Part II

Fundamental Interrogations

Chapter 3

Fundamental Concepts

In this chapter, I rigorously delineate six foundational concepts: object, interrogation, subject, proposition, model, and axiom system, drawing upon definitions from key references [1, 11, 190, 188, 224]. Figure 3.1 shows the relationship among the six concepts.

3.1 Object

The world consists of objects. An *object* is a class of entities owning a set of *intrinsic characteristics* that can be interrogated by those other than itself [224]. An intrinsic characteristic is often referred to as *a property*. I will formally define interrogation in Section 3.2.

A typical object could be a thing, a life, a phenomenon, an abstract concept, a process, or even a policy in the natural and social sciences, as well as engineering.

Counting is assigning a value to a property. A *quantity* is a countable property of an object, such as volume or mass. A *variable* [214] is a symbol that represents an unspecified or changing quantity.

An object has other properties that are not countable. Psychologist Stanley Smith Stevens explored this topic and classified four natures of quantities: nominal, ordinal (based on order [17]), interval, and ratio [189]. Definitely, there are other natures of quantity. For example, I consider interrogation and free will as two fundamental properties of intelligent life, which are also objects.

In the original article, Stanley Smith Stevens used the term "levels or scales of measurement." I use "nature" to avoid confusion with the specific meaning of "scales" or "levels" used in this book.

The nominal nature represents "the simplest form of measurement, where numbers serve as labels or identifiers, establishing an equality relation." The ordinal nature, in contrast, involves "ranking the items in a specific order." The interval nature exhibits an "equality of interval relation, where (1) the choice of a zero point is based on convention

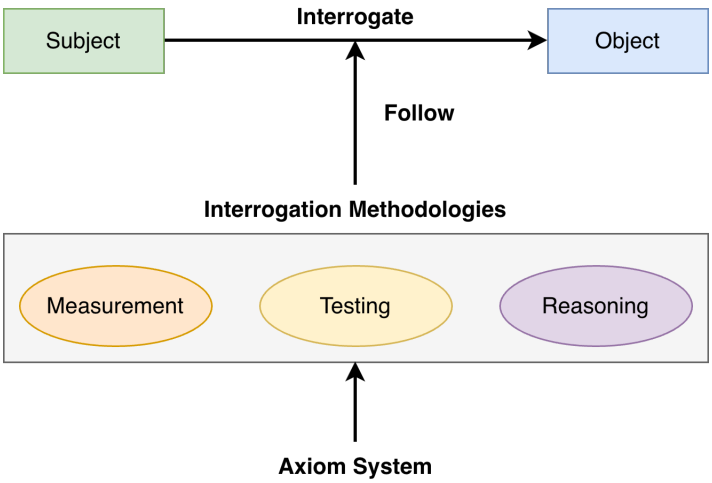

Figure 3.1: The relationships among the concepts of object, subject, interrogation, and axiom system.

or convenience, (2) there is rank ordering, and (3) the scale remains invariant when a constant is added to all values, preserving the differences between them." The ratio nature allows for "all four types of relations: equality, rank-ordering, equality of intervals, and equality of ratios."

Several Examples on Objects

A thing example

The weight of a table could be measured. A table is the object; its weight is its *quantity*.

A life example

A man could measure his height. A man is both the object and the subject; the height is his *quantity*.

A natural phenomenon example

The time of sunrise could be measured. Sunrise is the object, and the time of sunrise is its *quantity*.

An abstract concept example

A superstar can measure his or her number of social media followers. A superstar is both the object and the subject, and the number of social media followers is his or her *quantity*.

A policy example

The number of applicants applying for welfare could be measured. A welfare is the object, and the number of applicants is its *quantity*.

An *object* could be an *individual* or a *system*. An individual consists of components. A *system* is a coherent entity comprising interacting or interdependent individuals [1, 11, 224]. A system could be recursive. In the rest of this article, when referring to an object, we do not distinguish between an individual and a system unless stated explicitly.

An object has many instances. A *population* is the entire set of object instances, while a *sample* represents a smaller subset of object instances from the population [188]. A *parameter* is a number that describes some quantities of the population, while a *statistic* is a number that describes some quantities of a sample.

3.2 Interrogation

I begin by defining *interrogation* as *the capacity, capability, and process to understand objects and their mutual influences.* In Part II, I focus on the capacity, capability, and process to understand objects. The discussion of how objects influence one another will be addressed in Part III.

I define *an interrogation condition* as a setting under which objects and their mutual influences are interrogated. An interrogation condition consists of all objects that impact the interrogation outcomes. *Data* are raw interrogation outcomes or their derived ones in different interrogation conditions.

Interrogation has different levels of complexity. *Observation* is a kind of interrogation without the chance of changing the interrogation conditions. Instead, *experiment* is a kind of interrogation with the chance of changing the interrogated object, the interrogation conditions, and the interrogator.

There are three fundamental interrogations: *measurement*, *testing*, and *reasoning*. We will discuss them in Chapters 4, 5, and 6.

3.3 Subject

Free will is the capacity and capability to make free and intentional choices. Free will also has different degrees.

An automatic object is an object capable of interrogation but lacking free will, while *an intelligent life* is an object capable of interrogation with free will. Interrogation and free will are essentially two properties of an intelligent life. The *subject* is either an automatic object or an intelligent life.

A subject could be an object to be interrogated. Also, a subject could interrogate itself. An *artifact* refers to an object that demonstrates intentional conjecture, design, and fabrication by a subject.

According to the above definition, an *object*, defined in Section 3.1, could be redefined as a class of entities owning a set of *properties*, which a subject can interrogate.

Several Examples on Interrogation

A measurement example

A man could measure his height. A man is both the object and the subject; his height is the *quantity*.

A testing example

A man could test his visual acuity. A man is both the object and the subject; his visual acuity is the *properties*. The standard visual acuity chart serves as the *testing oracle*.

A reasoning example

Major Premise (Universal): All human-beings will die. *Minor Premise (Particular)*: Alice is a human being. *Conclusion*: Therefore, Alice will die.

3.4 Proposition

A *proposition* is a testable statement about an object. I will formally define what is testable in Chapter 5. A big letter A stands for a proposition.

Big letters $\mathcal{A}$ in calligraphy stand for a set that includes several (finite or infinite) numbers of propositions. $\mathcal{A}$ is usually *the power set* of A.

The power set of a given set A refers to a new set consisting of all possible subsets of A (including the empty set and A itself), denoted as $\mathcal{A}(A)$ or 2^A or $\mathcal{A}$.

A *premise* is a proposition that serves as the foundational statement from which a conclusion is logically derived. The *assumption* is a proposition accepted as truth or fact without proof, and it is a *provisional premise* adopted within a specific logical argument.

Hypothesis is a kind of proposition of an object. The difference between a hypothesis and an assumption is as follows.

Where P is a set of premises:

- *Assumptions* are elements of P: $P = \{A_1, A_2, \ldots, A_n\}$.
- *Hypotheses* are testable statements derived from P: $H \subseteq \mathcal{P}(P)$, which reads as "Let P be a set, and H is a subset of the power set of P."
- Validity holds only if: $(\bigwedge_{i=1}^{n} A_i) \to H$, which reads as "Validity holds only if the conjunction of all A_i from $i = 1$ to n implies H."

3.5 Model

A *model* is a streamlined representation of an object [221, 224]. A model can manifest as a physical, mathematical, or other construct. A valid model that passes the testing by a subject other than itself is a kind of *fact or truth*.

A mathematical model embodies a mathematical representation, frequently expressed through functions or equations, that captures the essence of an object.

Let's take the function as an example. A function [214] is a relation $f : X \to Y$ between sets X (domain) and Y (co-domain) that maps each $x \in X$ to exactly one $f(x) \in Y$.

According to [190, 224], "a function, denoted as f, is a rule that assigns a unique element, referred to as $f(x)$, from a set X to each element in a set Y." In this context, "the domain, denoted as X, refers to the set of all possible values for which the function is defined [190]." On the other hand, "the range of the function, denoted as $f(x)$, consists of all the possible values that $f(x)$ can take as x varies within the domain [190]."

The *independent variable* is represented by "a symbol that encompasses any arbitrary number within the domain of the function [190]." A *dependent variable*, represented by a symbol, "is used to denote a number within the range of the function [190]."

A random variable [103] is a measurable function $X : \Omega \to \mathbb{R}$ that extends from a probability space $(\Omega, \mathcal{F}, P)$ to $\mathbb{R}$ and assigns a number to each $\omega \in \Omega$. In this formulation, Ω is the total sample space of all possible events, $\mathcal{F}$ is the set of subsets of Ω representing events, and P stands for the probability.

For a random variable X, a distribution function [103] is a function $F_X : \mathbb{R} \to [0, 1]$ that is defined as $F_X(x) = P(X \leq x)$, stays non-decreasing, maintains right-continuity, and has $\lim_{x \to -\infty} F_X(x) = 0$ and $\lim_{x \to +\infty} F_X(x) = 1$.

3.6 Axiom System

A *fact or truth* is a proposition or model about an object that can be proven true or verified objectively by a subject other than itself. *Knowledge* contains facts or truths about objects.

The *axiom system* is *a collection of self-contained assumptions* for domain-specific knowledge or those beyond a specific domain.

For a certain proposition A, Proof(A) stands for a sequence of propositions, which starts from axioms, ends at A, and obeys the reasoning rules. The formulation $A \vdash B$ stands for the fact that there is at least a proof for B from A. If A is a false proposition, Proof(A) does not exist.

This book is built upon the axiom system. The next subsection explains what an axiom system is from different angles.

3.6.1 A Classical Perspective

From a classical perspective, axioms are self-evident assumptions.

- *Self-evident*: Axioms were seen as fundamental assumptions so obviously true that they required no proof.
- *Truths or Facts*: They were considered universal, unquestionable foundations.

A typical example of an axiom system

Euclid's 1st Axiom: "A straight line segment can be drawn between any two points."

The classical perspective, which is rooted in Euclidean geometry and rationalist philosophy (e.g., Descartes), holds that:

- *Self-evidence*: Axioms require no proof due to their intrinsic clarity. $\forall A \in \mathcal{A}_{\text{classical}}$, $\text{Proof}(A) = A$ where $\mathcal{A}_{\text{classical}}$ denotes classical axiom sets.
- *Truth or Fact*: Axioms are universally valid *a priori*:

$$\vdash A \quad \text{for all } A \in \mathcal{A}_{\text{classical}}. \tag{3.1}$$

3.6.2 Modern Perspective

Axioms are *not necessarily "self-evident" or absolute truths.* Instead, the consistency and arbitrariness of an axiom system are often used.

Consistent($\mathcal{A}$) stands for the fact that there is no proof from any propositions in $\mathcal{A}$ that leads to a contradiction. Arbitrary(A_1, A_2) stands for that, any inference taking A_1 or A_2 as an axiom is equivalent, namely $\forall B(A_1 \vdash B \iff A_2 \vdash B)$.

Axioms are defined as:

- *Foundational assumptions*: Arbitrary starting points chosen to build a logical system.
- *Defined by consistency*: Their validity depends on whether they generate non-contradictory results.
- *Relativity*: What is "self-evident" in one system may not hold in another.

A typical "self-evident" example in one system, which may not hold in another.

- *Euclid's 5th Axiom* (Parallel Postulate): "Through a point not on a line, exactly one parallel line exists."

- *Hyperbolic Geometry Axiom*: "Infinitely many parallel lines exist."

Both systems are logically consistent despite contradicting each other → Axioms are *conventions*, not universal truths.

Contemporary mathematics redefines axioms as:

1. *Formal Foundations*: $\mathcal{A}_{\text{modern}} := \{A_1, \ldots, A_n\}$ such that Consistent($\mathcal{A}_{\text{modern}}$) where consistency means no contradiction arises: $\nexists S\,(\mathcal{A}_{\text{modern}} \vdash S \wedge \neg S)$.
2. *Arbitrary Choices*: Axioms are conventions, not truths. For example:
   - Euclidean vs. non-Euclidean parallel postulates

3.6.3 Key Contrast

Classical View	*Modern View*
Axioms $\subseteq$ Truths	Axioms $\subseteq$ Assumptions
Self-evident	System-dependent

Table 3.1: Classical vs. modern view of axioms

3.6.4 Implications

Gödel's Incompleteness Theorems [62] further show:

- No consistent axiom system $\mathcal{A}$ can prove all arithmetical truths.

3.7 Interpreting Objects from a Perspective of Algebraic Structure

The algebraic system is established based on the axiomatic system. In this section, we describe objects from an algebraic perspective [1]. An algebraic structure [134] is a set A with one or more operations or mathematical properties that satisfy the specific axioms, e.g., $+ : A^2 \to A$, $\times : A^2 \to A$).

For example, a ring is denoted as $(A, +, \times)$. A ring is an algebraic structure.

[1] Mr. Hongxiao Li is the primary contributor of this section.

A category [54] is a mathematical structure consisting of a class of objects, denoted as $\mathcal{O}(\mathcal{C})$, and, for each pair of objects $A, B \in \mathcal{O}(\mathcal{C})$, a set of morphisms $\mathrm{Hom}_{\mathcal{C}}(A, B)$ from A to B.

Furthermore, a category must satisfy three axioms. First, morphisms admit composition: if $f \in \mathrm{Hom}_{\mathcal{C}}(A, B)$ and $g \in \mathrm{Hom}_{\mathcal{C}}(B, C)$, then their composition $g \circ f$ belongs to $\mathrm{Hom}_{\mathcal{C}}(A, C)$. Second, the composition of morphisms is associative: if $f \in \mathrm{Hom}_{\mathcal{C}}(A, B)$, $g \in \mathrm{Hom}_{\mathcal{C}}(B, C)$, and $h \in \mathrm{Hom}_{\mathcal{C}}(C, D)$, then the composition must satisfy the equation $h \circ (g \circ f) = (h \circ g) \circ f$. Third, each object has an identity morphism: for each $A \in \mathcal{O}(\mathcal{C})$, there exists an identity morphism $\mathrm{id}_A \in \mathrm{Hom}_{\mathcal{C}}(A, A)$ such that for all $f \in \mathrm{Hom}_{\mathcal{C}}(A, B)$ and $h \in \mathrm{Hom}_{\mathcal{C}}(C, A)$, the equations $f \circ \mathrm{id}_A = f$ and $\mathrm{id}_A \circ h = h$ hold.

A morphism [54] is a mathematical arrow (denoted as $f : A \to B$) that connects two objects A and B within a category $\mathcal{C}$. It belongs to the set $\mathrm{Hom}_{\mathcal{C}}(A, B)$ (the hom-set of the category) and can be composed with other morphisms. Specifically, if $f : A \to B$ and $g : B \to C$, their composition $g \circ f : A \to C$ satisfies the associativity condition: for any $h : C \to D$, we have $h \circ (g \circ f) = (h \circ g) \circ f$.

An object or its model can be represented as a variable, including a definite variable or a random variable, a function, a set, an algebraic structure, or other complicated structures like a category.

3.8 Summary

This chapter presents a concise system of concepts, including object, interrogation, subject, proposition, model, and axiom system.

Chapter 4

Metrology

Metrology is the science of measurement and its applications [17]. This chapter presents our interpretation of Metrology, including the basic concepts, problem statements, assumptions, methodologies, and case studies. Dr. Lei Wang contributed to Sections 4.7, 4.8.

4.1 Basic Concepts

A *reference* is a convention or standardization, or axiom, as I have discussed in subsection 3.6.2.

The *unit of measurement* is a definition of an ideal reference object with a unit quantity.

4.2 Definition of Measurement

I define *measurement* as attributing values to a quantity of an object by comparing with the unit quantity of a reference object under an interrogation condition. Measurement is a kind of experiment, as the subject could control the reference object. *Being measurable or measurability* means that an object's quantity can be compared with that of a reference object.

In metrology, measurement is often defined as "objective obtaining of one or more values attributed to a quantity [17]." I much like my definition, as it reveals the essence of measurement.

Another widespread definition of measurement in the social sciences is "the assignment of numerals to objects or events according to some rule [189]", dating back to 1946.

4.3 Problem Statement

I state the measurement problem as follows: Given a quantity, how can we define an ideal reference object with a unit of measurement, and hierarchically realize the definition of the reference object with different accuracies and overheads?

Specifically, the input of the measurement problem is a given quantity of objects, the outputs include the definition of a unit of measurement, and hierarchical realizations of the unit of measurement. The constraints are the accuracy and overhead of the definitions and realizations.

4.4 Basic Assumptions

For measurement, there are three basic assumptions.

First, for a countable property, there is a true value, independent of any other object. The *true value of a quantity* represents an inherent property of an object that is independent of any observer. For example, it can be the radius of a specific circle or the kinetic energy of a particular particle within a given system [97, 17]. In terms of measurement, the true quantity value is a target that any measurement approaches.

Second, an ideal reference object possesses a utilizable property that can serve as the basis for defining a unit of measurement.

Third, for a given quantity of any object, the subject could realize the definition of a reference object with a unit quantity, independent of time, space, and subjects.

These assumptions have four implications as follows.

- The reference object is from the definition by the subject, which is a convention or axiom.
- A ideal reference object with a utilizable property plays an important role in defining the unit of measurement.
- The unit quantity is independent of time, space, and any subjects.
- The subject could realize a reference object with a unit quantity.

4.5 Fundamental Roles of Measurement

Measurement plays two important roles in the subjects' interrogations. First, it provides a universally available reference for the same quantity of different objects. Second, it assigns values to the same quantity of different objects, which are the basis for generating propositions or models. The latter are the essential elements in testing and reasoning.

Building the measurement system obviously relies upon reasoning, which we will discuss in Chapter 6. In metrology, four implicit assumptions, which I clarify in Section 4.4, are essentially an axiom system.

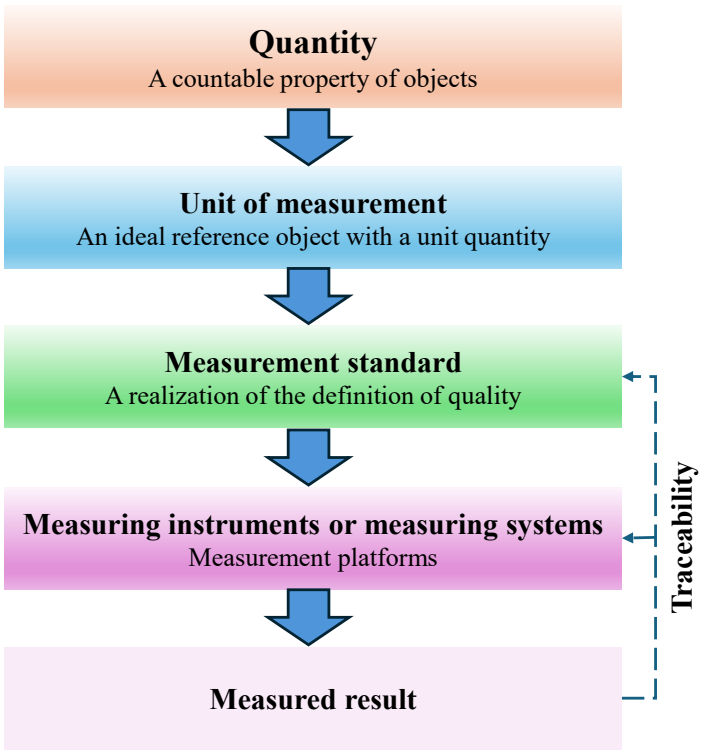

Figure 4.1: A simplified yet systematic conceptual framework for metrology [17, 97].

4.6 Methodology

The essence of metrology lies in quantities and their corresponding measurements. Figure 4.1 illustrates the systematic methodology. As shown in Figure 4.1, the quantity, the unit of measurement, the measurement standard, and measurement traceability are key components of Metrology.

The *unit of measurement* plays a fundamental role in the field of measurement [97], acting as a reference standard for comparing quantities of the same kind. It is a real scalar magnitude that is defined and adopted by convention, ensuring consistency across measurements. The unit serves to standardize measurements, allowing for their comparison and ensuring that different systems and observers can communicate precise, comparable data. Units are established through international agreements, forming the basis for uniformity and accuracy in scientific, industrial, and everyday applications.

Measurement standard [97] is a realization of the definition of quality. It is characterized by a stated metric value and an associated measurement uncertainty. To establish a measurement standard, it is important to use a *measurement methodology* that is both repeatable (performed by the same team) and reproducible (performed by different

teams). This ensures consistency and reliability in the reference for measurements. Such measurements can be conducted using *measuring instruments* or *measuring systems* [17], providing a reliable foundation for further analysis and comparison.

The hierarchy of measurement standards follows a progression from lower to upper levels, with increasing accuracy and cost. This progression starts from national measurement standards and extends to international standards. As a property of a measurement result, *measurement traceability* [17] establishes a connection between the result and a reference (measurement standards, measuring instruments, and measuring systems). This connection is established through a documented, unbroken chain of calibrations, with each calibration contributing to the measurement uncertainty. To ensure accuracy, each level of measurement standards in the hierarchy should be calibrated using a higher standard with greater precision.

4.7 Definition and Realization of the Fundamental Quantities

As mentioned earlier, a quantity is a countable property of an object that has a true value independent of any other object. The corresponding definition of a unit of measurement comes from an ideal reference object with a utilizable property. For example, length is a fundamental quantity, and its unit of measurement, the meter, is currently defined based on the speed of light in a vacuum. In this case, the light serves as the ideal reference object, and its property—the constant speed—is utilized to provide a precise and universally reproducible standard for measurement.

The international system of metrology encompasses *seven fundamental quantities*: time, length, mass, electric current, thermodynamic temperature, amount of substance, and luminous intensity [17]. These quantities form the foundation of all physical measurements, and all physical units and measurements are defined based on these fundamental quantities.

The seven fundamental quantities form the basis of the International System of Units (SI) and provide the foundation for all physical measurements. The International System of Units is defined by fixing the numerical values of seven defining physical constants as follows:

- *second* (s)
  The second, the SI unit of time, is defined by taking the fixed numerical value of the cesium frequency, $\Delta\nu_{\mathrm{Cs}}$, the unperturbed ground-state hyperfine transition frequency of the cesium-133 atom, to be $\Delta\nu_{\mathrm{Cs}} = 9\,192\,631\,770$ Hz, which defines the duration of one second [48]. In this case, the cesium-133 atom serves as the ideal reference object, and its property—the precise frequency of the hyperfine transition—is utilized to provide a standard for measuring time.

- *meter* (m)
  The meter,the SI unit of length, is defined by fixing the numerical value of the speed

of light in a vacuum, $c = 299\,792\,458$ m/s, such that one meter is the distance light travels in a vacuum during a time interval of $1/299\,792\,458$ seconds [48]. In this case, the ideal reference object is light in a vacuum, and its property of a fixed, invariant speed is used, thereby providing a precise and universally reproducible standard.

- *kilogram* (kg)
  The kilogram,the SI unit of mass, is defined by taking the fixed numerical value of the Planck constant at $h = 6.626\,070\,15 \times 10^{-34}$ kg m^2 s^{-1} [48]. In this case, the ideal reference object is a Kibble balance, a precision instrument used to measure the Planck constant. Its key property is the ability to measure the mechanical power required to balance the force of gravity on a mass using an electromagnetic force. By linking this property to the Planck constant, the Kibble balance enables the definition of the kilogram in terms of fundamental constants, ensuring a precise and reproducible standard of mass.

- *ampere* (A)
  The ampere,the SI unit of electric current, is defined by taking the fixed numerical value of the elementary charge at $e = 1.602\,176\,634 \times 10^{-19}$ C, where 1 C = 1 A s [48]. In this case, the ideal reference object is a single proton or electron. The elementary charge is the smallest indivisible unit of electric charge. The utilizable property is that the elementary charge carried by a single proton or electron is absolutely invariant and constant, which can be used to define the flow of electric charge in terms of the ampere. This precise definition of the elementary charge allows for the accurate measurement of current based on the number of charges passing a given point per second, providing a universally reproducible standard for electric current.

- *kelvin* (K)
  The kelvin,the SI unit of thermodynamic temperature, is defined by taking the fixed numerical value of the Boltzmann constant at $k = 1.380\,649 \times 10^{-23}$ kg m^{2}s^{-2}K^{-1} [48]. In this case, the ideal reference object is an ideal gas or a thermodynamic system whose particles exhibit random thermal motion. The property utilized is the average kinetic energy of the particles in the system, which is directly proportional to the thermodynamic temperature as described by the Boltzmann constant. By fixing the value of k, the kelvin is defined in terms of this fundamental physical relationship, providing a precise and reproducible standard for temperature measurement.

- *mole* (mol)
  The mole,the SI unit of amount of substance, is defined by taking the fixed numerical value of the Avogadro constant at $N_\mathrm{A} = 6.022\,140\,76 \times 10^{23}$ mol^{-1}, so that one mole contains exactly $6.022\,140\,76 \times 10^{23}$ specified elementary entities [48]. In this case, the ideal reference object is a collection of elementary entities, which can be atoms, molecules, ions, or other specified particles. The property utilized

is the exact number of these entities that corresponds to one mole. By fixing the value of the Avogadro constant, the mole is defined as the amount of substance that contains precisely this number of elementary entities, providing a universal and reproducible standard for quantifying matter.

- *candela* (cd)
  The candela, the SI unit of luminous intensity, is defined by taking the fixed numerical value of the luminous efficacy of monochromatic radiation of frequency 540×10^{12} Hz, K_{cd}. This definition implies the exact relation $K_{\mathrm{cd}} = 683\,\mathrm{cd\,sr\,kg^{-1}\,m^{-2}\,s^{3}}$ for monochromatic radiation of frequency $\nu = 540 \times 10^{12}$ Hz, where sr (steradian) is the SI unit of solid angle. Inverting this relation yields an exact expression for the candela: $1\ \mathrm{cd} = (K_{\mathrm{cd}}/683)\ \mathrm{kg\,m^{2}\,s^{-3}\,sr^{-1}}$ [48]. In this case, the ideal reference object is a perfectly monochromatic light source, specifically a laser or another light-emitting device that emits radiation at a frequency of 540×10^{12} Hz, which corresponds to green light. The property utilized is the luminous efficacy of this radiation, which describes the perceived brightness or intensity of the light per unit of energy. By fixing this value, the candela is defined as the luminous intensity of such a source emitting this frequency of radiation, providing a universal and reproducible standard for measuring light intensity.

4.8 Case Study: Definition and Realization of Meter

This section takes the Meter as an example to explain the definition and realization of the fundamental quantity.

4.8.1 Historical Definitions of Meter

Initial Definition of the Meter (1793) In 1793, the meter was defined as $1/10,000,000$ of the Earth's meridian quadrant [143]. *The Earth is the ideal reference object. Its property, the meridian quadrant, is utilized to define the unit of measurement.*

Before this definition, Europe used a variety of different units (such as feet, inches, and leagues), which differed widely across regions. To standardize measurements, the French Academy of Sciences proposed defining the meter as one ten-millionth of the distance from the North Pole to the Equator along the Paris meridian. This definition was a groundbreaking attempt to define a unit based on an unchanging natural property of the Earth, accessible to any nation through measurement. While subsequent definitions improved precision, the original concept, which rooted the meter in the Earth itself, remains central to the spirit of metrology.

To determine this length, astronomers Jean-Baptiste Joseph Delambre and Pierre Méchain undertook a monumental geodetic survey (1792-1799), measuring the meridian arc from Dunkirk to Barcelona [16]. This data formed the basis for calculating the full quadrant. During the survey, a provisional meter was created in 1793 based on the

preliminary results. However, the formal legal definition enacted that year was based on the survey's final results.

The International Prototype Meter (1875) In 1875, the CGPM (General Conference on Weights and Measures) redefined the meter using a platinum-iridium alloy bar (90% Pt, 10% Ir) [135]. This became known as the International Prototype Meter (IPM), or Bar No. 27, which was stored at BIPM (Bureau International des Poids et Mesures/International Bureau of Weights and Measures). *The ideal reference object is a platinum-iridium alloy bar (90% Pt, 10% Ir). Its property, the durability and low thermal expansion (8.7 µm/m°C), is utilized to define the unit of measurement.* The bar needed to be measured at 0°C. The design featured an X-shaped cross-section, with a total length of 102 cm, and two engraved lines marking 1 meter between their midpoints. Thirty copies were distributed to member nations for calibration, with Bar No. 6 being sent to the USA. Despite its revolutionary nature for 19th-century metrology, the physical standard posed risks, such as the potential for damage, microscopic wear, and accessibility challenges.

The Krypton-86 Definition (1960) The CGPM redefined the meter in 1960 using the emission spectrum of krypton-86 (^{86}Kr), marking the first definition based on a natural constant [41]. *The ideal reference object is krypton-86 (^{86}Kr), and its emission spectrum is utilized to define the unit of measurement.* The new definition specified that:

> 1 m = 1 650 763.73 wavelengths of the orange-red spectral line emitted
> by the ^{86}Kr isotope's transition between energy levels 2p and 5d,
> with 2p and 5d denoting specific atomic energy levels.

This definition achieved an accuracy of ±4 parts per billion (ppb), where 1 ppb denotes one part in a billion, i.e., a relative uncertainty of 10^{-9}, surpassing the limitations of the platinum-iridium bar. It used a discharge lamp filled with pure Kr vapor, excited electrically to emit the reference wavelength (605.780210 nm in vacuum). The new standard enabled global laboratories to independently realize the meter without needing to compare physical artifacts.

Transition to the Light-Speed Definition (1983) The 1960 krypton-86 definition demonstrated the SI system's adaptability to scientific progress. It laid the groundwork for the 1983 redefinition of the meter, based on the speed of light [201]. While the krypton-86 definition has since been superseded, it was a significant step in establishing metrological principles, particularly the use of atomic phenomena as the basis for measurements.

4.8.2 State-of-the-Art Definition of Meter

In 1983, the meter was defined as fixing the numerical value of the speed of light in vacuum (c) to be exactly 299 792 458 when expressed in the unit $\mathrm{m\,s^{-1}}$ [201]. *The ideal*

reference object is the light in vacuum, and its property, the constant speed, is utilized to define the unit of measurement. This definition indicates that: 1 m = c/299,792,458 s, where the speed of light in vacuum is defined as c = 299,792,458m s^{-1} (exact).

4.8.3 State-of-the-Art Realization of the Meter

The current definition of the meter is based on fixing the speed of light in a vacuum, and its realizations must be traceable to atomic time standards [48]. Among the various implementation methods, one approach is officially recommended by the BIPM as a primary standard for realizing the meter: the iodine-stabilized helium-neon laser.

- *Principle*: Wavelength locked to $^{127}I_2$ transition R(127) at 632.991 nm.
- *Components*:
  - HeNe laser (633 nm).
  - Temperature-controlled iodine cell (±0.01 °C).
  - Fabry-Pérot cavity (finesse > 100).

The following outlines the replication requirements for primary standards, including conditions for vacuum, thermal control, vibration, and traceability.

- *Vacuum*: $\leq 1 \times 10^{-6}$ mbar for primary standards.
- *Thermal Control*: ±0.1 °C (lab), ±1 °C (industrial).
- *Vibration*: $< 10\,\mathrm{nm\,s^{-2}}$ RMS.
- *Traceability*: all secondary measurement equipment must have valid calibration certificates traceable to national standards.

4.9 Summary

This chapter presents our interpretation of Metrology. Different from the classical definition of measurement: "objective obtaining of one or more values attributed to a quantity [17]", I define measurement based on comparing with a reference object, which reads "attributing values to a quantity of an object by comparing its quantity with the unit quantity of a reference object."

Also, I state the measurement problem and basic assumption, which serve as the axiom system for measurement. This thinking paradigm makes us ponder the essence of measurement and the fundamental role of measurement in interrogations. The historical reflections of the realization of fundamental quantities confirmed our thinking paradigm.

Chapter 5

Testology: The Science of Testing and Its Application

I propose to present the universal testing principles and methodology across different domains. I coined a new term, *Testology*, to cover this area. I define Testology as the science of testing and its application. Especially regarding the verification of theories and hypothesis testing, I believe these should fall under Testology, as they inherently follow the same testing principles. Dr. Lei Wang and I implemented this idea.

This chapter provides our interpretation of testing, including the basic concepts, problem statement, basic assumptions, fundamental principles, methodologies, and exemplary cases of testing.

I contributed to Sections 5.1, 5.2, 5.3, 5.4, and 5.5. Dr. Lei Wang contributed to Sections 5.6, 5.7, 5.8, 5.9, 5.10, and 5.11.

5.1 Basic Concepts

For better reading, we repeat some definitions in Section 3.6. A *proposition* is a testable statement about an object. A *model* is a streamlined representation of an object [221, 224]. A *fact or truth* is a proposition or a model about an object that can be proven true or verified objectively by a subject other than itself.

Test inputs refer to the data or stimuli provided to the object under test to drive the test. These may include parameters, user inputs, or configuration settings. For example, in a login system, the input might be a username and a password.

Test preconditions are the conditions set to ensure the object under test is in a valid state before running the test. Preconditions may involve system configurations or external conditions (such as power supply) that need to be met. For instance, for the same login system, a precondition could be that the user account must already exist in the database before testing the login functionality.

Test execution procedures define the steps or actions taken to run the test. In the

case of the login system, the execution procedure might involve opening the login page, entering the username and password, and clicking the "Login" button.

5.2 Definition of Testing

In this section, I present several essential testing concepts.

A test oracle is a ground truth or fact about an object. Additionally, for an artifact, a test oracle could be the intended behavior expected by the subject.

A test case is a predefined interrogation condition for testing, including test inputs, preconditions, and execution procedure, that is designed and implemented for a test oracle and is ready for executing an object under test to verify whether the actual outcomes are consistent with the mandated or expected results defined by the test oracle.

Testing is a verification process of running test cases to determine whether a proposition or a model of an object conforms to the test oracle through comparing their outcomes [224, 223]. *Being testable or testability* means a proposition or a model of an object can be falsifiable through testing.

5.3 Problem Statement

The testing problem could be stated as follows.

Given a test, how to design a set of test oracles and design and implement a corresponding set of test cases to balance the tradeoff between the testing accuracy and overhead?

The input of the testing problem is the object under test, while the output consists of a set of test oracles and a corresponding set of test cases that meet the stakeholders' requirements for testing accuracy and overhead.

5.4 Fundamental Assumptions

The most fundamental assumption in testing is that, for any object under test, we can always derive a complete set of test oracles and design and implement a corresponding complete set of test cases. Based on these, we can conclusively verify whether the object under test meets all testing criteria with 100% accuracy in the ideal scenario.

5.5 Fundamental Role of Testing

Testing is one of the fundamental interrogation methodologies, and its primary role is to falsify a proposition or a model about an object. Moreover, testing is the only means of interrogation that can validate a proposition or model as true or false.

From this perspective, it serves as a pillar for reasoning, which we will discuss in Chapter 6, as reasoning will generate different propositions or models, which can only be falsified by testing.

Meanwhile, it also serves as a pillar for evaluation, through which a subject could infer the effect induced by a cause object, which we will discuss in Part III. As inference is also reasoning, the inferred effect can also only be falsified by testing.

Just as the evaluation discussed in Section 2.2.9, testing is also practiced in an ad-hoc manner across different areas without a consensus on universal principles and methodologies. For example, testing is widely utilized in both hardware and software.

The fundamental role of testing is the reason why we coined a new term *Testology* to describe this area, the goal of which is to propose universal testing principles and methodologies.

5.6 Basic Principles

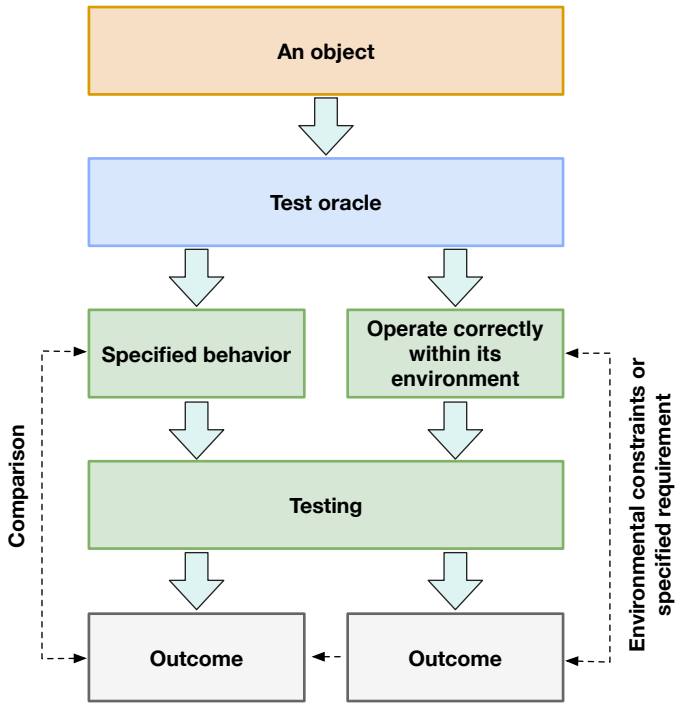

Figure 5.1: A simplified yet systematic conceptual framework for testing [13, 215].

As illustrated in Figure 5.1, the central concept of testing is the *test oracle*. A test

oracle functions as a reference, which we defined in Section 4.1, enabling the tester to distinguish between acceptable and unacceptable outcomes. The mandated or expected outcomes defined by a test oracle may be deterministic (exact values) or statistical (statistical values).

The process of testing involves running test cases to execute an object under test in predefined interrogation conditions defined by the test cases and subsequently comparing the outcomes against those mandated by the test oracles.

Two principal categories of test oracles may be distinguished. The first category focuses on verifying whether the object under test exhibits the outcome mandated by the test oracle. For instance, a sorting algorithm should return an outcome sequence in non-decreasing order for any valid input sequence.

The second category emphasizes the varying environments in which the object under test runs. The goal is to verify whether the object under test exhibits the outcome mandated by the test oracle in varying environments. This category of testing is to ensure that the artifact functions properly within the context under which it was designed, interacts appropriately with external services, complies with environmental constraints, and maintains stability under realistic operating conditions.

We further illustrate the principles of two typical testing methods: *Probability-based Testing* and *Deterministic Testing*, which differ in the test oracle.

5.6.1 Principles of Probability-based Testing Methodology

Probability-based Testing relies on a generalized ground truth: *under a hypothesis, if an observation outcome is unlikely, we would rather reject the hypothesis.*

In section 3.4, we have formally defined what a hypothesis is. A common example of probability-based testing is *hypothesis testing*, which tests whether an observed result is unlikely. The test oracle is: if the p-value, that is, the probability under the assumption that the null hypothesis is true, is less than the predetermined significance level α (e.g., 0.05), the null hypothesis H_0 is rejected.

For instance, in a clinical trial evaluating a new drug, the null hypothesis H_0 may assert that the drug does not affect blood pressure, implying that the average change in the Treatment Group is zero. The observed data are compared to this null hypothesis, and if the observed mean change significantly differs from zero (e.g., p-value < 0.05), the null hypothesis H_0 is rejected, suggesting that the drug likely affects blood pressure.

5.6.2 Principles of Deterministic Testing Methodology

Deterministic Testing verifies whether a system behaves exactly as expected under predefined conditions, producing a *definitive pass or fail* [215].

In deterministic testing, the test oracle is usually a set of predefined specifications or requirements, and correctness is determined by comparing whether the system's behavior matches these expectations.

For example, in the field of computing, two major forms of testing are *software testing* and *hardware testing*. Software and hardware are essentially artifacts, which we defined

in Section 3.3. Those methodologies could be extended to other artifacts.

- *Software testing* uses *test oracles*, such as specifications, reference implementations, or previous system versions, to determine correctness [37]. Failures can be classified as functional, when actual outcomes deviate from expected outcomes, or environmental, when constraints such as memory, performance, or compatibility are violated [13]. These failures could be cascading: for example, insufficient memory (an environmental failure) may cause the application to crash, resulting in a functional failure.

  A concrete example is testing a web application login: the test oracle specifies that valid credentials should return a successful login message. If the application crashes or returns an error despite correct credentials, it is a functional failure. If the server has insufficient memory, causing the application to slow down or become unresponsive under heavy load, it is an environmental failure that may indirectly trigger functional failures.

- A *test vector* consists of input sequences, expected outcomes, and fault models, simulating real-world operations to detect design or manufacturing defects. Test vectors are widely used test oracles in hardware testing for CPUs and digital circuits.

  For example, consider a 64-bit CPU with a 64-bit unsigned adder. A test vector may apply inputs A=0x0000000000000001 and B=0x0000000000000002. The expected outcome is S=0x0000000000000003 with carry-out=0. If the CPU produces the wrong sum or carry-out, it indicates a functional failure. Additionally, if a register has a stuck-at-0 fault (modeled in the fault model), the CPU will fail to produce the correct outcome, revealing a hardware defect.

5.7 Basic Methodologies

The basic methodology of testing is designing a subset of test oracle, designing and implementing a corresponding subset of test cases, and finally executing a subset of test cases to determine if the test passes by comparing the test results with those of the set of test oracle.

The testing process can be generalized into a unified, high-level framework, consisting of the following sequential steps:

1. *Test Oracle Definition*: A test oracle is defined to specify the mandated outcomes of a proposition or a model about an object.

2. *Test Case Design for a Test Oracle*: A set of test cases, including test inputs, preconditions, and execution procedures, is defined to compare the object's behavior against the expected outcomes described by the test oracle.

3. *Test Case Implementation*: According to the test case design, the test cases are implemented taking into account the characteristics of different environments.

4. *Test Case Execution*: The test case is executed, and the actual outputs or behaviors are then compared with the expected outcomes defined by the test oracle. Based on this comparison, a pass/fail determination is made for each test case.

In the following sections, we introduce the application of Testology in three areas: verifying a theory in Physics, significant testing in statistics, and software and hardware testing in computers. There are other applications of Testology, and we consider drafting a book on this topic in the future.

5.8 Verifying a Theory

We take the verification of parity non-conservation as an example to illustrate how to test a theory.

Parity (P) symmetry is one of the fundamental symmetries in classical physics and early quantum theory, describing the invariance of physical laws under spatial inversion [73]. Formally, a parity transformation replaces the spatial coordinates of a system (x, y, z) with $(-x, -y, -z)$, effectively converting a right-handed coordinate system into a left-handed one. A physical process is said to *conserve parity* if it remains unchanged under this transformation, meaning that its mirror-reflected version—expressed in the opposite-handed coordinate system—is physically indistinguishable from the original.

For a long time, many interactions, including electromagnetic and strong interactions, were believed to obey this symmetry. However, in 1956, Tsung-Dao Lee and Chen-Ning Yang conducted an in-depth analysis of experimental data and theoretical assumptions, proposing the hypothesis that parity might not be conserved in weak interactions [113]. They also put forward specific experimental ideas to detect potential asymmetries in particle emission directions. Parity non-conservation implies that the mirror image of a physical process—equivalently, its realization in the opposite-handed coordinate system—may not occur in nature, leading to observable asymmetries in particle behavior. In other words, under weak interactions, nature is capable of distinguishing between left-handed and right-handed coordinate systems.

In 1957, Chien-Shiung Wu and her team first directly verified parity non-conservation in weak interactions through experiments on cobalt-60 (Co^{60}) nuclei [219]. In the experiment, Co^{60} nuclei emit electrons via β-decay. The team used *adiabatic demagnetization* to polarize the Co^{60} nuclear spins at extremely low temperatures and employed a *single anthracene crystal scintillation detector*. By reversing the polarization magnetic field and observing changes in the counting rate of the same detector, they measured the asymmetry in the electron angular distribution. This approach effectively eliminated systematic errors caused by efficiency differences among multiple detectors. The experimental results clearly showed that electrons were emitted preferentially in the direction opposite to the nuclear spin, a phenomenon later confirmed through control experiments

to rule out possible magnetic artifacts. This discovery completely overturned the previous widespread belief that "parity is conserved in all interactions," exerting a profound impact on particle physics theory and experimental methods.

To systematically understand the scientific methodology of Wu's experiment, the *Testology* framework can be adopted.

In the definition of *test oracle*, Wu's testing experiments state that: in β-decay, if parity is conserved, the spatial distribution of emitted electrons along the nuclear spin axis should be symmetric. Any observed bias in the electron emission direction would indicate parity violation.

The *test case design* is a predefined experimental setup, including test inputs, preconditions, and execution procedures as follows:

- *Inputs*: High-purity Co^{60} radioactive sample (with known decay half-life and intensity).
- *Preconditions*:
    1. *Nuclear Polarization Apparatus*: The experiment used the Rose-Gorter method via adiabatic demagnetization of a cerium magnesium nitrate crystal to polarize Co^{60} nuclei.
    2. *Integrated Beta-Particle Detector*: A thin anthracene crystal scintillator was placed inside the vacuum chamber above the ^{60}Co source to detect the emitted beta particles.
    3. *Polarization Monitoring System*: Two additional NaI gamma-ray scintillation counters were installed—one in the equatorial plane and one near the polar position to monitor the polarization state of the sample.
    4. *Sample Preparation*: The Co^{60} sample was grown as a thin crystalline layer on the upper surface of good single crystals of cerium magnesium nitrate.
    5. *Control Experiments Preparation*: Two control specimens were prepared to eliminate potential artifacts from magnetic effects during the experiment.
- *Execution Procedures*:
    1. *Sample Mounting*: The Co^{60} sample crystal was mounted at the center of the demagnetization apparatus.
    2. *Polarization Process*: Adiabatic demagnetization was performed, followed by activation of the vertical solenoid to align the spins (20-second process).
    3. *Data Acquisition*: Beta and gamma counting commenced immediately:
        - Beta pulses were analyzed using a 10-channel pulse-height analyzer (1-minute intervals).
        - Gamma anisotropy was continuously monitored as an indicator of polarization.

4. *Field Reversal*: The polarizing field direction was reversed in subsequent runs to verify the authenticity of any observed asymmetry.
5. *Results*: The experimental data showed clear asymmetry effects, with electron emission significantly favored in the anti-spin direction.
6. *Control Verification*: Control experiments confirmed the absence of asymmetry under non-polarizing conditions.

In the *Test case implementation* stage, the experimental design is transformed into practical operations: the nuclei are magnetized at low temperature to orient their spins, the detectors are calibrated and arranged along the nuclear spin axis, and the instrument sensitivity and background noise are strictly controlled to ensure reliable data.

In the *Test case execution* stage, the electron counting rates along the nuclear spin direction and the opposite direction are recorded. If parity were conserved, the distributions in both directions would be symmetric. Wu's experiment, however, observed a significant bias in electron emission towards the anti-spin direction, providing direct evidence of parity non-conservation in weak interactions.

So, the universal methodology in *Testology* can be used to verify a theory.

5.9 Hypothesis Testing

Hypothesis testing is a statistical procedure that uses sample data to evaluate a null hypothesis (H_0) against an alternative (H_1) [132]. It employs a test statistic to measure the discrepancy between the data and H_0, and a decision rule (based on comparing the p-value to a significance level α) to either reject or fail to reject H_0.

In hypothesis testing, the *test oracle* comprises the null hypothesis (H_0), the alternative hypothesis (H_1), and the significance level (α), which together set the criteria for assessing whether the object's behavior significantly deviates from expectation. The *test case input* is the sample data, while the *test execution procedures* involve calculating the test statistic according to the *test oracle*. Finally, based on the calculated test statistic and the decision criteria, making the final decision to either reject H_0 or fail to reject H_0 is a pass/fail determination.

Under the *Testology* framework, hypothesis testing can be described as follows.

1. *Test Oracle Definition*: In hypothesis testing, the *test oracle* comprises the null hypothesis (H_0), the alternative hypothesis (H_1), and the significance level (α), which together set the criteria for assessing whether the object's behavior significantly deviates from expectation.

    - *Null Hypothesis (H_0)*: the observed outcome is consistent with the theoretical model, indicating no difference (e.g., the mean blood pressure of patients taking a new drug is equal to the population mean, $\mu = \mu_0$).

- *Alternative Hypothesis (H_1)*: the observed outcome show a statistically significant deviation from what is expected under H_0 (e.g., the mean blood pressure of patients taking the new drug is different from the population mean, $\mu \neq \mu_0$).

The decision rule depends on the significance level (α) and the p-value:

- *Significance Level (α)*: The pre-specified probability of rejecting H_0 when it is actually true. Commonly, $\alpha = 0.05$ is used:

$$P(\text{Reject } H_0 \mid H_0 \text{ true}) = \alpha. \tag{5.1}$$

- *p-value*: The probability of obtaining a test statistic at least as extreme as the observed one, assuming H_0 is true. The p-value computation depends on the type of test:
  - *Right-tailed test:* tests if the mean is greater than the hypothesized value ($H_0 : \mu \leq \mu_0, \quad H_1 : \mu > \mu_0$).

$$p\text{-value} = P(X \geq x_{\text{obs}} \mid H_0). \tag{5.2}$$

  - *Left-tailed test:* tests if the mean is less than the hypothesized value ($H_0 : \mu \geq \mu_0, \quad H_1 : \mu < \mu_0$).

$$p\text{-value} = P(X \leq x_{\text{obs}} \mid H_0). \tag{5.3}$$

  - *Two-tailed test:* tests if the mean is different from the hypothesized value ($H_0 : \mu = \mu_0, \quad H_1 : \mu \neq \mu_0$).

$$p\text{-value} = 2P(X \geq |x_{\text{obs}}| \mid H_0). \tag{5.4}$$

The decision rule is:

$$\text{Reject } H_0 \text{ if } p\text{-value} \leq \alpha.$$

2. *Test Case Design*: The test case is based on the sample data. The test statistic quantifies the discrepancy between the sample estimate and the null hypothesis. It is generally expressed as:

$$X = \frac{\hat{\theta} - \theta_0}{\text{SE}(\hat{\theta})}, \tag{5.5}$$

where $\hat{\theta}$ is the estimator derived from the sample, θ_0 is the parameter value under H_0, and $\text{SE}(\hat{\theta})$ denotes its standard error. The test case design specifies how to calculate the test statistic under the test oracle, including:

- *Test inputs (sample data)* to compute the estimator $\hat{\theta}$;
- *Preconditions*: primarily include assumptions about the underlying distribution, whether the variance is known or unknown, and the independence of samples.

- *Execution Procedures*:
  (a) *Calculate the test statistic:* Depending on the preconditions, and common test statistics include:
    – *t-test* (for population variance unknown):

$$t = \frac{\bar{X} - \mu_0}{s/\sqrt{n}}, \tag{5.6}$$

where $\bar{X}$ is the sample mean, μ_0 is the population mean under the null hypothesis H_0, s is the sample standard deviation, and n is the sample size.

    – *z-test* (for population variance known):

$$z = \frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}}, \tag{5.7}$$

where $\bar{X}$ is the sample mean, μ_0 is the population mean under the null hypothesis H_0, σ is the population standard deviation, and n is the sample size.

  (b) *Derive the p-value:* After obtaining the observed test statistic x_{obs} (t or z), the p-value is computed based on Equation 5.2, Equation 5.3, or Equation 5.4.

3. *Test Case Implementation*: This step transforms the test case design into an executable statistical procedure, for example, by looking up values in statistical tables or by writing a program in MATLAB to automate the calculations.

4. *Test Case Execution*: Execute the implemented test case and obtain actual outputs:
   (a) Calculate the observed test statistic x_{obs};
   (b) Calculate the corresponding p-value;
   (c) Apply the oracle decision rules:
     - If p-value $\leq \alpha$, *Reject* H_0;
     - If p-value $> \alpha$, *Fail to reject* H_0.

To illustrate this framework, consider a clinical trial conducted to evaluate the effect of a new drug on blood pressure. The null and alternative hypotheses are: $H_0 : \mu = 0$, the drug has no effect on blood pressure; $H_1 : \mu \neq 0$, the drug affects blood pressure. μ is the population mean change in blood pressure in the treatment group. The sample data if from 20 patients and obtain a sample mean and standard deviation of: $\bar{X} = 0.774$, $s = 2$. For population variance unknown, we use the t-test, the t statistic is calculated as: $t = \frac{\bar{X}-\mu_0}{s/\sqrt{n}} = \frac{0.774-0}{2/\sqrt{20}} = 1.73$. The two-tailed p-value is: $p\text{-value} = 2P(T \geq |t|) = 0.1$. Since p-value $= 0.1 > 0.05$, we fail to reject the null hypothesis. There is not enough statistical evidence to conclude that the drug significantly affects blood pressure.

In summary, hypothesis testing provides a probabilistic decision framework for assessing whether observed system outcomes are consistent with an expected model (H_0) or whether sufficient evidence exists to support an alternative hypothesis (H_1).

5.10 Software Testing

Software testing aims to verify whether a software system behaves as expected and satisfies its specified requirements. Failures detected during testing can be categorized into functional and environmental failures. Functional failures occur when the system produces incorrect results, while environmental failures—such as insufficient memory or slow execution—may indirectly lead to functional failures [13].

Under the *Testology* framework, software testing can be described as follows.

1. *Test Oracle Definition*: A test oracle is defined to specify the expected outcomes of a system or software component. The purpose is to ensure that the system's behavior aligns with the expected results. For instance, in a login system, if a user enters a valid username and the correct password, the expected outcome according to the test oracle is that the login is successful and the user is redirected to the homepage.

2. *Test Case Design*: A test case is defined, including specific inputs, preconditions, and execution procedures, to compare the system's behavior against the expected outcomes described by the test oracle. For example, in a login system, the input might be a username "admin" and the correct password; a precondition might be that the user account "admin" already exists in the system's database; and the execution procedure might involve opening the login page, entering the username and password, and clicking the "Login" button.

3. *Test Case Implementation*: According to the test case design, the test cases are implemented by creating the necessary test artifacts (e.g., test scripts, input files, or execution instructions), setting up the required testing environment, and ensuring that all inputs, preconditions, and execution steps are correctly instantiated. For instance, to implement the login test case, a tester could create a test script that opens the login page, inputs username "admin" and the correct password, clicks "Login" and verifies that the system redirects to the homepage. This ensures that the test case is executable and can reliably verify the expected behavior across different environments.

4. *Test Case Execution*: The implemented test cases are executed to verify whether the system behaves as expected. For example, the login test case would be executed by running the test script or performing the manual steps: open the login page, enter username "admin" and the correct password, click "Login" and observe the system's response. The actual output is compared against the expected outcome defined by the test oracle (redirection to the homepage). Based on this comparison, a determination is made that the test passes if the user is successfully redirected, or fails if the expected behavior does not occur.

Software testing methodologies can be categorized along different dimensions. Based on the level of knowledge about the system's internals, we have:

- Black-Box Testing. Black-box testing focuses on validating the functionality of the system without knowledge of its internal workings. This testing is based purely on the system's inputs and outcomes. For example, on an e-commerce site, "equivalence partitioning" can be used to test various inputs for a login form, such as ensuring that email addresses of different formats (valid or invalid) trigger the appropriate system response. "Boundary analysis" might be applied to test password length, ensuring that both the minimum and maximum character limits are correctly enforced by the system.

- White-Box Testing. White-box testing ensures that the system's internal logic is functioning correctly and aims to maximize test coverage. For example, on an e-commerce site, "code coverage" can be used to validate that every part of the code base—such as shipping calculations—is adequately tested. "Mutation testing" involves introducing small, controlled changes (mutations) to the code base and running tests to ensure that the test suite can detect these errors, ultimately confirming the thoroughness of the tests.

These approaches can be further implemented through different means:

- Manual Testing. The goal of manual testing is to ensure that the system functions as expected from the user's perspective. For example, consider an e-commerce site; testers may manually simulate the checkout experience to verify that all form fields are correctly populated and that users can successfully complete a purchase.

- Automated Testing. Automated testing utilizes scripts and tools to execute test cases without human intervention, enabling efficient and frequent regression validation. This approach ensures that previously verified features continue to function correctly after system updates, changes, or new feature deployments. For example, in an e-commerce site, automated tests can be programmed to simulate a wide range of user actions—such as user registration, product search, adding items to the shopping cart, and completing the checkout process—and automatically verify that the actual outcomes at each step match the expected results. The key advantage of automation is the ability to rapidly repeat these critical validation steps with consistent accuracy, significantly improving testing coverage and efficiency compared to manual execution.

5.11 Hardware Testing

The test vector approach is widely used in hardware testing for CPUs and digital circuits, and it involves predefined input sequences applied to a circuit (or hardware system) under test to verify its correctness against a test oracle [126].

Under the *Testology* framework, hardware test vector testing can be described as follows.

1. *Test Oracle Definition*: In hardware test vector testing, the test oracle is the reference used to determine the correct expected outcome for a given input vector V_i. It defines the expected behavior of the system under test by specifying the expected result $R_{\text{expected}}(V_i)$ for each input vector.

2. *Test Case Design*: A test case consists of input, preconditions, and execution procedures. A specific input vector V_i is the input. Test case preconditions include the necessary system state setup, such as powering on the device, initializing registers, and ensuring the circuit is in a known starting state. Execution procedures define the exact steps to apply V_i to the circuit and measure the output.

3. *Test Case Implementation*: Implementing a test case entails preparing the input vector and setting up the hardware test environment. Specifically, this involves loading the input vector into the test equipment, initializing the hardware under test, and verifying that all preconditions are met to ensure reliable application of the stimuli.

4. *Test Case Execution*: During execution, the input vector V_i is applied to the hardware, and the actual outcome $R_{\text{actual}}(V_i)$ is observed. The actual output is then compared against the expected output defined by the test oracle. The system passes the test for V_i if $R_{\text{actual}}(V_i) = R_{\text{expected}}(V_i)$; otherwise, a failure is reported.

Hardware testing methodologies for CPU validation can be categorized along different dimensions. Based on the level of knowledge about the system's internal microarchitecture, we have:

- Black-Box Testing. Black-box testing in CPU testing focuses on validating the behavior of the processor's instruction set architecture without any knowledge of its internal workings. This method tests the processor's ability to execute instructions as expected. "Equivalence partitioning" is used to validate the behavior of specific instructions, such as the "ADD" operation, ensuring that different input ranges are handled correctly. For example, boundary checks for overflow might be used to ensure that adding two large numbers does not cause incorrect results due to overflow. "Fuzzing illegal opcodes" is another technique used in black-box testing, where random or malformed instructions are fed into the CPU to identify any potential vulnerabilities or undefined behaviors.

- White-Box Testing. White-box testing focuses on verifying the internal logic of the CPU by ensuring that the processor's internal structure and pipeline handle all possible execution paths. One common technique is "path coverage," which ensures that all possible paths through the processor's pipeline (such as handling various exceptions) are tested. Another approach is "mutation testing," where small, controlled changes (mutations) are made to the processor's logic (e.g., flipping a comparator) to check if the existing test suite can detect these errors.

These approaches can be further implemented through different technical means:

- *Simulation-Based Testing.* This is the most common approach, dynamically verifying CPU functionality by running test programs in a simulation environment. For instance, engineers develop or generate extensive test vector programs that are executed in a simulator, with results compared against a golden reference model. This method's advantage lies in its flexibility to simulate complex scenarios and long execution sequences.

- *Formal Verification.* Formal verification is a static verification approach that uses mathematical methods to prove certain aspects of design correctness exhaustively, with-

out running the test system. For example, critical properties of the microarchitecture, such as data consistency in the pipeline, can be formally verified to hold under all possible instruction sequences. This method is particularly strong at uncovering subtle, deep design errors that are difficult to hit with random testing.

- *Hybrid Methods.* Complex CPU validation often employs hybrid techniques. *Concurrent hybrid verification*, for instance, combines simulation's flexibility with formal verification's exhaustiveness. While simulation runs, formal tools analyze reachable states from the current state and may automatically generate new test vectors to cover unexplored paths, thereby more systematically exercising various corner cases.

5.12 Summary

Testing can only show the presence of violating the ground truth or fact, but can not prove that there is no existence of violating the ground truth or fact. Exhaustive Testing is Impossible. Testing identifies defects; debugging (by developers) fixes them. These are separate but complementary processes. For example, in software testing, testing can demonstrate that defects exist in the software, but it cannot prove that the software is defect-free. Even if no bugs are found, it does not mean the system is perfect—only that no defects were detected under the tested conditions. It is impractical to test every possible input, combination, or scenario, especially in complex systems. Instead, testers use risk analysis and prioritization to focus on the most critical areas.

Chapter 6

Reasoning

This chapter provides the basic concepts, problem statement, basic assumptions, fundamental principles, methodologies, and exemplary cases of reasoning. Dr. Fanda Fan and I co-authored this chapter. Dr. Fanda Fan authored Sections 6.6, 6.7, 6.8, 6.9, and 6.10. I authored Sections 6.1, 6.2, 6.3, 6.4, and 6.5.

6.1 Basic Concepts

For better reading, I repeat several definitions in Sections 3.4, 3.5, and 3.6.

A *proposition* is a testable statement about an object. A *model* is a streamlined representation of an object [221, 224]. A *fact or truth* is a proposition or model about an object that can be proven true or verified objectively by a subject other than itself. *Knowledge* contains facts or truths about objects. *An Evidence* is data about an object that supports, refutes, or informs a proposition or model.

A *premise* is a proposition or model that serves as the foundational statement from which a conclusion is logically derived. A *conclusion* is the proposition or model obtained through a reasoning process.

Hypothesis is a kind of proposition or model of an object. The *assumption* is a proposition or model accepted as truth or fact without proof, and it is a *provisional premise* adopted within a specific logical argument. *An axiom system* is *a collection of self-contained assumptions* for domain-specific knowledge or those beyond a specific domain.

6.2 Definition of Reasoning

I defined *reasoning or inference* as *a capacity, capability, and mental process of generating new propositions or models about an object based on the premises, the facts or truths, and the interrogation outcomes.* In the rest of this article, we do not distinguish between reasoning and inference if not explicitly stated.

A reasoning or inference rule is a rule that generates a new proposition or model based on the premises, the facts or truths, and the interrogation outcomes.

Prediction is a kind of reasoning to infer the interrogation outcome of an object according to its truth, fact, assumption, or model.

6.3 Problem Statement

Depending on different purposes, the reasoning problem can be stated in different ways. For those interested in the knowledge system, the reasoning problem can be stated as "how to propose a simple and concise axiom system, based on which we can build the corresponding knowledge systems."

For those who are interested in reasoning or inference itself, they care about the effectiveness or efficiency of reasoning. For the former purpose, the focus is on the outcome: the problem could be stated as "whether and how the desired propositions or models can be achieved, which is often referred to as proof." For the latter purpose, the focus is on the reasoning process: the problem could be stated as "how to reason according to the correct rules with minimal waste in terms of time, resources, and effort."

6.4 Fundamental Assumptions

There are two fundamental assumptions in reasoning.

First, it is assumed that several propositions are correct without the need for proof, which is often called the axiom system.

Second, there is a concise set of perfect reasoning rules, which can pass testing 100%. However, if a contradiction happens by testing, the reasoning rule is invalidated.

6.5 Fundamental Role of Reasoning

Reasoning is a mental capacity and capability, and only a subject can own this capability and capacity. A subject can use a mental activity to replace a real activity in the physical world.

From this perspective, reasoning has three fundamental roles. First, it augments the subject's capability to understand objects. The subject could generate new propositions or models about objects; Reasoning, as a mental activity, could replace physical activities in the real world, and relieve some tasks through reasoning.

Second, through reasoning, we can build a simple and concise axiom system for the knowledge system. For example, we can restructure or improve the measurement and testing methodologies by building them upon a different axiom system.

Third, through studying the nature of reasoning, we can understand the inherent flaws of our knowledge system, e.g., Gödel's Incompleteness Theorems [62] discussed in Section 3.6.4.

6.6 Basic Principles

Logical reasoning is grounded in *formal systems*: structured collections of symbols, rules, and axioms that specify how valid conclusions can be derived. Among these systems, *first-order logic (FOL)* [64] provides the canonical foundation. It consists of:

- **Variables** $x, y, z, \ldots$
- **Predicates** $P, Q, \ldots$
- **Logical connectives** $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- **Quantifiers** $\forall, \exists$

A typical expression takes the form $\forall x\,(P(x) \rightarrow Q(x))$, which asserts a universally quantified implication.

Operator	Symbol	Meaning
Negation	$\neg P$	Not P
Conjunction	$P \wedge Q$	P and Q
Disjunction	$P \vee Q$	P or Q
Implication	$P \rightarrow Q$	If P, then Q
Biconditional	$P \leftrightarrow Q$	P iff Q
Universal Quantifier	$\forall x$	For all x
Existential Quantifier	$\exists x$	There exists x

Table 6.1: Common logical operators and their meanings.

Axiom Systems in Logic

Based on these symbols and rules of inference, various *axiom systems* have been proposed to formalize valid reasoning. Classical logic begins with foundational principles such as:

Non-Contradiction (Aristotle). Aristotle's metaphysics [7] formulates the basic law:

$$\neg(P \wedge \neg P), \tag{6.1}$$

stating that no proposition P can be true and false at the same time. This principle underlies all coherent evaluative reasoning: contradictory claims cannot simultaneously support a valid conclusion.

Limits of Axiomatic Systems (Gödel). Kurt Gödel's incompleteness theorems [69] reveal a fundamental constraint: no sufficiently expressive axiomatic system can be both *complete* and *consistent*. That is, some true statements cannot be derived from the system's own axioms.

These results clarify an important point for reasoning in evaluative contexts: even with well-defined rules, *no formal system can capture all truths solely through axioms*. Thus, logical reasoning provides structure and rigor, but it does not eliminate the need for empirical testing, contextual analysis, or interpretive judgment.

First-order logic [64] is a formal system that includes individual variables $x, y, z, \cdots$, predicates $P, Q, \cdots$, quantifiers $\forall, \exists$, and connectives $\rightarrow, \leftrightarrow$, and takes forms like $\forall x(P(x) \rightarrow Q(x))$.

The reasoning is built upon the axiom system regarding logic. The Propositional and Predicate Logics are as follows (See in Table 6.1).

Based on the above logic, several axiom systems have been proposed. For example, Aristotle's ontological law [7] stated a Non-Contradiction Principle:

$$\neg(P \wedge \neg P). \tag{6.2}$$

No proposition P can be simultaneously true and false.

Kurt Gödel's incompleteness theorems [69] fundamentally demonstrate inherent limitations in formal axiomatic systems –sets of rules and symbols used to derive mathematical truths.

6.6.1 A Brief about Kurt Gödel's Incompleteness Theorems

First Incompleteness Theorem: Any consistent formal system powerful enough to express basic arithmetic (like Peano arithmetic) is *incomplete*. This means there will always be true statements expressible within the system that cannot be proven true using the axioms and rules of that system itself. The system contains "gaps" in provability.

Second Incompleteness Theorem: Such a system cannot prove its own consistency using only its own axioms and rules. If it's consistent, the statement asserting "this system is consistent" remains unprovable within the system.

6.6.2 How They Limit Formal Systems

In essence, Gödel [69] proved that any formal system rich enough to handle elementary arithmetic is either incomplete (missing proofs for some truths) or inconsistent (capable of proving contradictions). Furthermore, it can never certify its own soundness. This imposes fundamental, unsurpassable barriers on what any single formal system can achieve in capturing mathematical reality.

6.7 The Summary of Fundamental Reasoning Methodologies

There are three fundamental reasoning methodologies: *deductive reasoning*, *inductive reasoning*, and *abductive reasoning*. Each embodies a different way of moving from what is already known or observed to what is newly inferred.

Deductive reasoning proceeds from the general to the particular. Starting from accepted premises and axioms, it derives conclusions that follow with *logical necessity*: if the premises are true and the inference rules are correctly applied, the conclusion cannot be false. Deduction thus preserves truth by logical form alone, independently of empirical content.

Inductive reasoning proceeds from the particular to the general. It abstracts new propositions or models from finite collections of interrogation outcomes, yielding conclusions that are *probable* rather than certain. Induction is ampliative: the conclusion goes beyond the premises, supporting generalizations that may later be refined or overturned as new evidence arrives.

Abductive reasoning proceeds from interrogation outcomes to *plausible hypothesis*. Abduction does not guarantee truth, but proposes the *best available* hypothesis under current knowledge.

Each of the following sections adheres to a consistent reasoning structure. For every reasoning type, we begin with its *definitional principles*, clarifying its logical foundations and epistemic status; then provide an example to demonstrate its concrete process; follow with *methodological paradigms* that trace its historical and contemporary formalizations.

6.8 Deductive Reasoning

6.8.1 Definitional Principles

Building on the definitions of propositions, premises, and axiom systems introduced earlier, *deductive reasoning* provides the most stringent mechanism for generating new propositions. It proceeds from the general to the particular—deriving specific conclusions from universal premises with logical necessity. Unlike induction, which abstracts generalities from empirical observations, deduction operates entirely within a formal structure of axioms and inference rules. Once the premises are accepted as valid elements of the knowledge system, the resulting conclusion follows with certainty. In this sense, deduction establishes the *architecture of reasoning certainty*: truth is preserved by logical structure rather than experiential variability.

Formally, let $P_1, P_2, \ldots, P_n$ denote a set of *premises* (propositions or models), and let C denote a *conclusion*. The notation

$$P_1, P_2, \ldots, P_n \vdash_{\text{ded}} C, \tag{6.3}$$

indicates that, under the rules of deductive inference, C follows necessarily from the premises $\{P_i\}$. Here, $\vdash_{\text{ded}}$ denotes the *deductive entailment relation*, meaning that the

validity of the inference depends exclusively on logical form rather than empirical content.

A canonical example is the rule of *modus ponens*. Given a conditional proposition $P \to Q$ [1] and the premise P, deductive entailment yields:

$$(P \to Q),\ P \ \vdash_{\text{ded}}\ Q. \tag{6.4}$$

This inferential pattern remains valid regardless of the semantic content of P and Q—its justificatory force derives solely from logical structure.

Deductive systems rest on a small set of axioms and inference rules that generate an infinite set of valid propositions. The most common logical connectives are summarized in Table 6.1. Each connective defines a lawful transformation within propositional or predicate logic. Deductive reasoning, then, is the disciplined application of these transformations to derive conclusions already implicit in the premises.

A typical example of deductive reasoning

Premise 1: All even numbers are divisible by 2.
Premise 2: 14 is an even number.
Deductive conclusion: Therefore, 14 is divisible by 2.

6.8.2 Methodological Paradigms

Deductive reasoning has evolved from classical syllogistic forms to symbolic and computational logic, which now underlie mathematics and artificial intelligence.

Syllogistic Deduction. The earliest deductive framework traces back to *Organon* [6]. A syllogism takes the canonical form:

$$\text{All } A \text{ are } B; \quad \text{C is } A; \quad \therefore \text{C is } B. \tag{6.5}$$

Here, A, B, and C denote *classes of objects*. The statement "All A are B" asserts that every object belonging to class A also belongs to class B, while "C is A" asserts that the object C is a member of class A. Given these premises, the conclusion that C is also a member of class B follows with logical necessity.

This form illustrates that the conclusion introduces no new information beyond what is already contained in the premises. Classical syllogistic inference thereby established the principle that deductive validity depends solely on logical form. This principle later becomes crucial in determining whether propositions and models about objects remain structurally coherent under different interrogation conditions.

Formal and Mathematical Deduction. In modern logic, formal deduction was codified through the axiomatic systems of symbolic logic [64, 82, 165]. These frameworks define rules of inference independent of content or interpretation. For instance:

$$\forall x\ (A(x) \to B(x)),\ A(a)\ \vdash\ B(a). \tag{6.6}$$

[1] $P \to Q$: read as "if P, then Q"

This symbolic precision supports theorem proving and verification across mathematical and computational domains.

Computational Deduction. With the rise of computing, deduction became mechanized through automated reasoning frameworks, including logic programming [105] (e.g., Prolog), resolution-based theorem provers [9, 10], SAT/SMT solvers [8, 70], description logic reasoners [204], and model checkers [63]. These systems perform large-scale consistency checking, proof search, constraint satisfaction, and validation of complex rule sets. In artificial intelligence, computational deduction enables knowledge-base maintenance [211], formal verification of learning systems [50, 147], and the enforcement of safety constraints [112].

6.9 Inductive Reasoning

6.9.1 Definitional Principles

Inductive reasoning allows us to generalize a proposition or model from specific cases (like seeing multiple white swans and generalizing that all swans are white). However, this reasoning doesn't guarantee certainty, as new observations might contradict the generalization. The conclusion is probable, but not logically necessary.

This is different from deductive reasoning, which guarantees that the conclusion is true if the premises are true (e.g., if "all swans are white" and "this is a swan," then the swan must be white). In inductive reasoning, conclusions emerge based on a similarity across multiple interrogations, even though each individual observation might not guarantee the outcome.

Formally, we may write:

$$P_1, P_2, \ldots, P_n \vdash_{\text{ind}} C \quad \text{where } \forall i, P_i \nvdash_{\text{ded}} C. \tag{6.7}$$

That is, the conclusion C does not follow with logical necessity from any single premise P_i, but emerges as a plausible proposition or model across them all.

Three principles underlie valid induction:

1. **Evidence Accumulation.** Confidence in a *hypothesis* H grows as more consistent *evidence* $E_1, E_2, \ldots, E_n$ is observed. Under classical i.i.d. (independent and identically distributed) assumptions, repeated confirmation increases the posterior probability of H:
$$\lim_{n\to\infty} P(H \mid E_1, \ldots, E_n) = 1. \tag{6.8}$$
Thus, reliable induction depends not on any single evidence item E_i, but on the convergence of evidence across repeated observations.

2. **Projective Invariance.** Inductive generalizations must remain stable when applied to different contexts or domains. Let $P(H)$ denote the prior probability of hypothesis H, and $P(H \mid E)$ its posterior probability after observing evidence E. The ratio
$$\frac{P(H \mid E)}{P(\neg H \mid E)} > \frac{P(H)}{P(\neg H)}, \tag{6.9}$$
expresses that evidence E should *increase* the odds in favor of H. A generalization worth preserving in one domain should, when properly abstracted, retain its inductive support across others.

3. **Error Bounding.** Every inductive conclusion involves uncertainty, which must be formally quantified. Let $\hat{\theta}$ denote a sample's estimator for the parameter of a population, $z_{\alpha/2}$ the critical value of a standard normal distribution at significance level α, and n the sample size. A classical α-level confidence interval CI_α takes the form:
$$\mathrm{CI}_\alpha = \hat{\theta} \pm z_{\alpha/2}\sqrt{\frac{\hat{\theta}(1-\hat{\theta})}{n}}. \tag{6.10}$$
Inductive reasoning succeeds not by eliminating error, but by measuring, bounding, and managing it systematically.

A simple example of inductive reasoning

Observation 1: The sun rose in the east yesterday.
Observation 2: The sun rose in the east today.

Inductive conclusion: The sun *will* rise in the east tomorrow.

This conclusion does not follow with logical necessity from any single measurement or testing; Rather, it emerges from recognizing a consistent similarity across multiple observations. Hence, the inference is **probable** rather than **certain**, illustrating the nature of inductive reasoning.

6.9.2 Methodological Paradigms

Inductive reasoning is central to many methodologies, and it manifests across a variety of frameworks, each formalizing the logic of generalization under different epistemic assumptions. These frameworks provide tools for reasoning from observations, updating and fitting models, and making predictions according to a truth, fact, or hypothesis, each with unique strengths and applications. Below, we explore three key paradigms of inductive reasoning: the Bayesian framework, Statistical Learning Theory, and the Algorithmic Probability framework.

Bayesian Framework. The Bayesian framework [15, 92, 127, 86] dynamically updates hypotheses as new evidence arrives. This approach provides a principled mechanism for

updating hypotheses H based on observed evidence E, using the following expression:

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)}. \tag{6.11}$$

As evidence E accumulates, the posterior probability $P(H|E)$ strengthens or weakens accordingly. This framework is foundational to modern inductive reasoning.

Statistical Learning Theory. Within the Statistical Learning Theory framework [209], induction is framed as the process of model fitting and prediction. It formalizes the learning process as the search for a hypothesis function $\hat{f}$ that minimizes the expected loss:

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \lambda R(f). \tag{6.12}$$

Here, y_i represents the true output (or target) value for the i-th observation outcome, while x_i represents the corresponding input (or feature). The term L is the empirical loss function that quantifies the difference between predicted and true values, R is a regularization term that controls the complexity of the model, and λ is the trade-off parameter that balances model fit and generalization.

In this context, learning is an inductive process where the goal is to find the simplest model that best fits the observed outcomes while avoiding overfitting. The learner seeks a function $\hat{f}$ that generalizes well to unseen outcomes, ensuring that the model not only fits the observed outcomes, which is often called *training data*, well, but also maintains simplicity, preventing overfitting.

Algorithmic Probability Framework. At the theoretical frontier lies the Algorithmic Probability framework [180, 88], which defines the idealized limit of all computable inductive systems. This framework assigns a universal prior probability $P_U(x)$ to a data sequence x by summing over all programs p that produce x (or any extension of it) on a universal Turing machine U:

$$P_U(x) = \sum_{p:U(p)=x*} 2^{-\ell(p)}, \tag{6.13}$$

where $\ell(p)$ is the length of program p, and $U(p) = x*$ denotes that the output of p on U has x as its prefix.[2] Shorter programs—simpler explanations—receive exponentially higher prior probability, providing a formal realization of Occam's Razor [148]. Although incomputable in practice, this framework serves as a conceptual ideal that unifies probability, simplicity, and prediction into a single theory of rational inference.

Together, these paradigms reveal the gradient of inductive reasoning—from pragmatic updating (Bayesian) to statistical optimization (Learning) to universal reasoning

[2] Formally, $x*$ represents the set of all strings whose initial segment equals x. Thus, any program whose output begins with x contributes to $P_U(x)$, reflecting the framework's role in sequence prediction.

(Algorithmic). Each paradigm contributes a distinct lens through which to reason from evidence, with applications ranging from hypothesis updating to model fitting and prediction.

6.10 Abductive Reasoning

6.10.1 Definitional Principles

While deduction moves from general rules to necessary conclusions, and induction moves from repeated observations to probable generalizations, *abductive reasoning* proceeds in the opposite direction: it infers the *most plausible* hypothesis that could explain an observed outcome.

First introduced by Peirce—who characterized it as "the logic of discovery" [142]—abduction is the process of forming explanatory hypotheses. It begins with an observation and seeks an explanatory hypothesis. Abduction thus provides the conceptual bridge between observation and explanation. It does not ask "What must be true?" as in deduction, nor "What is probably true?" as in induction, but rather "What would best explain what we observe?"

Formally, let E denote an observed *evidence* and let H denote a candidate *hypothesis*. Abductive reasoning can be expressed as:

$$H \to E, \quad E \text{ observed} \quad \therefore \quad H \text{ (as a plausible hypothesis).} \tag{6.14}$$

This inference is inherently **non-monotonic**: the acceptance of H as a plausible hypothesis can be overturned by new evidence, additional hypotheses, or alternative hypotheses. Unlike deductive entailment, the relation

$$H \vdash_{\text{abd}} E, \tag{6.15}$$

does *not* guarantee that H is true—only that H is a *reasonable* explanation under the current knowledge.

A simple example of abductive reasoning

Observation: The ground is wet this morning.
Possible explanation H_1: It rained last night.
Possible explanation H_2: The sprinkler system ran overnight.

Abductive conclusion: The most plausible explanation is that it rained.

This conclusion may later be revised if new evidence appears (e.g., sprinkler logs). Abduction thus illustrates reasoning that is **explanatory but fallible**.

6.10.2 Methodological Paradigms

Abductive reasoning plays a foundational role in scientific modeling, diagnosis, hypothesis formation, and mechanism discovery. Its methodological paradigms include:

Inference to the Best Explanation. Inference to the Best Explanation [115] formalizes abduction as selecting the hypothesis that best explains the observed evidence, based on certain explanatory virtues. These virtues include criteria such as simplicity (the hypothesis is not overly complex), coherence (the hypothesis fits well with established knowledge), breadth (the hypothesis explains a wide range of phenomena), and explanatory power (the hypothesis accounts for the evidence in a compelling way).

Given multiple candidate hypotheses $\{H_1, H_2, \dots\}$, each of which could explain the evidence E, the abductive task is to select the one that maximizes an explanatory score:

$$H^* = \arg\max_{H_i} \text{Explains}(H_i, E), \tag{6.16}$$

where $\text{Explains}(H_i, E)$ quantifies how well hypothesis H_i explains E. This framework is qualitative, relying on reasoning about the explanatory virtues of each hypothesis.

Probabilistic Abduction. In contrast, probabilistic abduction takes a quantitative approach, viewing abduction through the lens of Bayesian inference. Here, abduction corresponds to selecting the hypothesis H that maximizes its posterior probability, given the observed evidence E:

$$H^* = \arg\max_{H} P(H \mid E), \tag{6.17}$$

where $P(H \mid E)$ is the posterior probability of the hypothesis H after observing evidence E. This approach provides a formal way of evaluating hypotheses, integrating prior knowledge about the hypotheses with the likelihood of the evidence under each hypothesis. Probabilistic abduction thus makes the reasoning process more rigorous and quantifiable by combining both prior knowledge and empirical data.

Model-based Abduction. Widely used in diagnosis and scientific modeling, this approach reasons over structured models (causal graphs [144], knowledge bases [121], or generative mechanisms [19]). The hypothesis H is considered plausible if incorporating it into the model makes the observed evidence E highly likely.

Together, these paradigms reflect the nature of abduction as a search for explanatory adequacy—a reasoning process that is creative, defeasible, and indispensable for forming new hypotheses.

6.11 Summary

Built upon axioms and inference rules, reasoning allows subjects to derive new propositions or models from existing truths, revealing the internal logic that governs both

knowledge systems. Deductive reasoning ensures internal validity by deriving necessary conclusions from established premises; inductive reasoning generalizes from repeated evidence to formulate empirical laws; and abductive reasoning infers the most plausible causes behind observed effects, driving discovery and interpretation. Together, these three modes form a continuous epistemic cycle—abduction proposes, induction confirms, and deduction secures coherence.

Chapter 7

Interrelationships Among Three Interrogations

In this chapter, I first posit the existence of a primitive interrogation among the three fundamental interrogations, and then proceed to examine the relationships between these three fundamental interrogations.

7.1 Primitive Interrogation: Comparison with a Reference

In Chapters 4, 5, and 6, I define measurement, testing, and reasoning as follows, respectively.

Measurement is a capacity, capability, and process that attributes values to a quantity of an object by comparing its quantity with the unit quantity of a reference object under an interrogation condition.

Testing is a capacity, capability, and verification process of running test cases to determine whether a proposition or a model of an object conforms to the test oracle through comparing their outcomes [224, 223].

Reasoning or inference is a capacity, capability, and mental process that generates new propositions or models about an object based on the premises, the facts or truths, and the interrogation outcomes.

Among the three fundamental interrogations, we could find a primitive interrogation, which is the comparison of an object with a reference object. Comparison is employed to establish the order among the same quantities of different objects. For measurement, the subject compares a quantity of any object with a unit of measurement of a reference object. For testing, the subject runs test cases and compares the outcomes with those of a test oracle. Hereby, a test oracle defines a reference. Reasoning relies upon reasoning rules. These rules can only be validated through testing, which requires a reference (test oracle) for comparison.

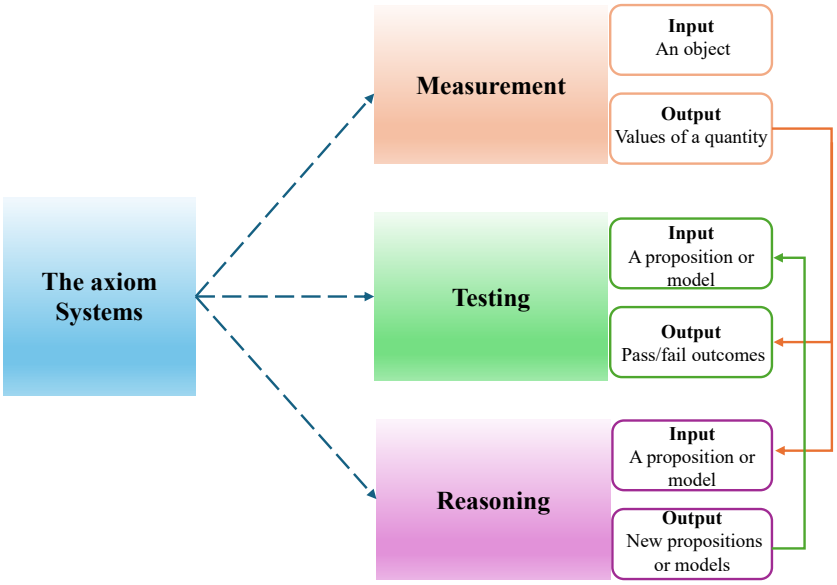

Figure 7.1: The relationship among three interrogations.

7.2 Relationships among Three Interrogations

The relationships among the three interrogations are shown in Figure 7.1 [1].

When testing is performed on a hypothesis about an object, it is essential to compare the outcome of running a test case to the test oracle. Most comparisons involve measured quantity values, which are inherently derived from measurement. That is to say, testing often relies on quantitative data obtained through measurement. The input for reasoning comprises propositions and models of objects, which can only be derived from quantitative data obtained through measurement. However, I believe many properties are not countable.

Measurement, testing, and reasoning rely upon an axiom system, which is one of the central topics in reasoning. For example, measurement relies upon an assumption that a property of an ideal reference object could be utilized to define the unit of measurement. However, the axiom system in measurement can only be validated by testing. Over the course of metrological evolution, numerous such ideal reference objects have been adopted and subsequently rendered obsolete following rigorous testing. Meanwhile, the inputs, outputs, intermediates, and even the reasoning rules in reasoning can only be validated through testing.

[1] Dr. Lei Wang contributed this figure based on the discussion with me.

7.3 Summary

This chapter presents that three fundamental interrogations are built on a primitive interrogation: comparison. I also clarify the interrelationships among measurement, testing, and reasoning.