir Ronald Fisher [13]) are called **maximum likelihood estimates** (MLE). The method generally leads to satisfactory estimates, but, as is true of other procedures in statistical analysis, the method has its good and bad features, as noted below.

- *The Good*
  - Maximum likelihood estimates are *consistent*, that is, the RMS of estimates goes to zero as more and more data are included in the likelihood. This basically says that acquiring more data is worthwhile because the accuracy of results is expected to improve.
  - If an *unbiased* estimate of a parameter exists, the maximum likelihood procedure will find it.
  - Given the MLE for s, the MLE for any function $y = g(s)$ of s is simply $\hat{y} = g(\hat{s})$. This is an extremely useful feature because it makes it possible to maximize the likelihood using any convenient parameterization of it, say s, because at the end we can transform back to the

parameter of interest using $\hat{y} = g(\hat{s})$.

- *The Bad*
  - In general, MLEs are biased.

Exercise 7: Show this
Hint: Taylor expand $\hat{y} = g(s + \hat{s} - s)$ about s and consider its ensemble average.

- *The Ugly*
  - Most MLEs are biased, which, unfortunately, encourages the routine application of bias correction. But correcting for bias makes sense only if the RMS of an unbiased result is less than or equal to the RMS of a biased result. Recall that the RMS $= \sqrt{V + b^2}$, where V is the variance and b is the bias.

Returning to the profile likelihood, we note that to make an inference about the mean signal count, s, the 2-parameter model $L(s, b)$ must be reduced to one involving s only. In principle, this must be done while respecting the frequentist principle, that is, $f \geq p$, where f is the coverage probability of an ensemble of statements and p is the desired confidence level. In practice, all nuisance parameters are replaced by their MLEs conditional on the parameters of interest. For the Higgs boson example, an estimate of b, $\hat{b}(s)$, is found conditional on s and b is replaced by $\hat{b}(s)$ in $L(s, b)$. This leads to a new function $L_p(s) = L(s, \hat{b}(s))$ called the **profile likelihood**. For the likelihood in Eq. (2.27),

$$\begin{aligned} \hat{b}(s) &= \frac{g + \sqrt{g^2 + 4(1+k)Ms}}{2(1+k)}, \\ \text{where } g &= N + M - (1+k)s. \end{aligned} \tag{2.28}$$

Figure 5 shows a density plot of the likelihood $L(s, b)$ with the function $\hat{b}(s)$ superimposed. Notice that $\hat{b}$ goes through the mode of $L(s, b)$, which occurs at $s = \hat{s} = N - B = 15.6$ events. Figure 6 shows the profile likelihood. Replacing the (unknown) true value of b with an estimate of it is clearly an approximation. Therefore, it should come as no surprise that inferences based on the profile likelihood are not guaranteed to be satisfy the frequentist principle exactly. However, it is found that for the typical applications in high-energy physics (as will be evident below), the procedures based on the profile likelihood work surprisingly well. Moreover, the use of the profile likelihood has a sound theoretical justification. Consider the **profile likelihood ratio**

$$\lambda(s) = L_p(s)/L_p(\hat{s}), \tag{2.29}$$

where $\hat{s}$ is the MLE of s. Taylor expand the associated quantity

$$t(s) = -2 \ln \lambda(s) \tag{2.30}$$

about $\hat{s}$,

$$t(\hat{s} + s - \hat{s}) = t(\hat{s}) + t'(\hat{s})(s - \hat{s})$$

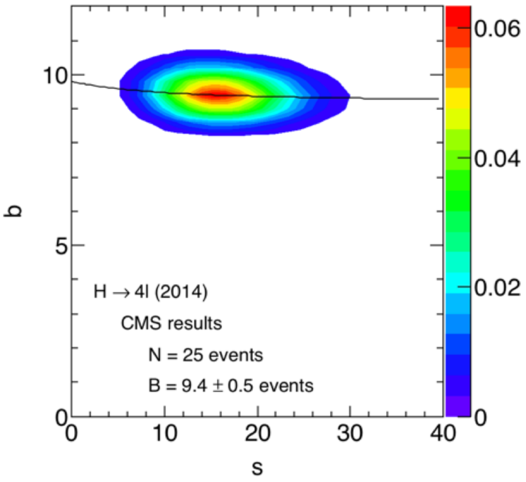

Fig. 5: The likelihood $L(s, b)$ and the graph of the function $\hat{b}(s)$.

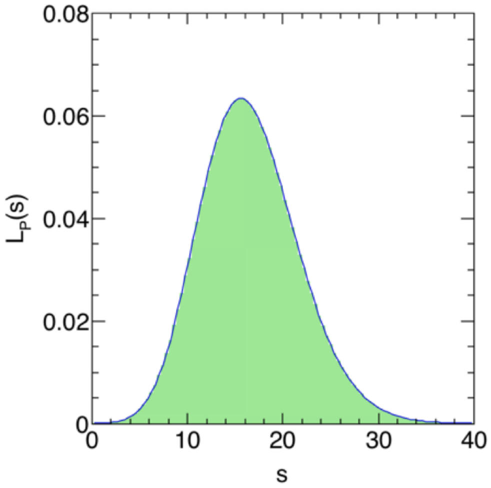

Fig. 6: The profile likelihood $L_p(s) \equiv L(s, \hat{b}(s))$.

$$\begin{aligned} &+ t''(\hat{s})(s-\hat{s})^2/2 + \cdots \\ &\approx (s-\hat{s})^2/2/\sigma^2 + \cdots, \\ \text{where } &\sigma^2 \approx 2/t''(\hat{s}). \end{aligned} \tag{2.31}$$

The quadratic approximation is called the Wald approximation (1943) (see Cowan et al. [14]). An important result obtains if certain so-called regularity conditions are met: 1) if $\hat{s}$ does not lie on the boundary of the parameter space (in which case the derivative of t at $\hat{s}$ is zero), 2) the sample is large enough (that is, when the density of $\hat{s}$ is approximately Gaussian$(\hat{s}, s, \sigma)$), and 3) if s is the true value of the mean signal count, then the density of $t(s)$ converges to a χ^2 density of one degree of freedom. The result, which is important because of its generality, is a special case of Wilks' theorem (1938) (Cowan et al. [14]).

Since $t(s) \approx \chi^2$, we can use the fact noted above that $\hat{s}$ is Gaussian-distributed then the solution of $\chi^2 = (s-\hat{s})^2/\sigma^2 = 1$ yields a 68% confidence interval. Therefore, we can compute an *approximate*

68% confidence interval by solving

$$t(s) = -2 \ln \lambda(s) = 1, \tag{2.32}$$

for the lower and upper limits of the interval. Given $N = 25$ observed 4-lepton events, a background estimate of $B \pm \delta B = 9.4 \pm 0.5$, we can state that

$$s \in [10.9,\ 21.0] \quad @\ 68\%\ \text{CL} \tag{2.33}$$

Exercise 8: Verify this interval.

As noted, intervals constructed using the profile likelihood are not guaranteed to satisfy the frequentist principle exactly. However, for applications in high-energy physics the coverage of these intervals is usually very good even for small quantities of data.

2.6 Hypothesis tests

In the previous section, we concluded that $s \in [10.9, 21.0]$ @ 68% CL. This result strongly suggests that a signal exists in the $N = 25$ 4-lepton events observed by CMS. But a qualitative statement such as this is generally considered insufficient. The accepted practice is to perform an hypothesis test. Indeed, in particle physics, a discovery is declared only if a certain quantitative threshold has been reached in an hypothesis test.

An hypothesis test in the frequentist approach is a procedure for *rejecting* an hypothesis that adheres to the following protocol.

1. Decide which hypothesis is to be *rejected.* This is called the **null hypothesis**. At the LHC this is usually the background-only hypothesis.
2. Construct a function of the data called a **test statistic** with the property that large values of it would cast doubt on the veracity of the null hypothesis.
3. Choose a test statistic threshold above which we agree to reject the null hypothesis. Do the experiment, compute the statistic, and reject the null if the threshold is breached.

There are at least two related variants of this protocol, one by Fisher [13] and the other by Neyman, both developed in the 1930s. Fisher and Neyman disagreed strenuously about hypothesis testing, which suggests that the topic is rather more subtle than it seems. Fisher held that an hypothesis test required consideration of the null hypothesis only, while Neyman argued that a proper test required consideration of both a null as well as an alternative hypothesis. Physicists ignore these disagreements and see utility in a shotgun marriage of the two approaches. This is eminently pragmatic, whereas our quasi-religious adherence to a 5σ threshold before declaring a discovery is not always sensible.

We first illustrate Fisher's theory of hypothesis testing and follow with a description of Neyman's theory.

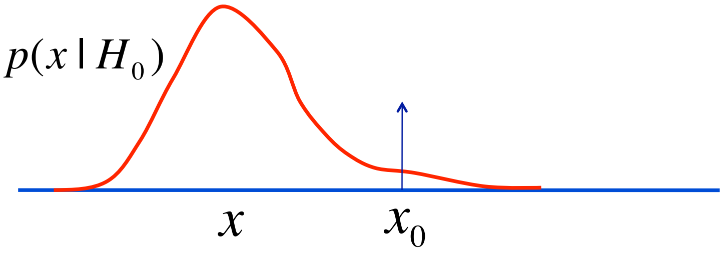

Fig. 7: The p-value is the tail-probability, $P(x > x_0|H_0)$, calculated from the probability density under the null hypothesis, H_0. If the null hypothesis is true then the probability density of the p-value under the null hypothesis is uniform$(0, 1)$.

Fisher's Approach

Suppose that the null hypothesis, which is denoted by H_0, is the background-only hypothesis, that is, the Standard Model without a Higgs boson[8] and compute a measure of the incompatibility of H_0 with the observations, called a **p-value**, defined by

$$\text{p-value}(x_0) = P(x > x_0|H_0), \tag{2.34}$$

where x is a test statistic, designed so that large values indicate departure from the null hypothesis, and x_0 is the observed value of the statistic. Figure 7 shows the location of x_0. The p-value is the probability that x could have been equal to or higher than x_0. Fisher argued that a sufficiently small p-value implies that either the null hypothesis is false or something rare has occurred. If the p-value is extremely small, say $\sim 3 \times 10^{-7}$, then of the two possibilities the response of the high-energy physicist is to reject the null hypothesis, that is, the background-only hypothesis and declare that a discovery has been made. The p-value for our example, neglecting the uncertainty in the background estimate, is

$$\text{p-value} = \sum_{k=N}^{\infty} \text{Poisson}(k, 9.4) = 1.76 \times 10^{-5}, \text{ with } N = 25.$$

Since the value of p-value is somewhat non-intuitive, it is conventional to map it to a Z**-value**, that is, the number of standard deviations the observation is *away from the null* if the distribution were a Gaussian. The Z-value can be computed using [9].

$$Z = \sqrt{2}\,\text{erf}^{-1}(1 - 2\text{p-value}). \tag{2.35}$$

A p-value of 1.76×10^{-5} corresponds to a Z of 4.14σ. The Z-value can be calculated using the `Root` function

$$Z = \texttt{TMath::NormQuantile(1-p-value)}.$$

If the p-value is judged to be small enough or the Z-value is large enough then the background-only hypothesis is rejected.

[8] That is, a thoroughly inconsistent theory!

[9] $\text{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^{x} \exp(-t^2)\, dt$ is the error funtion.

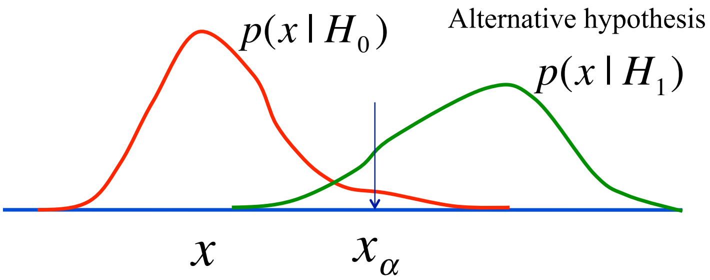

Fig. 8: Distribution of a test statistic x for two hypotheses, the null H_0 and the alternative H_1. In Neyman's approach to testing, $\alpha = P(x > x_\alpha|H_0)$ is a *fixed* probability called the significance of the test, which for a given class of experiments corresponds the threshold x_α. The hypothesis H_0 is rejected if $x > x_\alpha$.

Neyman's Approach

As noted, Neyman insisted that a correct hypothesis test required consideration of *two* hypotheses, the null hypothesis H_0 and an alternative hypothesis H_1. This is illustrated in Fig. 8. The null is the same as before but the alternative hypothesis is the Standard Model with a Higgs boson, that is, the background plus signal hypothesis. Again, the statistic x is constructed so that large values would cast doubt on the validity of H_0. However, the Neyman test is specifically designed to respect the frequentist principle. A *fixed* probability α called the **significance (or size) of the test** is chosen, which corresponds to some threshold value x_α defined by

$$\alpha = P(x > x_\alpha|H_0). \tag{2.36}$$

Should the observed value $x_0 > x_\alpha$ or equivalently the p-value(x_0) $< \alpha$ then the hypothesis H_0 is rejected in favor of the alternative. By construction if the null hypothesis is true then repeated application of this test will reject the null hypothesis a fraction α of the time. These *false* rejections are called **Type I errors**. Neyman's test discards the p-value and reports only α and whether or not the null was rejected. However, in high-energy physics, in addition to reporting the results of the test, and perhaps announcing a discovery, we also report the *observed* p-value. This is good practice because the observed p-value provides more information than merely reporting the fact that a null hypothesis was rejected at a significance level of α.

Given that Neyman's test requires an alternative hypothesis there is more that can be said than simply reporting the result of the test and the observed p-value. Figure 8 shows that we can also calculate

$$\beta = P(x \leq x_\alpha|H_1), \tag{2.37}$$

which is the relative frequency with which we reject a true alternative hypothesis H_1 if it is true. This mistake is called a **Type II error**. The quantity $1 - \beta$ is called the **power** of the test and is the relative frequency with which we would accept the true alternative hypothesis upon repeated application of the test. The defining feature of the Neyman test is that, in accordance with the Neyman-Pearson lemma (see for example Ref. [1]), the power is maximized subject to the constraint that α is fixed. The Neyman-

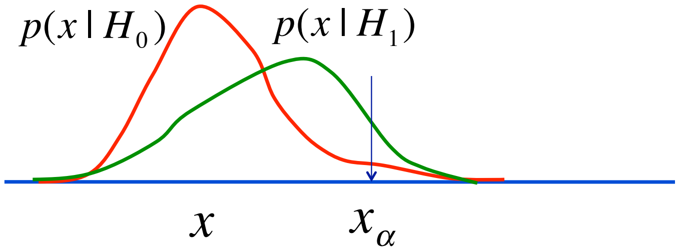

Fig. 9: See Fig. 8 for details. Unlike the case in Fig. 8, the two hypotheses H_0 and H_1 are not that different. It is then not clear whether it makes practical sense to reject H_0 when $x > x_\alpha$ only to replace it with an hypothesis H_1 that is not much better.

Pearson lemma asserts that given two simple hypotheses—that is, hypotheses in which all parameters have specified values—the optimal test statistic t for conducting an hypothesis test is the likelihood ratio $t = p(x|H_1)/p(x|H_0)$.

Maximizing the power seems like a reasonable procedure. Consider Fig. 9, which shows that the significance of the test in this figure is the same as that in Fig. 8. Therefore, the Type I error rates are identical. However, the Type II error rate is much greater in Fig. 9 than in Fig. 8 because the power of the test is considerably weaker in the former. Consequently, it is debatable whether rejecting the null is a wise course of action since the alternative hypothesis is not that much better. This insight was one source of Neyman's disagreement with Fisher. Neyman objected to the possibility that one might reject a null hypothesis regardless of whether it made sense to do so. He argued that the goal of hypothesis testing is always one of deciding between competing hypotheses. Fisher's counter argument was that an alternative hypothesis may not be available, in which case we either give up or we have a method to test the only hypothesis that is available and to decide whether it is worth keeping. In a Bayesian analysis an alternative hypothesis is also needed, in agreement with Neyman viewpoint, but is used in a way that neither he nor Fisher agreed with.

So far we have assumed that the hypotheses H_0 and H_1 are simple, that is, fully specified. Alas, most of the hypotheses that arise in realistic high-energy physics analyses are not of this kind. In the Higgs boson example, the probability models depend on a nuisance parameter for which only an estimate is available. Consequently, neither the background-only nor the background plus signal hypotheses are fully specified. Such hypotheses are examples of **compound hypotheses**. In the following, we illustrate how hypothesis testing proceeds in this case using the 4-lepton example.

Compound Hypotheses

In Sec. 2.5, we reviewed the standard way nuisance parameters are handled in a frequentist analysis, namely, their replacement by their conditional MLEs, thereby converting the likelihood function to the profile likelihood. In the 4-lepton example, this yielded the function $L_p(s) = L(s, f(s))$. The justification for this is that the statistic $t(s) = -2\ln\lambda(s)$, where $\lambda(s) = L_p(s)/L_p(\hat{s})$ and $\hat{s}$ is the MLE of s can be used to compute (approximate) confidence intervals in light of Wilks' theorem, which as noted above essentially states that $t(s) \approx \chi^2$. Therefore, the same statistic can also be used as a test statistic with the associated p-values calculated using the χ^2 density. Moreover, since, by definition, $Z = \sqrt{\chi^2}$, the p-value calculation can be sidestepped altogether. Using $N = 25$ and $s = 0$, we find $\sqrt{t(0)} = 4.13$, which is to be compared with $Z = 4.14$, the value found neglecting the ± 0.5 event uncertainty in the background.

In summary, the statistic $t(s)$ can be used to test null hypotheses as well as compute confidence intervals and, therefore, provides a unified way to deal with both tasks. If s is the true value of the mean signal, then the distribution of $t(s)$ under that hypothesis is a χ^2 density with one degree of freedom, $p(\chi^2|ndf = 1)$. Sometimes, however, it is necessary to consider $t(s)$ when the value of s in the argument differs from the value s, say s_0, which determines the density of $t(s)$. For example, suppose that a model of new physics predicts a mean count s_0 and an analysis is planned to test this model. We may be interested to know, for example, what value of $t(s)$ we might expect for a given amount of data. If $s = 0$, the goal may be to determine the average or median significance with which we may be able to reject the background-only hypothesis. Since the predicted signal s_0 differs from $s = 0$, the density of $t(s, \hat{s})$—where for clarity, the dependence on the estimate $\hat{s}$ is made explicit—will no longer be χ^2, but rather a non-central χ^2 density, $p(\chi^2|ndf = 1, nc)$ with non-centrality parameter nc. An approximate value for the non-centrality parameter is $nc = t(s, s_0)$, that is, it is the test statistic computed using an **Asimov**[10] data set [14] in which the “observed" count N is set equal to the true mean signal count, $s_0 + b$.

3 Bayesian analysis

Bayesian analysis is merely applied probability theory with the following significant twist: a method is Bayesian if

- it is based on the degree of belief interpretation of probability and
- it uses Bayes' theorem

$$p(\theta, \omega|D) = \frac{p(D|\theta, \omega)\, \pi(\theta, \omega)}{p(D)}, \tag{3.38}$$

where

$$\begin{aligned} D =&\ \text{observed data}, \\ \theta =&\ \text{parameters of interest}, \\ \omega =&\ \text{nuisance parameters}, \\ p(D|\theta, \omega) =&\ \text{likelihood}, \\ p(\theta, \omega|D) =&\ \text{posterior density}, \\ \pi(\theta, \omega) =&\ \text{prior density}, \end{aligned}$$

for *all* inferences. The posterior density is the final result of a Bayesian analysis from which, if desired, various summaries can be extracted. The posterior density assigns a weight to every hypothesis about the

[10] The name of this special data set is inspired by the short story *Franchise* by Isaac Asimov describing a futuristic United States in which, rather than having everyone vote in a general election, a single (presumably representative) person is chosen to answer a series of questions whose answers are analyzed by an AI system. The AI system then decides the outcome of the election by determining what would have been the outcome had the general election been held!

values of the parameters of the probability model, which, in addition to the likelihood, also includes a function called the prior density or **prior** for short. The parameters can be discrete, continuous, or both, and nuisance parameters are eliminated by marginalization,

$$\begin{aligned} p(\theta|D) &= \int p(\theta, \omega|D)\, d\omega, \\ &\propto \int p(D|\theta, \omega)\, \pi(\theta, \omega)\, d\omega. \end{aligned} \tag{3.39}$$

The prior $\pi(\theta, \omega)$ encodes whatever assumptions we make and information we have about the parameters θ and ω independently of the data D. A key feature of the Bayesian approach is recursion: the use of the posterior density $p(\theta, \omega|D)$ as the prior in a subsequent analysis. The Bayesian approach also permits an intuitive way to quantify the uncertainty in predictions. Consider the statistical model $p(t|x, \theta)$, where t, x could be random variables and the posterior density $p(\theta|D)$ has been computed with data D. For example, x could be the inputs to a machine learning (ML) model, t the model outputs, θ the model parameters and D the training data. In principle, we can compute the **predictive distribution**

$$p(t|x, D) = \int p(t|x, \theta)\, p(\theta|D)\, d\theta, \tag{3.40}$$

which is a probability distribution over the machine learning model outputs. ML models typically provide only **point estimates**, that is, estimates without a quantitative measure of uncertainty. But to compute the predictive distribution requires finding a feasible way to approximate the high-dimensional integral in Eq. (3.40).

The Bayesian rules are simple, yet they yield an extremely powerful and general inference algorithm. However, high-energy physicists remain wedded to the frequentist approach because of the still widespread perception that the Bayesian algorithm is too subjective to be useful for scientific work. However, there is considerable published evidence to contrary, including in particle physics, witness the successful use of Bayesian analysis in the discovery of single top quark production at the Tevatron [19,20] and searches for new physics at the LHC [21–23].

So, why do high-energy physicists, for the most part, remain skeptical about Bayesian analysis? For many, the Achilles heel of the Bayesian approach is the difficulty of specifying a believable prior over the parameter space of the likelihood function. In our example, to make an inference about the mean event count s using the data $N = 25$ events with a background of $B \pm \delta B = 9.4 \pm 0.5$ events, a prior density $\pi(s, b)$ must be constructed. Even after more than two centuries of effort, discussion, and argument, statisticians have failed to reach a consensus about how to do this in the general case. Nevertheless, Bayesian analysis is widely and successfully used, even within high-energy physics. This strongly suggests that we should refrain from overstating the difficulties. After all, physics is replete with approximations, both of a technical and conceptual nature. The same is true of statistical analysis. But, of course, this is no excuse for sloppy reasoning. Rather it is a reminder not to make perfection the enemy of the good.

The high-energy physicists who have given this topic some thought generally agree with the statisticians who argue that the following invariance property should hold for any prior, at least ideally,

$$\pi_\phi(\phi)d\phi = \pi_\theta(\theta)d\theta, \tag{3.41}$$

where $\phi = f(\theta)$ is a one-to-one mapping of the parameter vector θ, e.g., $\theta = (s, b)$, to the new parameter vector ϕ and π_ϕ and π_θ are, in general, different functions of their arguments. If the above invariance holds, then the posterior density will likewise be reparametrization invariant in the same sense as the prior. Suppose we have a rule for creating a prior $\pi(*)$ and we apply this rule to create the density π_ϕ. The same rule is now used to create π_θ after which we transform from $\pi_\theta(\theta)d\theta$ to $\pi(\phi)d\phi$. Invariance with respect to the choice of parametrization demands that $\pi = \pi_\phi$. It surely ought not to matter whether we parametrize the likelihood $p(D|s, b)$ in terms of s and b or in terms of s and $u = \sqrt{b}$. After all, the likelihood hasn't really changed, therefore, it would be odd if this "non-change" altered the posterior density. Whether or not a change occurs depends on the nature of the prior, as the following example illustrates.

Consider the probability model $p(D|s) = \text{Poisson}(D|s)$, written in two different ways: $p(D|s) = \exp(-s)s^D/D!$ and $p(D|\sigma) = \exp(-\sigma^2)\sigma^{2D}/D!$, where $\sigma = \sqrt{s}$. To compute the posterior densities $p(s|D)$ and $p(\sigma|D)$ priors must be specified. The most widely used rule for doing so is: choose the prior to be flat, that is, uniform: $\pi(s) = 1$ and $\pi(\sigma) = 1$ in the parameter space. For an unbounded parameter space this choice yields $\int \pi(s)\, ds = \int \pi(\sigma)\, d\sigma = \infty$. While this has a bad look it is not necessarily a problem [15]! The posterior density in the s parametrization is $p(s|D) = \exp(-s)s^D/D!$, while it is $p(\sigma|D) = \exp(-\sigma^2)\sigma^{2D}/\Gamma(D + 1/2)$ in the σ parametrization.

We now transform $p(\sigma|D)d\sigma$ to $p'(s|D)ds$. The result is $p'(s|D) = \exp(-s)s^{D-1/2}/\Gamma(D+1/2)$, which differs from $p(s|D)$. But this is not surprising given that the flat prior is not reparametrization invariant. Many regard this as a serious problem, one that worsens as the dimensionality of the parameter space increases. Others point to the numerous successful uses of the flat prior even in problems with high-dimensional parameter spaces, and accept the lack of invariance as a price worth paying to avoid the not inconsiderable effort of constructing an invariant prior.

A general method to create invariant priors was suggested by the geophysicist Sir Harold Jeffreys in the 1930s [18], which in the intervening years has received considerable mathematical validation through many different lines of reasoning (see, for example, [25]). The Jeffreys prior is given by

$$\pi(\theta) = \sqrt{\det I(\theta)}, \tag{3.42}$$

$$\text{where } I_{ij} = E\left[\frac{\partial \ln p(x|\theta)}{\partial\theta_i}\frac{\partial \ln p(x|\theta)}{\partial\theta_j}\right] \tag{3.43}$$

is the **Fisher information matrix** and and where the average is with respect to potential observations x sampled from the density $p(x|\theta)$. For most distributions of interest to physicists, the Fisher information matrix can be written as

$$I_{ij} = -E\left[\frac{\partial^2 \ln p(x|\theta)}{\partial\theta_i\,\partial\theta_j}\right]. \tag{3.44}$$

When the Jeffreys prior is applied to $p(x|\mu, \sigma) = \text{Gaussian}(x, \mu, \sigma)$ it yields

$$\pi(\mu, \sigma) d\mu\, d\sigma = \frac{d\mu\, d\sigma}{\sigma^2}. \tag{3.45}$$

$$\boxed{\textbf{Exercise 9:}\ \text{Show this}}$$

Ironically, the resulting posterior density was rejected by Jeffreys, and subsequently by statisticians because it yielded unsatisfactory inferences! The preferred prior for the Gaussian is

$$\pi(\mu, \sigma) d\mu\, d\sigma = \frac{d\mu\, d\sigma}{\sigma}, \tag{3.46}$$

because it leads to excellent results.

So what is a confused physicist to make of this? One way forward is to reject the Bayesian omelette and stick to the frequentist gruel. The gruel may be thin, but it is at least relatively easy to make. The other way forward is to dismiss the arguments that yield Eq. (3.42) in favor of reasoning that yields Eq. (3.46) (see, for example, [24]). Yet another way forward is to take seriously the many persuasive arguments that lead to Eq. (3.42) and try to understand what the reported failures of the Jeffreys prior for problems involving more than one parameter is telling us. Here is possible path to some understanding. Note that Eq. (3.46) can be written as

$$\begin{aligned}\pi(\mu, \sigma) d\mu\, d\sigma &= \sigma \left[\frac{d\mu\, d\sigma}{\sigma^2}\right], \\ &= \sigma_0 \exp(\ln \sigma / \sigma_0) \left[\frac{d\mu\, d\sigma}{\sigma^2}\right].\end{aligned} \tag{3.47}$$

This suggests, in the spirit of [25], that it is better to interpret the Jeffreys prior as nothing more than an invariant measure on the parameter space of the associated statistical model, one that assigns equal weight to every *probability density* labeled by θ. Assigning equal weight to every probability density is a reparametrization-invariant procedure, while, as we saw above, assigning equal weight to every *parameter* is not. If this interpretation is accepted, then the prior density is actually given by

$$\pi(\theta) = g(\theta) \sqrt{\det I(\theta)}, \tag{3.48}$$

where $g(\theta)$ is a function that could assign non-equal weights to the probability densities, such as the term before the brackets in Eq. (3.47). That term is essentially the exponential of the entropy of the Gaussian density, which assigns a weight $\propto \sigma$ to every density indexed by μ, σ. This is promising. What is missing, however, is a convincing theoretical framework for choosing $g(\theta)$, a challenge that we leave to the reader.

For our example, to keep things simple we shall forego invariance and use a flat prior in both s and b. But before returning to the example, we review hypothesis testing from a Bayesian perspective.

3.1 Model selection

Hypothesis testing (also known as model selection) in Bayesian analysis requires the calculation of an appropriate posterior density or probability, as is true of all fully Bayesian calculations,

$$p(\theta, \omega, H|D) = \frac{p(D|\theta, \omega, H)\, \pi(\theta, \omega, H)}{p(D)}, \tag{3.49}$$

where we have explicitly included the index H to identify the different hypotheses. By marginalizing $p(\theta, \omega, H|D)$ with respect to all parameters except the ones that label the hypotheses or models, H, we arrive at

$$p(H|D) = \int p(\theta, \omega, H|D)\, d\theta\, d\omega, \tag{3.50}$$

that is, the probability of hypothesis H given observed data D. In principle, the parameters ω could also depend on H. For example, suppose that H labels different parton distribution function (PDF) models, say CT14, MMHT, and NNPDF, then ω would depend on the PDF model and should be written as ω_H. Like a Ph.D., it is usually convenient to arrive at the end-point, here the probability $p(H|D)$, in stages.

1. Factorize the prior, e.g.,

$$\begin{aligned}\pi(\theta, \omega_H, H) &= \pi(\theta, \omega_H|H)\, \pi(H), \\ &= \pi(\theta|\omega_H, H)\, \pi(\omega_H|H)\, \pi(H).\end{aligned} \tag{3.51}$$

   In many cases, we can assume that the parameters of interest θ are independent, *a priori*, of both the nuisance parameters ω_H as well as the model label H, in which case we can write, $\pi(\theta, \omega_H, H) = \pi(\theta)\, \pi(\omega_H|H)\, \pi(H)$.

2. Then, for each hypothesis, H, compute the function

$$p(D|H) = \int p(D|\theta, \omega_H, H)\, \pi(\theta, \omega_H|H)\, d\theta\, d\omega_H. \tag{3.52}$$

3. Then, compute the probability of each hypothesis,

$$p(H|D) = \frac{p(D|H)\, \pi(H)}{\sum_H p(D|H)\, \pi(H)}. \tag{3.53}$$

Clearly, to calculate the probabilities $p(H|D)$ it is necessary to specify the priors $\pi(\theta, \omega|H)$ and $\pi(H)$. With some effort, it is possible to arrive at an acceptable form for $\pi(\theta, \omega|H)$, however, it is highly unlikely that consensus could ever be reached on the prior $\pi(H)$. At best we would have to agree on a convention. For example, we could by convention assign equal probabilities to the two hypotheses H_0 and H_1, that is, $\pi(H_0) = \pi(H_1) = 0.5$. But do we really believe that the Standard Model and the MSSM are equally probable models?

One way to sidestep the polemics of assigning $\pi(H)$ is to *compare* probabilities,

$$\frac{p(H_1|D)}{p(H_0|D)} = \left[\frac{p(D|H_1)}{p(D|H_0}\right] \frac{\pi(H_1)}{\pi(H_0)}, \tag{3.54}$$

but use only the term in brackets, called the global **Bayes factor**, B_{10}, as a way to compare hypotheses. The Bayes factor is the factor by which the relative probabilities of two hypotheses *changes* as a result of incorporating the data, D. The word global indicates that we have marginalized over all the parameters of the two models. The *local* Bayes factor, $B_{10}(\theta)$ is defined by

$$B_{10}(\theta) = \frac{p(D|\theta, H_1)}{p(D|H_0)}, \tag{3.55}$$

where,

$$p(D|\theta, H_1) \equiv \int p(D|\theta, \omega_{H_1}, H_1)\, \pi(\omega_{H_1}|H_1)\, d\omega_{H_1}, \tag{3.56}$$

are the **marginal** or integrated likelihoods in which we have assumed the *a priori* independence of θ and ω_{H_1}. We have further assumed that the marginal likelihood that depends on H_0 is independent of θ, which is a very common situation. For example, θ could be the expected signal count s, while $\omega_{H_1} = \omega$ could be the expected background b. In this case, the hypothesis H_0 is a special case of H_1, namely, it is the same as H_1 with $s = 0$. An hypothesis that is a special case of another is said to be **nested** within the more general hypothesis. All this will become clearer when we work through the Bayesian analysis of the 4-lepton data.

There is a notational subtlety that may be missed: because of the way we have defined $p(D|\theta, H)$, we need to multiply $p(D|\theta, H)$ by the prior $\pi(\theta)$ and then integrate with respect to θ to calculate $p(D|H)$.

3.2 Bayesian analysis of 4-lepton data

In this section, as a way to illustrate a Bayesian analysis, we

1. compute the posterior density $p(s|D)$,
2. compute a 68% credible interval $[l(D), u(D)]$, and
3. compute the global Bayes factor $B_{10} = p(D|H_1)/p(D|H_0)$,

associated with the 4-lepton data.

Statistical model

The likelihood is the same as that used in the frequentist analysis, namely, Eq. (2.27). But in a Bayesian analysis the likelihood is only part of the model; we also need a prior $\pi(s, b)$ that encodes what we *know*, or *assume*, about the mean background and signal independently of the observations D. How exactly that should be done remains an active area of debate and research. Below, we take the easy way out!

One point that should be noted is that the prior $\pi(s, b)$ can be factorized in two ways,

$$\begin{aligned} \pi(s, b) &= \pi(s|b)\, \pi(b), \\ &= \pi(b|s)\, \pi(s). \end{aligned} \tag{3.57}$$

It is worth noting because $\pi(s, b)$ is routinely written as $\pi(s, b) = \pi(s)\pi(b)$, which is not true, in general. The *a priori* independence of s and b is an assumption, one that we shall make. What do we know about s and b? We know that s and b are ≥ 0. We also know the probability model and how s and b enter it.

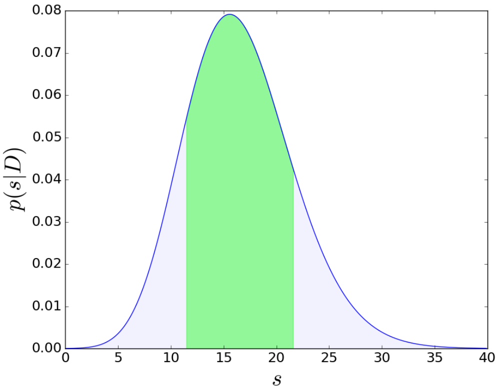

Fig. 10: Posterior density for 4-lepton data. The shaded area is the 68% central credible interval.

Given this information, there are well-founded methods to construct $\pi(s, b)$. However, for simplicity for b we shall use the improper prior $\pi(b) = k$, where k is the scale factor in the likelihood $p(D|s, b)$, and either the improper prior $\pi(s) = 1$, or the proper prior $\pi(s) = \delta(s - 15.6)$. An improper prior is one that integrates to infinity, which as noted above is not necessarily problematic [15].

Marginal likelihood

Having completed the probability model, the rest of the Bayesian analysis proceeds in a routine manner. First it is convenient to eliminate the nuisance parameter b, using the improper prior $\pi(b) = k$,

$$\begin{aligned} p(D|s, H_1) &= \int_0^\infty p(D|s, b)\, \pi(b)\, db, \\ &= \frac{1}{M}(1 - x)^2 \sum_{r=0}^{N} \text{Beta}(x, r + 1, M)\, \text{Poisson}(N - r|, s), \end{aligned} \tag{3.58}$$

where $x = 1/(1 + k)$,

$\boxed{\textbf{Exercise 10: } \text{Show this}}$

and thereby arrive at the marginal likelihood $p(D|s, H_1)$. The symbol H_1 has been introduced to represent the hypothesis that the signal is non-zero.

Posterior density

Given the marginal likelihood $p(D|s, H_1)$ and $\pi(s)$ we can compute the posterior density,

$$p(s|D, H_1) = p(D|s, H_1)\, \pi(s)/p(D|H_1), \tag{3.59}$$

where,

$$p(D|H_1) = \int_0^\infty p(D|s, H_1)\, \pi(s)\, ds.$$

Setting $\pi(s) = 1$ yields,

$$p(s|D, H_1) = \frac{\sum_{r=0}^{N} \text{Beta}(x, r+1, M)\,\text{Poisson}(N-r|s)}{\sum_{r=0}^{N} \text{Beta}(x, r+1, M)}. \tag{3.60}$$

Exercise 11: Derive an expression for $p(s|D, H_1)$ assuming $\pi(s) = \text{Gamma}(qs, 1, U+1)$ where q and U are known constants.

The posterior density $p(s|D, H_1)$ completes the inference about the mean signal s. In principle we could stop there, but in practice summaries of the posterior density are furnished, such as a **credible interval**, which is the Bayesian analog of a confidence interval. Like confidence intervals credible intervals, $[l(D), u(D)]$, at credible level p defined by

$$\int_{l(D)}^{u(D)} p(s|D, H_1)\, ds = p \tag{3.61}$$

are not unique. The analog of Neyman's central interval is the central credible interval defined by

$$\begin{aligned} \int_{0}^{l(D)} p(s|D, H_1)\, ds &= (1-p)/2, \\ \int_{u(D)}^{\infty} p(s|D, H_1)\, ds &= (1-p)/2. \end{aligned} \tag{3.62}$$

For the 4-lepton data this leads to the central credible interval $[11.5, 21.7]$ for s with $p = 0.683$, which is shown in Fig. 10. The statement $s \in [11.5, 21.7]$ at 68% CL means there is a 68% probability that s lies in the specified interval. Unlike the analogous frequentist statement this one is about this particular interval and the 68% is a degree of belief, not a relative frequency. Statements of this form do, of course, have a coverage probability. However, *a priori*, there is no reason why the coverage probability of credible intervals should satisfy the frequentist principle. In practice, it is found that credible intervals with appropriately chosen priors can moonlight as approximate confidence intervals. But when this happens it does not mean that their interpretations somehow are the same, it simply means that a misinterpretation of the intervals is likely to be benign.

Bayes factor

We noted above that

$$p(D|H_1) = \int_{0}^{\infty} p(D|s, H_1)\, \pi(s)\, ds.$$

Furthermore, $p(D|H_1) < \infty$ even with the improper prior $\pi(s) = 1$. However, another *arbitrary* constant besides unity could have been chosen, for example, $\pi(s) = C$. That constant would not have altered the posterior density $p(s|D, H_1)$ and therefore choosing $C = 1$ as a matter of convenience was fine. However, here we wish to compute the global Bayes factor $B_{10} = p(D|H_1) \,/\, p(D|H_0)$. The background-only hypothesis, H_0, is nested in H_1 and has marginal likelihood $p(D|H_0) \equiv p(D|0, H_1)$. Since the constant k in the background prior $\pi(b) = k$ scales both $p(D|H_1)$ and $p(D|H_0)$ the constant cancels and no issue arises from using an improper background prior. However, for H_1 $\pi(s) = C$ and the parameter s appears only in the calculation of $p(D|H_1)$. Therefore, the Bayes factor is scaled by the arbitrary constant C. Consequently, the Bayes factor can be assigned any value merely by choosing an appropriate value for C. This is clearly unsatisfactory. The upshot is that while improper priors may yield reasonable results for the posterior density $p(s|D, H_1)$, albeit ones that are not reparametrization invariant unless the priors are chosen carefully, that is not the case for Bayes factors. To arrive at a satisfactory Bayes factor, a proper prior *must* be used. The simplest such prior is $\pi(s) = \delta(s - \hat{s})$, where $\hat{s} = N - B = 15.6$ events. With this prior, the Bayes factor is

$$B_{10} = \frac{p(D|H_1)}{p(D|H_0)} = 4967.$$

We conclude that the 4-lepton observations increase the probability of hypothesis $s = 15.6$ events relative to the probability of the hypothesis $s = 0$ by ≈ 5000. Large numbers can be avoided if we map the Bayes factor to a measure akin to the frequentist "n-sigma",

$$Z = \sqrt{2 \ln B_{10}}, \tag{3.63}$$

which gives $Z = 4.13$.

The Bayesian and frequentist results are approximately the same, which is typically the case when the data are sufficient. This is because the influence of the prior is smaller than when the data are sparse.

This brings to a close our discussions of the frequentist and Bayesian approaches to statistical analysis. We conclude these lecture notes with a brief look at machine learning, which is now widely used in many domains.

4 Introduction to supervised machine learning

For millennia visionaries have dreamed of creating artificial beings exhibiting human and superhuman characteristics. In 1950, the great English mathematician Alan Turing whose genius helped save millions of lives during the World War II proposed an operational definition of an intelligent agent, a test now known as the *Turing test* [26]. The test cuts to the chase: if it is impossible for you to tell whether you are conversing with a person or a machine and it turns out that you are in fact conversing with a machine then the latter is intelligent. In the decades following the publication of the Turing test progress towards creating such agents was slow in part because the required conceptual breakthroughs were lacking and in part because the available computing power was severely limited.

During the past two decades algorithmic breakthroughs and the exponential growth in the size of data sets and computing power has caused the field of artificial intelligence (AI)—powered by *machine learning* (ML), that is, computer-based algorithms to construct useful models of data—to go from research lab to impressive commercial applications. There are many things humans do that seem far beyond current machine learning capabilities. It is still the case that we are unable to replicate a young child's ability to intuit the fact that the noises she hears from the people around her have meaning. Nor can we replicate the extraordinary human ability to be "trained" on a relatively small number of instances of, say, pictures of the Golden Gate bridge and yet be able to identify the Golden Gate in other pictures of the bridge taken from perspectives that may never have been seen before. Nevertheless, impressive progress has been made recently. A notable breakthrough was made by the Google subsidiary *DeepMind* in creating an agent that taught itself to play to superhuman levels the ancient Chinese game of Go, as well as Chess and Shogi (Japanese chess) *tabula rasa*. These self-teaching feats were achieved in a mere 24 hours [27]! And then there is ChatGPT, the chatbot that took the world by storm in 2023.

Our purpose here is considerably more modest; it is to emphasize something that can get lost in the hype, namely, that most contemporary AI systems are, for the most part, large highly non-linear functions that are capable of mapping from one space to another, where large refers to the parameter space of these functions. For example, ChatGPT uses a mathematical function called a large-language model (LLM) with 175 billion parameters. This function, in a way that is far from clear, has encoded everything useful or interesting that is on the web. ChatGPT leverages this vast encoded knowledge to generate new text following prompts from the user. In many cases, including for ChatGPT, these mathematical functions approximate probabilities. The breakthrough has been the ability to fit these enormously complicated functions on timescales that are practical. Since our scope is relatively modest and we wish to emphasize key ideas that span many classes of machine learning models, we avoid unnecessary complications by considering a simplified version of the following problem: separating Higgs boson events in which the Higgs boson is produced via vector boson fusion (VBF) from events in which the Higgs boson is created via gluon gluon fusion (ggF). But to give a taste of the state-of-the-art, we end with a brief qualitative description of the transformer, the machine learning model that powers ChatGPT and similar chatbots. First with discuss a few key ideas of machine learning.

Most machine learning algorithms fall into five broad categories:

1. supervised learning,
2. semi-supervised learning,
3. unsupervised learning (i.e., pattern detection)
4. reinforcement learning, and
5. generative learning.

The simplest category of algorithm is supervised learning in which the data for fitting models, i.e., *training* them, consist of labeled objects. If the labels define the class to which objects belong, for example, 0, for gluon gluon fusion events and $+1$ for vector boson fusion events, then as shown below the resulting function will be a *classifier*. If the labels form a continuous set, then the resulting function will be a *regression function* (sometimes called a "regressor"). For example, suppose the objects are jets characterized by their transverse momentum p_T and pseudo-rapidity η and possibly other detailed characteristics such as the electromagnetic fraction, while the labels are the true jet transverse momenta. The regressor will be a correction function that maps the jet characteristics to an approximation of the true jet p_T. Our example will be a simple VBF/ggF classifier.

4.1 A bird's eye view of supervised machine learning

Supervised machine learning can be construed as a game in which winning means picking the best function (or functions) from a function space. The game includes three elements:

1. a function space $\mathcal{F} = \{f(x, w)\}$ containing parametrized functions $f(x, w)$, where x are object characteristics—**features** in machine learning jargon—and w are the parameters;

2. a loss function $L(t, f)$, which measures the cost of making a bad function choice, and where t are labels associated with the features x, and
3. a constraint $C(w)$ that places some restriction on the choices of functions.

The best function $f(x, w*)$ is found by minimizing the constrained **empirical risk**,

$$R(w) = \frac{1}{K}\sum_{i=1}^{K} L(t_i, f_i) + C(w), \text{ where } f_i = f(x_i, w), \tag{4.64}$$

with respect to the choice of function f, which in practice means with respect to the parameters w.

4.1.1 Minimization via gradient descent

A loss function, through the empirical risk, defines a high-dimensional "landscape" in the space of model parameters, or equivalently in the space of functions. The goal is to find the lowest point in that landscape through repeated application of the algorithm

$$w \leftarrow w - \eta\nabla R, \tag{4.65}$$

where η is the so-called learning rate and ∇R is the gradient of the empirical risk. Why does the update algorithm in Eq. (4.65) reduce R? Consider the value of the empirical risk $R(w - \eta\nabla R)$ at the updated parameter point $w - \eta\nabla R$. The function $R(w - \eta\nabla R)$ can be expanded as follows

$$R(w - \eta\nabla R) = R(w) - \eta\nabla R \cdot \nabla R + \mathcal{O}(\eta^2). \tag{4.66}$$

Equation (4.66) shows that $R(w - \eta\nabla R) < R(w)$ provided that η is small enough.

Used as-is the algorithm in Eq. (4.65) would fail miserably because of the complexity of the landscape defined by $R(w)$ and the possibility that the minimizer could get stuck in a bad local minimum or diverge away from the minimum because of the instability caused by a saddle point. To alleviate this problem the standard approach is to replace the exact gradient ∇R by a *noisy* estimate of it at any given point. This is usually achieved by replacing R by an approximation that uses a small subset—that is, **batch**—of the training data in the sum that defines R. Typically, a new batch is used at every step of the minimization algorithm. This minimization algorithm is called **stochastic gradient descent**, of which there are many variants. The addition of noise increases the chance that the minimizer will escape from an unfavorable location in the parameter space and move towards a better minimum.

4.1.2 Minimizing the risk functional

It is intuitively clear that a successful minimization of the empirical risk, Eq. (4.64), will yield a solution $f(x, w*)$ that is as close as possible to the labels, or **targets**, t. But in mathematics, as in physics, we can often gain a clearer understanding of a construct by taking a suitable limit of it. To that end consider the limit of $R(w)$, that is, the empirical risk (aka the *average loss*) as $K \to \infty$. In that limit and assuming the effect of the constraint goes to zero the empirical risk becomes the **risk functional**

$$\begin{aligned} R[f] &= \int dx \int dt\, L(t, f)\, p(t, x), \\ &= \int dx\, p(x) \left[\int dt\, L(t, f)\, p(t|x) \right], \end{aligned} \tag{4.67}$$

where we have used $p(t|x) = p(t, x)/p(x)$. The function $p(t, x)$ is the (typically unknown) joint probability density of the data (t, x). Whether the features x represent an event, a jet, an image, or piece of writing and t represents known data about each instance of x all the information about the mapping from x to t is contained in the joint probability density $p(t, x)$. This is an important point because the failures of machine learning are almost always due to an object with known characteristics x' but unknown label t' not being a member of the population $\{(t, x)\}$ that defines $p(t, x)$. If an agent is trained on a million images of dogs and cats it is not surprising that it will classify a horse as either a dog or a cat because the probability $p(t, x)dt\, dx$ does not encompass images of horses. The point is that the function $f(x, w)$ will do what it is designed to do. Which prompts the question: what exactly is $f(x, w)$ designed to do?

To answer this question we need to minimize the risk functional, Eq. (4.67), by setting its functional derivative $\delta R/\delta f$ to zero for all values of x, if possible. This will be possible if the function f is sufficiently flexible. If this is the case then $\delta R/\delta f = 0$ leads to the very important result

$$\boxed{\int \frac{\partial L}{\partial f}\, p(t|x)\, dt = 0.} \tag{4.68}$$

From the above we conclude that 1) with sufficient training data, 2) a sufficiently flexible function f, and 3) a minimizer capable of finding the global minimum the quantity approximated by the function f depends solely on the form of the loss function $L(t, f)$ and the conditional density $p(t|x)$ of the training data. Equation (4.68) is a general result that does *not* depend on the nature or form of the function f;f does not have to be a neural network. The reason neural network models have become so popular is because they have proven to be highly flexible functions.

4.1.3 Loss functions

Let us apply Eq. (4.68) to the widely used **quadratic loss**,

$$L(t, f) = (t - f)^2. \tag{4.69}$$

From $\partial L/\partial f = -2(t - f)$ and noting that $\int p(t|x)\, dt = 1$ we find

$$\boxed{f(x, w*) = \int t\, p(t|x)\, dt}, \tag{4.70}$$

where $w*$ is the best-fit value of the parameter vector w. Equation (4.70) is an important result because it tells us precisely what the function $f(x, w*)$ approximates. If one uses the quadratic loss then the function $f(x, w*)$ approximates the conditional average of the targets. This result was first derived in the context of neural networks [28–30], however, as noted it is not restricted to this (admittedly large) class of functions.

The quadratic loss is often used in regression problems. But it may not always be appropriate. Consider the task of approximating the mapping $f : \mathbb{R}^d \to \mathbb{U} \in \mathbb{R}$, with $\mathbb{U}$ a compact subset of $\mathbb{R}$, say the unit interval. Equation (4.70) informs us that we should not be surprised to find an upward bias in regression values near $t = 0$ and a downward bias near $t = 1$. Another point worth noting is that if we minimize the average quadratic loss using training data in which one class of objects is labeled with target $t = 0$ and the other with target $t = 1$, the function $f(x, w*)$ will approximate the probability that the object with features x belongs to the class labeled with $t = 1$; that is, $f(x, w*)$ will be a classifier that approximates the class probability $p(1|x)$, which from Bayes' theorem can be written as

$$p(1|x) = \frac{p(x|1)\,p(1)}{p(x|1)\,p(1) + p(x|0)\,p(0)}, \tag{4.71}$$

where $p(1)$ and $p(0)$ are the prior probabilities associated with the two classes. If this were indeed a classification problem one typically trains with a **balanced data set** for which $p(1) = p(0)$. In this case $p(1|x)$ is referred to as a **discriminant**, $D(x)$, given by

$$D(x) = \frac{p(x|1)}{p(x|1) + p(x|0)}. \tag{4.72}$$

Notice that $p(1|x)$ and $D(x)$ are related:

$$p(1|x) = \frac{D(x)}{D(x) + [1 - D(x)]/[p(1)/p(0)]}. \tag{4.73}$$

This is useful because sometimes we need to model Eq. (4.71) with $p(0) \neq p(1)$. For example, in a signal/background discrimination task $p(1)/p(0)$ would be the expected prior signal to background ratio. If that ratio is very far from unity it would be very difficult to model Eq. (4.72) directly. Suppose that the ratio was 10^{-3} and the training sample size was 10^5 events. This sample would contain only 100 signal events out of 100,000. Almost any numerical algorithm to construct a classifier would struggle with such an unbalanced data set. Equation (4.73) shows, however, that it not necessary to use an unbalanced data set to model $p(1|x)$. We can use a balanced data set to approximate $D(x)$ and use Eq. (4.73) to map $D(X)$ to the correct class probability $p(1|x)$. This discussion illustrates the utility of understanding what the function $f(x, w)$ approximates.

In practice, for binary classification the preferred loss function is the **binary cross-entropy loss**,

$$L(t, f) = -\begin{cases} \log(f) & \text{if } t = 1 \\ \log(1 - f) & \text{if } t = 0, \end{cases} \tag{4.74}$$

which for algebraic purposes can be conveniently written in terms of Dirac delta functions as

$$L(t, f) = -\log(f)\delta(t - 1) - \log(1 - f)\delta(t). \tag{4.75}$$

This loss function is less sensitive to outliers than the quadratic loss, but yields the same result as

Eq. (4.71). The binary cross entropy loss is a special case of the cross entropy loss

$$L(t, f) = -\sum_{k=1}^{K} \log(f_k), \quad \sum_{k=1}^{K} f_k = 1, \tag{4.76}$$

for a function f_k with K outputs that sum to unity with each output associated with a different class.

Boosted Decision Trees

Boosted decision trees (BDT) [31] are a popular machine learning method in particle physics; and for good reason. They perform well, they are faster to train than neural networks, they are insensitive to poorly performing variables, and they are resistant to over-fitting. In view of their widespread use it is worth taking the time to understand exactly what this machine learning model entails. We shall highlight key features of BDTs using a simple example in which we seek to separate Higgs boson events produced via vector boson fusion (VBF) from gluon gluon fusion (ggF) produced events. In this section, we first discuss decision trees (DT) and then the notion of boosting, that is, enhancing the performance of a machine learning model by averaging over many models.

A decision tree is a nested sequence of *if then else* statements, which can also be viewed as a histogram whose bins are created recursively through **binary partitioning**. The VBF/ggF example uses two discriminating variables (features) $|\Delta\eta|_{jj}$ and m_{jj}, the absolute pseudo-rapidity difference between the two most forward (i.e., largest rapidity) jets in the event and the associated di-jet mass, respectively. Fig. 11 shows two representations of a decision tree for our VBF/ggF discrimination example.

At face value, decision trees do not seem to fit into the mathematical ideas about loss functions discussed above. In particular, it is far from clear what loss function, if any, is being minimized when a decision tree is grown. However, all successful uses of decision trees entail averaging over many of them. As we shall see, it is the averaging that provides the connection to a loss function. Averaging also mitigates a serious problem with decision trees, namely, their instability. Even minor changes to the training data can radically alter the structure of a tree.

The first successful averaging algorithm, called `AdaBoost`, was published by AT&T researchers Freund and Schapire in 1997 [32] who showed that it was possible to create high performance classifiers by averaging ones (called **weak learners**) that perform only marginally better than classification via a coin toss. The algorithm builds a classifier using training data labeled by the discrete labels $t = -1$ or $t = +1$. In the VBF/ggF example below, $t = -1$ is assigned to ggF events and $+1$ is assigned to VBF events. The algorithm, for N training events and K decision trees, proceeds as follows:

1. **initialize** event weights $\omega_{1,n} = 1/N$, $n = 1, \cdots, N$
2. **repeat for** $k \in 1, \cdots K$
    (a) fit a tree $f_k(x)$ that returns either -1 or $+1$, using the current event weights $\{w_{k,n}\}$
    (b) compute error rate $\epsilon_k = \sum_{n=1}^{N} \omega_{k,n} \mathbb{I}[-t_n f_k(x_n)], \mathbb{I}(z) = 1$ if $z > 1$, 0 otherwise
    (c) compute coefficient $\alpha_k = \frac{1}{2} \ln[(1 - \epsilon_k)/\epsilon_k]$
    (d) update weights $w_{k+1,n} = w_{k,n} \exp(-\alpha_k t_n f_k(x_n))/Z_k$,
    where $Z_k = \sum_{n=1}^{N} \omega_{k,n} \exp(-\alpha_k t_n f_k(x_n))$

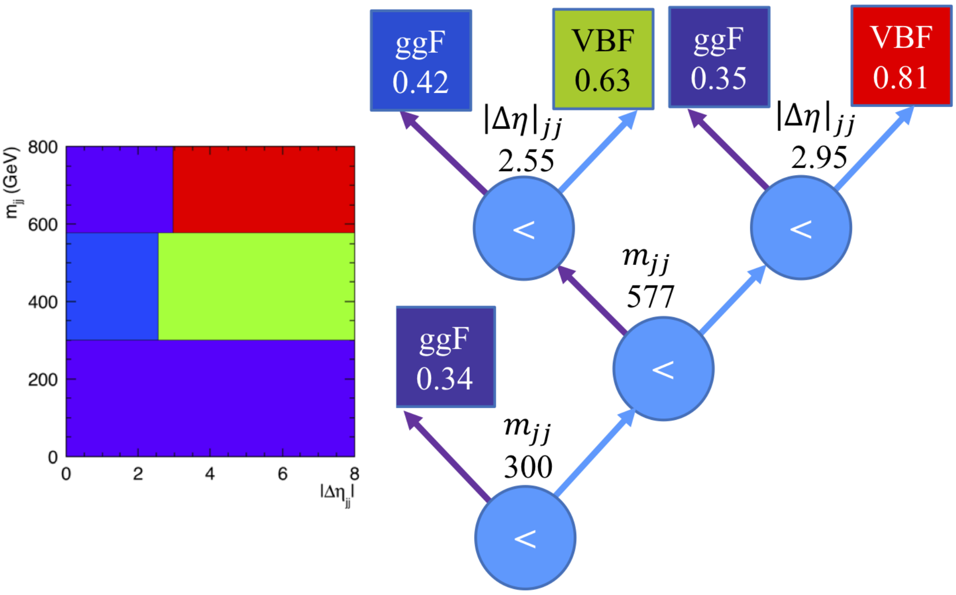

Fig. 11: Two representations of a decision tree to separate VBF from ggF events based on the variables $|\Delta\eta|_{jj}$ and m_{jj}. On the right, the decision tree is represented as a branching structure in which the circles, called **nodes**, represent *if then else* decisions, that is, *binary* decisions. The boxes terminate the tree and are referred to, appropriately, as **leaves**. On the left, the decision tree is represented as a 2D histogram in which the bins, which correspond to the leaves, have been defined by recursive binary partitioning. The bin boundaries, that is, the binary partitions, correspond to the decisions. At a given node, the left branch is taken if $x < x_{\text{cut}}$ otherwise the right branch is taken; x_{cut} is an optimal cut on the variable $x \in \{|\Delta\eta|_{jj}, m_{jj}\}$. The numbers within the leaves are the VBF purity $p = S/(S+B)$, where S and B are the VBF and ggF event counts in a given bin, that is, leaf.

3. classifier $f(x) = \sum_{k=1}^{K} \alpha_k \, f_k(x)$

In step 2(d) the weight of incorrectly classified events, that is events for which $t_n f_k(x_n) = -1$, is *increased*, while that of correctly classified events, for which $t_n f_k(x_n) = +1$, is *decreased*.

AdaBoost is a cryptic algorithm. However, a few years after its publication Friedman, Hastie, and Tibshirani [33] showed that AdaBoost can be viewed as a smart way to minimize the risk functional,

$$E[f] = \int dx \int dt \exp[-tf(x)] \, p(t, x), \tag{4.77}$$

whose minimum occurs at

$$f(x) = \frac{1}{2} \ln \frac{p(t = +1|x)}{p(t = -1|x)}. \tag{4.78}$$

Therefore, despite appearances boosted decision trees fit into the mathematical framework sketched above. Moreover, while the boosted decision tree $f(x)$ cannot be interpreted as a probability it can be mapped to a probability by inverting Eq. (4.78),

$$p(t = +1|x) = \frac{1}{1 + \exp(-2f(x))}. \tag{4.79}$$

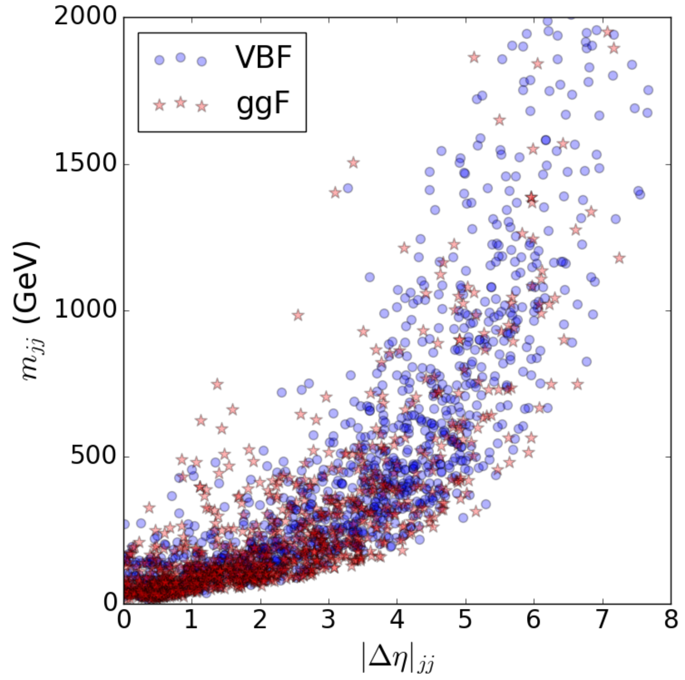

Fig. 12: Simulated distributions of the discriminating variables $(|\Delta\eta_{jj}|, m_{jj})$ for VBF and ggF events. As expected, there is a larger rapidity gap between the jets inVBF events than those in ggF, which arise from gluon radiation.

Below we illustrate the use of the `AdaBoost` algorithm using the Toolkit for Multivariate Analysis `TMVA` [34], which is released with the `ROOT` [35] package from CERN. Note, in the `TMVA` implementation, α_k is defined omitting the factor of 1/2, therefore, in order to convert the un-normalized BDT, $f(x)$, in `TMVA` to a probability the appropriate mapping is

$$p(t = +1|x) = \frac{1}{1 + \exp(-f(x))} \quad (\texttt{TMVA}). \tag{4.80}$$

VBF/ggF discrimination

In this example, a BDT is trained using the `AdaBoost` algorithm in `TMVA` to discriminate between events in which the Higgs boson is created via vector boson fusion (VBF) and events in which the Higgs boson is created via gluon gluon fusion (ggF). The key difference between VBF events and ggF events is that the former features a pair of forward (i.e., large rapidity) jets that is absent from the latter. It is found that the two most discriminating variables between these two classes of events are the absolute pseudo-rapidity difference $|\Delta\eta|_{jj}$ between the two jets and the associated di-jet mass m_{jj} and. The predicted distributions of the two variables is shown in Fig. 12.

We use a training sample size of $N = 20,000$ events, split equally between VBF and ggF events with assigned targets of $t = +1$ and $t = -1$, respectively. The `TMVA` training parameters are `BoostType=AdaBoost`, `NTrees=800`—the number of trees K, `nEventsMin=100`—the minimum number of events per bin, and `nCuts=50`—the number of binary partitions per variable to search for the optimal partition, i.e., cut. The optimal cut is the one which gives the greatest *decrease* in impurity as

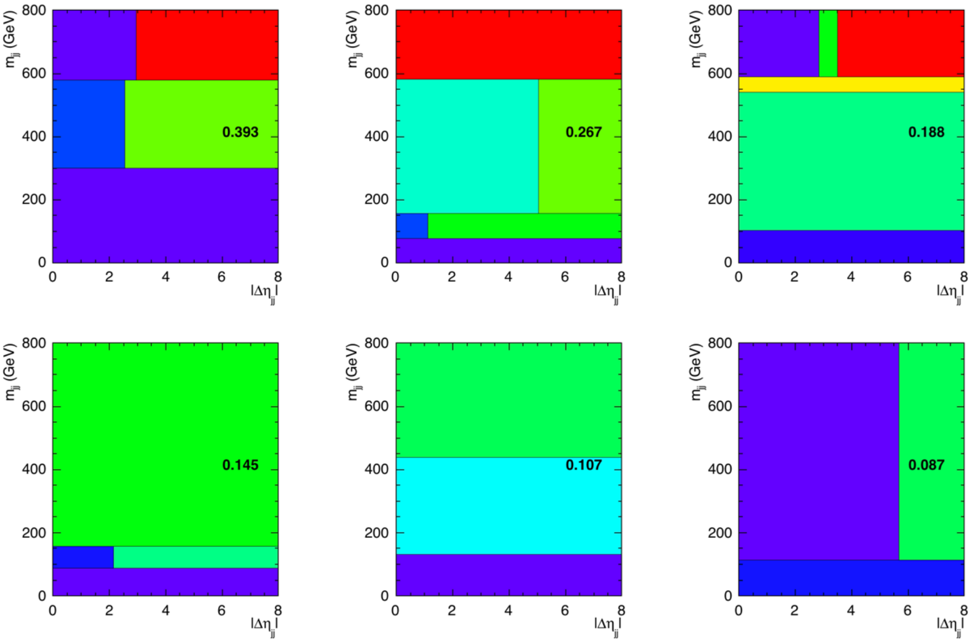

Fig. 13: The first six of the 800 decision trees, displayed as 2D histograms, showing the coefficients $\alpha_1, \cdots, \alpha_6$ associated with the threes.

measured by the Gini index[11], defined by $p(1-p)$ where $p = S/(S+B)$ is the purity and S and B are the signal and background counts, respectively, in a given bin. A bin is maximally pure, either pure signal or pure background, when the Gini index is zero.

Fig. 13 shows the first six decision trees as histograms, each with its associated coefficient $\alpha_k = \ln[(1-\epsilon_k)/\epsilon_k]$[12] printed on the histogram. A decision tree is a piecewise constant function in which each bin (i.e., leaf) is assigned a value. In the `AdaBoost` algorithm the values are $t = \pm 1$; in our example, $t = -1$ for bins in which $B > S$ (i.e., ggF bins) and $+1$ for bins in which $S > B$ (i.e., VBF bins). A given feature vector $x = |\Delta\eta|_{jj}, m_{jj}$, characterizing an event, will fall in a bin in each of the six decision trees of Fig. (13) and the BDT is equal to the average $\sum_{k=1}^{6} \alpha_k f_k(x)$ where each tree $f_k(x)$ returns either $+1$ or -1 depending on the bin in which x falls. In other words a BDT is an un-normalized weighted average over histograms each with a different set of bins. While the piece-wise constant nature remains, the more histograms (that is, trees) that are averaged the smoother one expects the BDT output to become. This is illustrated in Fig. 14, which shows the effect of averaging over an increasing number of trees. Finally, Figs. 15 and 16 show the distribution of the BDT in which the output has been mapped to the probability $p(\text{VBF} \,|\, x) \equiv p(t = +1 \,|\, x) = 1/[1 + \exp(-BDT(x))]$, and the receiver operating characteristic (ROC) curve of the BDT. The ROC curve, and the area underneath it (AUC), are often used as simple measures of the performance of a binary classifier. The larger the AUC the better the performance of the classifier.

Several general-purpose toolkits exist today that feature a wide range of machine learning models including the excellent toolkit `scikit-learn` [36][13] and the research-level toolkit `pytorch` [37], which

[11] After Italian statistician Corrado Gini, 1884-1965.

[12] As noted, `TMVA` omits the factor of 1/2.

[13] `AdaBoost` is available in the Python module `scikit-learn`. But in version 1.4.2 of `scikit-learn` the value returned

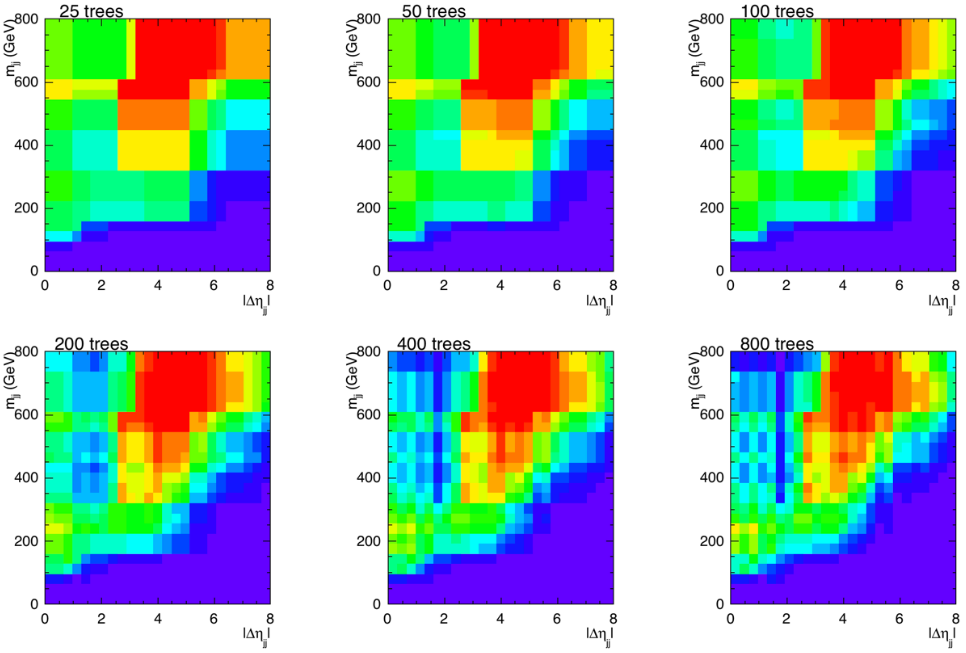

Fig. 14: The outputs of boosted decision trees averaged over differing numbers of decision trees, 25, 50,...,800. Each $BDT(x)$, with $x = |\Delta\eta|_{jj}, m_{jj}$, is mapped to the probability $p(y = +1 \,|\, x) = 1/[1 + \exp(-BDT(x))]$.

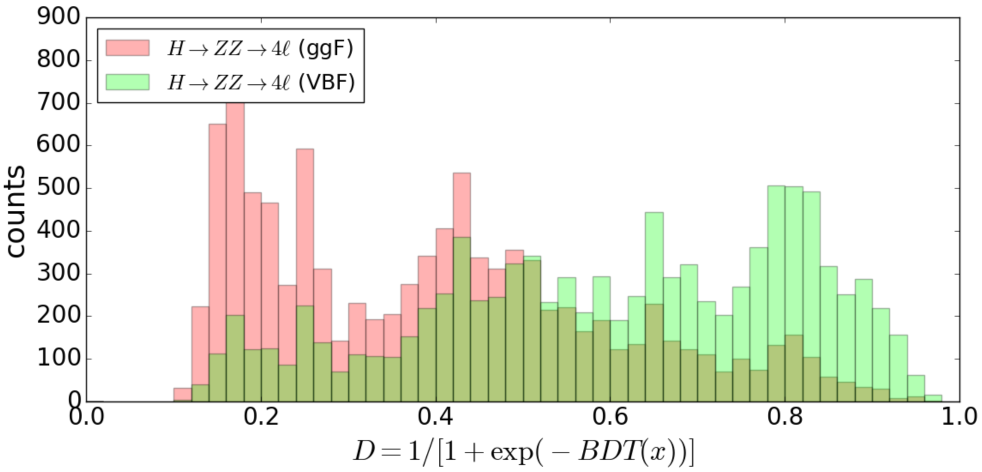

Fig. 15: The distributions of the discriminant $D(x) = 1/[1 + \exp(-BDT(x))]$, where $BDT(x)$ is a boosted decision tree with $K = 800$ trees.

makes it possible to build arbitrarily sophisticated machine learning models including models such as the `transformer`, to which we now turn.

by the `decision_function` method of the `AdaBoostClassifier` differs from that returned by step 3 of the `AdaBoost` algorithm. The `decision_function` returns $2\sum_{k=1}^{K} \alpha_k f_k(x) / \sum_{j=1}^{K} \alpha_j$, while the `predict_proba` method returns $1/[1 + \exp(-f(x))]$.

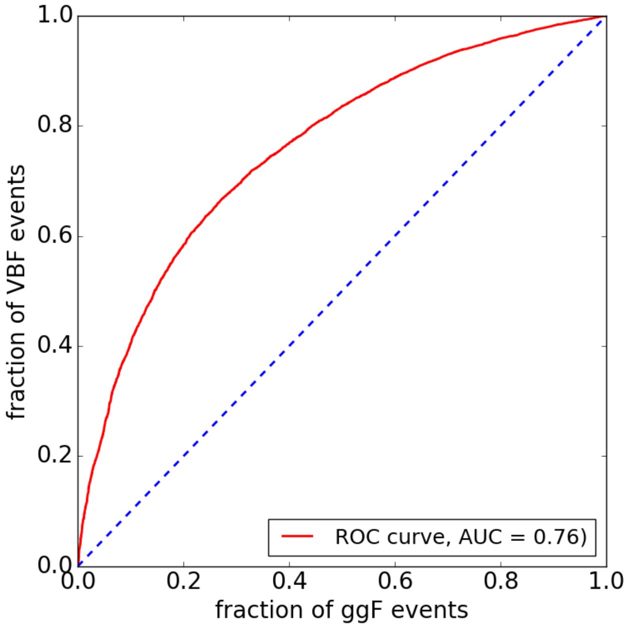

Fig. 16: Receiver operating characteristic (ROC) curve. The area under the curve (AUC) is a commonly used global measure of the discrimination power of a classifier.

4.2 Transformers

To give a sense of the impressive advances that have occurred recently in artificial intelligence/machine learning, we end with a qualitative description of the model that underpins the chatbot `ChatGPT` [38]. The latter uses a machine learning model called the `transformer` [39], which translates one sequence of tokens to another, where tokens can be, for example, the words or parts of words of natural languages or even mathematical symbols. The key features of transformers that makes these models extraordinary is that 1) they work on all the tokens of a sequence in parallel and 2) they are very good at encoding relationships between tokens.

A collection of sequences defines a **vocabulary** of tokens from which all sequences can be formed. For example, suppose the task is to build a model that outputs the Taylor series expansion to 5th order of a mathematical expression built from elementary functions, e.g, $\exp(-ax)[\exp(3ax) - \sin(cx)/\sinh(gx)]$. The vocabulary would presumably include the tokens x, a, c, g as well as sin, sinh, and exp, coded as integers. In the case of `ChatGPT`, the vocabulary for English is thought to be of order 50,000 tokens.

The transformer model uses one or more vocabularies and neural networks that consist of

- **embedding layers** that embed the tokens and their relative positions within sequences as points in a vector space;
- **transformer** layers that implement the syntactic and semantic encoding of the sequences, and
- the **output layer** that computes weights, one for every possible token in the output vocabulary, which can be converted to probabilistic predictions for the next token in the output sequence given the input sequence and the current predicted output sequence.

4.2.1 Sequence to sequence model

A transformer, which is an example of a sequence-to-sequence (seq2seq) model, comprises an **encoder** and a **decoder**. The encoder embeds every token in the source (that is, input) sequence $\boldsymbol{x}$ in a vector space and then processes these vectors through a chain of algorithms called **attention**. The transformed vectors together with the current target sequence $\boldsymbol{t}$ or current predicted output sequence $\boldsymbol{y}$ are sent to the decoder, which embeds the targets in the same vector space. The target vectors are likewise processed through a chain of attention algorithms, while the target vectors and those from the encoder are processed with another attention algorithm. The decoder assigns a weight to every token in the target vocabulary and these weights are used to choose the next output token.

The transformer model is **autoregressive**: the predicted token is appended to the existing predicted output sequence and the model is called again with the same source sequence and the updated predicted output sequence. The procedure repeats until either the maximum output sequence length is reached or an end-of-sequence (EOS) token is predicted as the next token.

4.2.2 Attention

When we translate from one sequence of tokens to another sequence of tokens, for example from one natural language to another, the meaning of the sequences is encoded in the tokens, their relative order, and the degree to which a given token is related to the other tokens. Consider the phrases “the white house" and “la maison blanche". In order to effect a correct translation the model needs to encode the fact that “la" and “maison" are strongly related, while “the" and “house" are less so. The model also needs to encode the strong relationship between “the" and “la", between “house" and “maison" and between “white" and “blanche". That is, the model needs to *pay attention to* grammatical and semantic facts and structures.

The need for the model to pay attention to relevant linguistic facts is the basis of the so-called **attention mechanism** [39]. In the encoding stage, the model associates a vector to every token that tries to capture the strength of a token’s relationship to other tokens. Since this association mechanism operates within the same sequence (that is, within the same point cloud in the vector space in which the sequence is embedded) it is referred to as **self attention**. Presumably an effective self attention mechanism will note the fact that “la" and “maison" are strongly related and that the relative positions of “maison" and “blanche" is important as are the relative positions of “white" and “house". In the decoding stage of the model, in addition to the self attention over the target sequences another attention mechanism should pay attention to the fact that “the" and “la", “house" and “maison" and “white" and “blanche" are strongly related. We, therefore, expect a successful seq2seq neural network to model self attention in both the encoding and decoding phases and source-to-target attention in the decoding phase. While the optimal way to do this is unknown, the transformer model implements an attention mechanism that empirically appears to be highly effective.

4.2.3 Prediction

As noted the transformer is trained and used autoregressively: given source, i.e., input, sequence $\boldsymbol{x} = x_0, x_1, \cdots, x_k, x_{k+1}$ of length $k + 2$ tokens, where $x_0 \equiv$ <sos>, and $x_{k+1} \equiv$ <eos> are sequence delimiters and the current output sequence $\boldsymbol{y}_l = y_0, y_1, \cdots, y_{l-1}$ of length l tokens, the model approximates a discrete conditional probability distribution over the target vocabulary of size m tokens,

$$p_{ij} \equiv p(y_{ij}|\boldsymbol{x}, \boldsymbol{y}_l), \quad i = 0, \cdots, l, \quad j = 0, \cdots, m-1.$$

For a vocabulary of size m and a sequence of size k every position in the sequence can be filled in m ways. Hence there are m^k possible sequences of which the most probable is sought. This presents a severe computational challenge. Consider, for example, a sequence of size $k = 85$ tokens and a target vocabulary of size $m = 28$ tokens. There are $\sim 1 \times 10^{123}$ possible sentences. Even at a trillion probability calculations per second an exhaustive search would be utterly futile as it would take far longer to complete than the current age of the universe ($\sim 4 \times 10^{17}$ s)! We have no choice but to use heuristic strategies to search for the best output sequence. The simplest heuristic strategy is the **greedy search** in which one chooses the most probable token as the next token. A potentially better strategy is **beam search** in which at each prediction stage the n most probable sequences so far at kept. At the end the most probable output sequence among the n output sequences is chosen.

Stephen Wolfram [40] has noted that it is both astonishing and unexpected that the transformer model works as well as it does as there is no reason *a priori* why the human encoding of information using natural language should be amenable to mathematical modeling with neural networks. Wolfram further argues that the fact that ChatGPT works at all should be considered a major discovery about the nature of natural languages and how they encode information.

Summary

We have given an overview of the frequentist and Bayesian approaches to statistical inference and a brief survey of the main mathematical ideas that underpin supervised machine learning. Frequentist analysis is based on the relative frequency interpretation of probability and, ideally, adheres to the frequentist principle: repeated application of a statistical procedure will yield statements a fraction $f \geq p$ of which are guaranteed to be true, where p is the desired confidence level. The Bayesian approach uses the degree of belief interpretation of probability and Bayes theorem as the primary inference algorithm. In both approaches, the key task is building an accurate probability model.

A brief introduction to supervised machine learning was given in which the emphasis was clarifying the critical role of the loss function. We noted the mathematical fact that the quantity approximated by a machine learning model is determined by the loss function and not by the particulars of the model provided that sufficient training data are used, the model is sufficiently flexible, and a good approximation to the minimum of the average loss can be found.

Acknowledgement

I thank Martijn Mulders, Markus Elsing and Kate Ross for organizing and hosting a very enjoyable school and the students for their keen participation and youthful enthusiasm. These lectures were supported in part by US Department of Energy grant DE-SC0010102.

References

[1] F. James, *Statistical Methods in Experimental Physics*, 2nd Edition, World Scientific, Singapore (2006).

[2] L. Lyons, *Statistics for Nuclear and Particle Physicists*, Cambridge University Press, Cambridge (1989).

[3] R. J. Barlow, *Statistics: A Guide To The Use Of Statistical Methods In The Physical Sciences*, The Manchester Physics Series, John Wiley and Sons, New York (1989).

[4] G. Cowan, *Statistical Data Analysis*, Oxford University Press, Oxford (1998).

[5] S.K. Chatterjee, *Statistical Thought: A Perspective and History*, Oxford University Press, Oxford (2003).

[6] L. Daston, "How Probability Came To Be Objective And Subjective," Hist. Math. 21, 330 (1994).

[7] Joel L. Horowitz, "Bootstrap Methods in Econometrics", Annual Review of Economics 2019 11:1, 193-224.

[8] S. Chatrchyan *et al.* [CMS Collaboration], "Measurement of the properties of a Higgs boson in the four-lepton final state," Phys. Rev. D **89**, no. 9, 092007 (2014) doi:10.1103/PhysRevD.89.092007 [arXiv:1312.5353 [hep-ex]].

[9] A. H*á*jek, "The reference class problem is your problem too," Synthese (2007) 156:563–585.

[10] K. Cranmer, S. Kraml, H.B. Prosper, et al., "Publishing statistical models: Getting the most out of particle physics experiments," SciPost Phys. 12, 037 (2022).

[11] J. Neyman, "Outline of a Theory of Statistical Estimation Based on the Classical Theory of Probability," Phil. Trans. R. Soc. London A236, 333 (1937).

[12] G. J. Feldman and R. D. Cousins, "Unified approach to the classical statistical analysis of small signals," Phys. Rev. D57, 3873 (1998).

[13] S. E. Fienberg and D. V. Hinkley, eds., *R.A. Fisher: An Appreciation*, Lecture Notes on Statistics, Volume 1, Springer Verlag (1990).

[14] G. Cowan, K. Cranmer, E. Gross, O. Vitells "Asymptotic formulae for likelihood-based tests of new physics," Eur. Phys. J. C71, 1554 (2011).

[15] G. Taraldsen and B.H. Lindqvist, "Improper Priors Are Not Improper," The American Statistician, Vol. 64, Issue 2, 154 (2010).

[16] G. Aad *et al.* [ATLAS Collaboration], "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC," Phys. Lett. B **716**, 1 (2012) [arXiv:1207.7214 [hep-ex]].

[17] S. Chatrchyan *et al.* [CMS Collaboration], "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC," Phys. Lett. B **716**, 30 (2012) [arXiv:1207.7235 [hep-ex]].

[18] H. Jeffreys, *Theory of Probability*, 3rd Edition, Clarendon Press, Oxford (1961).

[19] V. M. Abazov *et al.* [D0 Collaboration], "Observation of Single Top Quark Production," Phys. Rev. Lett. **103**, 092001 (2009) [arXiv:0903.0850 [hep-ex]].

[20] T. Aaltonen *et al.* [CDF Collaboration], "First Observation of Electroweak Single Top Quark Production," Phys. Rev. Lett. **103**, 092002 (2009) [arXiv:0903.0885 [hep-ex]].

[21] S. Sekmen *et al.*, “Interpreting LHC SUSY searches in the phenomenological MSSM,” JHEP **1202**, 075 (2012) doi:10.1007/JHEP02(2012)075 [arXiv:1109.5119 [hep-ph]].

[22] V. Khachatryan *et al.* [CMS Collaboration], “Phenomenological MSSM interpretation of CMS searches in pp collisions at sqrt(s) = 7 and 8 TeV,” JHEP **1610**, 129 (2016) doi:10.1007/JHEP10(2016)129 [arXiv:1606.03577 [hep-ex]].

[23] V. Khachatryan *et al.* [CMS Collaboration], “Search for supersymmetry in pp collisions at sqrt(s) = 8 TeV in final states with boosted W bosons and b jets using razor variables,” Phys. Rev. D **93**, no. 9, 092009 (2016) doi:10.1103/PhysRevD.93.092009 [arXiv:1602.02917 [hep-ex]].

[24] L. Demortier, S. Jain and H. B. Prosper, “Reference priors for high energy physics,” Phys. Rev. D **82**, 034002 (2010) [arXiv:1002.1111 [stat.AP]].

[25] I. J. Myung, V. Balasubramanian, and M. A. Pitt, “Counting probability distributions: Differential geometry and model selection," PNAS, **97** 11170-11175 (2000); doi: 10.1073/pnas.170283897.

[26] A. Turing, “Computing Machinery and Intelligence," Mind **59** 433-460 (1950).

[27] D. Silver et al., “Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm," Science **362** 1140-1144 (2018); DOI: 10.1126/science.aar6404.

[28] Ruck et al., IEEE Trans. Neural Networks 4, 296-298 (1990).

[29] Wan, IEEE Trans. Neural Networks 4, 303-305 (1990).

[30] Richard and Lippmann, Neural Computation. 3, 461-483 (1991).

[31] H. J. Yang, B. P. Roe and J. Zhu, “Studies of boosted decision trees for MiniBooNE particle identification,” Nucl. Instrum. Meth. A **555**, 370 (2005) doi:10.1016/j.nima.2005.09.022 [physics/0508045].

[32] Y. Freund and R.E. Schapire, “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," Journal of Computer and Sys. Sci. **55** (1), 119 (1997).

[33] J. Friedman, T. Hastie and R. Tibshirani, “Additive logistic regression: a statistical view of boosting,” The Annals of Statistics, **28** (2), 377-386 (2000).

[34] P. Speckmayer, A. Hocker, J. Stelzer and H. Voss, “The toolkit for multivariate data analysis, TMVA 4,” J. Phys. Conf. Ser. **219**, 032057 (2010). doi:10.1088/1742-6596/219/3/032057

[35] Rene Brun and Fons Rademakers, “ROOT - An Object Oriented Data Analysis Framework," Proceedings AIHENP 96 Workshop, Lausanne, Sep. 1996, Nucl. Inst. & Meth. in Phys. Res. A **389**, 81-86 (1997). See also https://root.cern.ch/.

[36] Pedregosa et al., “Scikit-learn: Machine Learning in Python," JMLR 12, pp. 2825-2830, 2011. See also https://scikit-learn.org/.

[37] “PyTorch: An Imperative Style, High-Performance Deep Learning Library," Adam Paszke et al., arXiv:1912.01703 [cs.LG]. See also https:://pytorch.org/.

[38] OpenAI. (2023). ChatGPT (Mar 14 version) [Large language model]. https://chat.openai.com/chat/.

[39] “Attention Is All You Need," Ashish Vaswani et al., arXiv:1706.03762 [cs.CL]. See also https://nlp.seas.harvard.edu/annotated-transformer/.

[40] “What Is ChatGPT Doing ... and Why Does It Work?", Stephen Wolfram, Wolfram Media Inc. (2023); ISBN-13 978-1579550813.