性能评价

郭健美

2024年秋

SOLE 系统优化实验室
华东师范大学

系统化的性能评价过程

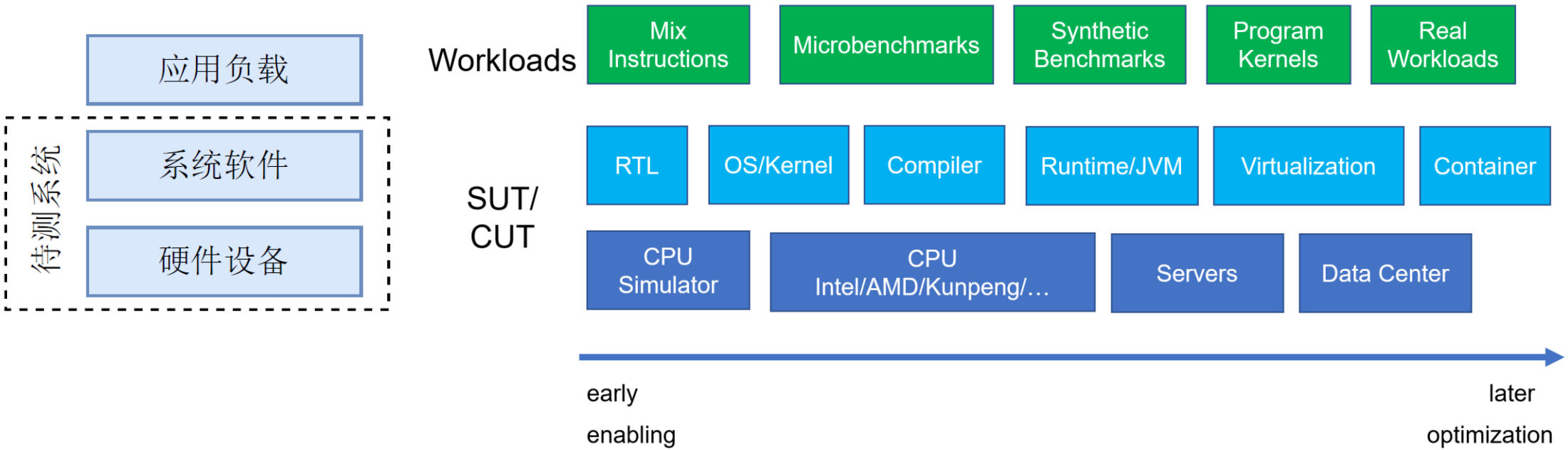
1. 设定评价目标（包括待测系统、服务和产出）；
2. 选择评价方法；
3. 选择评价指标；
4. 选择工作负载；
5. 实验设计；
6. 分析与解释数据；
7. 展示结果；
8. 返回步骤 1 （可以根据结果重新设定评价目标）。

内容

- 评价目标的设定
- 评价方法的选择
- 评价指标的选择
- 数据的分析与解释
- 常见错误与规避

评价目标的设定

- 明确待测系统（System Under Test, SUT）或待测组件（Component Under Test, CUT）的边界
- 确定用户、服务、结果
- 目标明确会避免陷入大量投入但收效甚微的窘境

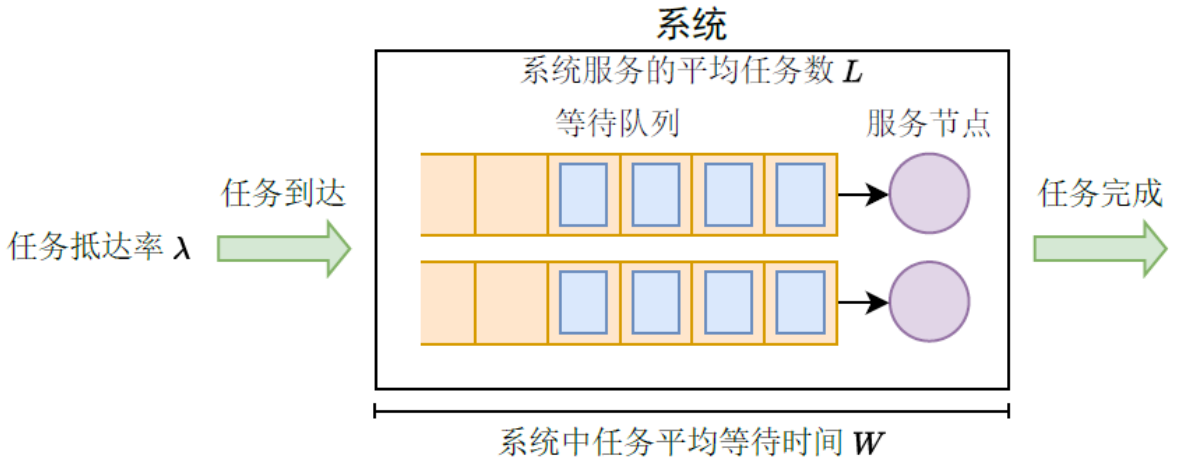


评价方法的分类

- 测量：在真实系统上进行性能测量以获得评价指标
- 仿真：在仿真器中模拟待测系统的行为以获得评价指标
- 分析建模：建立待测系统的数学模型或行为模型，进而推导计算评价指标

利特尔法则

$$L = \lambda W$$



评价方法的选择

选择条件	测量	仿真	分析建模
1. 适用阶段	存在原型系统时	任何阶段	任何阶段
2. 需要的时间	可多可少	中等	少
3. 需要的工具	测量工具	仿真器	分析建模的技能
4. 准确性	可高可低	中等	低
5. 综合成本	高	中等	低

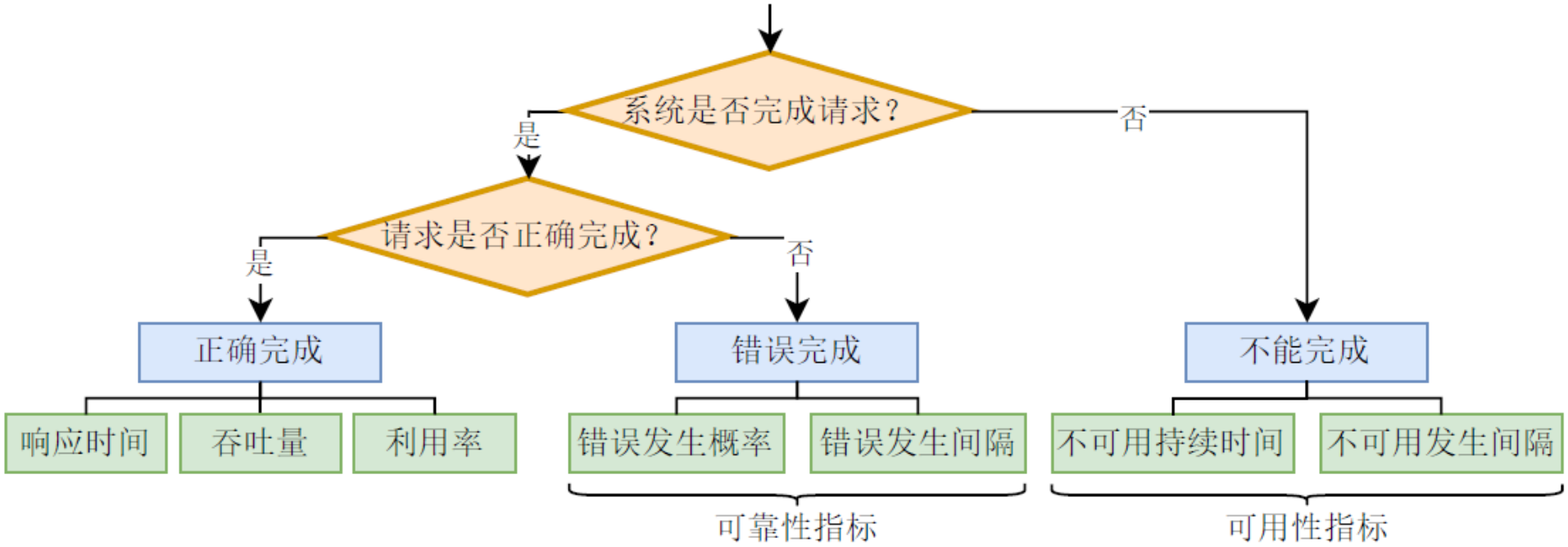
评价方法的选择

	优势	劣势	挑战
测量	提供真实系统的数据 能够测试压力极限	要求系统应能正常工作 难以得到因果关系	定义适当的评价指标 使用适合的工作负载
仿真	相较于制作原型系统进行测量，成本较低 能够测试各种压力场景	仿真并非真实系统 无法对预期性能提供保证	正确地模拟真实系统 正确地使用模拟器
分析建模	可以洞察因果关系，并对预期行为提供保证 无需构建原型系统	预测结果完全依赖于建模的好坏 依赖专业知识和经验	正确地构建模型 长期积累专业技能

实际中，可以结合使用、交叉验证

评价指标的分类

- 响应时间：完成一次服务请求所需的时间
- 吞吐量：完成服务请求的速率
- 利用率：完成服务请求时系统消耗的CPU、内存、存储、网络资源的比例



评价指标的选择

- 低波动
  - 有助于获得更稳定的结果，并减少重复实验次数
- 不冗余
  - 尽量减少冗余的性能指标，使结果更加简洁清晰
- 完整性
  - 应完整覆盖系统不同方面的性能

评价指标的验证

- 量纲分析（dimensional analysis）
  - 指物理学或工程学中使用物理量的量纲（例如，长度、质量、时间等）来分析或检查几个物理量之间的关系。在性能评价中，量纲分析可用于检查性能指标是否具有合理的量纲（包括无量纲），从而判定所采用或设计的指标是否具有物理意义或是否合理。
- 合理性检查（sanity check）
  - 指检查性能指标是否处于可接受的范围内，从而确保系统运行及其性能指标在一个基本水平上是合理的。

数据的分析与解释

- 数据的汇总
- 数据的比较

数据的汇总

- 集中趋势（central tendency）
- 离散程度（dispersion）

数据的集中趋势

- 即计算数据的平均，描述数据的集中位置或平均位置
  - 算术平均数

$$\bar{x}_{\mathrm{a}} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

  - 调和平均数

$$\bar{x}_{\mathrm{h}} = \frac{n}{\sum_{i=1}^{n} 1/x_i}$$

  - 几何平均数

$$\bar{x}_{\mathrm{g}} = \sqrt[n]{\prod_{i=1}^{n} x_i}$$

误用平均数的陷阱

- 算术平均数适合于单一物理量或单变量数据的汇总，例如，时间、长度、重量、温度等。
- 以程序的运行时间 T 为例，平均运行时间 $\bar{T}_{\mathrm{a}} = \frac{1}{n}\sum_{i=1}^{n} T_i$ 与总运行时间 $\sum_{i=1}^{n} T_i$ 成正比，符合直观理解。
- 如果对运行时间求调和平均数，则平均运行时间 $\bar{T}_{\mathrm{h}} = \frac{n}{\sum_{i=1}^{n} \frac{1}{T_i}}$ 与 $\sum_{i=1}^{n} \frac{1}{T_i}$ 成反比，不符合直观理解。
- 如果对吞吐量 R=W/T 这类比值型物理量求算术平均数，则平均吞吐量 $\bar{R}_{\mathrm{a}} = \frac{1}{n}\sum_{i=1}^{n} R_i = \frac{1}{n}\sum_{i=1}^{n} \frac{W}{T_i} = \frac{W}{n}\sum_{i=1}^{n} \frac{1}{T_i}$ 与 $\sum_{i=1}^{n} \frac{1}{T_i}$ 成正比，不符合直观理解。
- 如果对吞吐量求调和平均数，则平均吞吐量 $\bar{R}_{\mathrm{h}} = \frac{n}{\sum_{i=1}^{n} \frac{1}{R_i}} = \frac{n}{\sum_{i=1}^{n} \frac{T_i}{W}} = \frac{nW}{\sum_{i=1}^{n} T_i}$ 与 $\sum_{i=1}^{n} T_i$ 成反比，符合直观理解。

两种容易混淆的比值型指标

- 比率（rate）
  - 是两个不同量纲的物理量的比值，得到的是一个有量纲的指标。例如，速度（量纲：公里/小时）是距离（量纲：公里）和时间（量纲：小时）的比值。
- 比例（ratio）
  - 是两个相同量纲的物理量的比值，或两个无量纲的数字的比值，得到的是一个无量纲的指标。例如，比例尺是图上距离（量纲：米）与实际距离（量纲：米）的比值，加速比是旧机器运行时间（量纲：秒）与新机器运行时间（量纲：秒）的比值。

离散程度

指标	定义	特点
极差	$r = x_{\max} - x_{\min}$	体现指标波动的最大范围，对极端值或异常值过于敏感。
样本方差	$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})^2$	最常用的离散程度指标，样本方差的量纲与原始指标或平均值的量纲不一致，而采用样本标准差可以解决这个问题。
样本标准差	$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})^2}$	
变异系数	$\mathrm{CV} = s/\bar{x}$	消除了特定量纲问题，提供了一个无量纲的比值。

数据的比较

- 加速比的比较和汇总
- 置信区间的比较*

如何表述“方案 X 比方案 Y 快” “快多少”？

如何表述“方案 X 比方案 Y 快” “快多少”？

- 运行时间的加速比（无量纲）

$$\text{speedup} = \frac{T_{\text{Y}}}{T_{\text{X}}} = k$$

- 表示“Y 的运行时间是 X 的 k 倍”或“X 的速度是 Y 的 k 倍”

- 吞吐量的加速比

$$\text{speedup} = \frac{T_{\text{Y}}}{T_{\text{X}}} = \frac{\frac{W}{R_{\text{Y}}}}{\frac{W}{R_{\text{X}}}} = \frac{R_{\text{X}}}{R_{\text{Y}}} = k$$

- 表示“X 的吞吐量是 Y 的 k 倍”

加速比的汇总和比较：算术平均数

实验	T_{X}	T_{Y}	$T_{\mathrm{X}}/T_{\mathrm{Y}}$
1	9.00	3.00	3.00
2	8.00	2.00	4.00
3	2.00	20.00	0.10
4	10.00	2.00	5.00
平均数	(a) 7.25	(a) 6.75	(a) 3.03

有问题么？

加速比的汇总和比较：算术平均数

实验	T_X	T_Y	T_X/T_Y
1	9.00	3.00	3.00
2	8.00	2.00	4.00
3	2.00	20.00	0.10
4	10.00	2.00	5.00
平均数	(a) 7.25	(a) 6.75	(a) 3.03

有问题么？

$7.25/6.75 \approx 1.07$

加速比的算术平均数不等于
算术平均数的加速比!

加速比的汇总和比较：算术平均数

实验	T_X	T_Y	T_X/T_Y	T_Y/T_X
1	9.00	3.00	3.00	0.33
2	8.00	2.00	4.00	0.25
3	2.00	20.00	0.10	10.00
4	10.00	2.00	5.00	0.20
平均数	(a) 7.25	(a) 6.75	(a) 3.03	(a) 2.70

问题?

产生了矛盾!

规则：比值 A/B 的平均数应该是比值 B/A 的平均数的倒数。

加速比的汇总和比较：调和平均数

实验	T_X	T_Y	T_X/T_Y	T_Y/T_X
1	9.00	3.00	3.00	0.33
2	8.00	2.00	4.00	0.25
3	2.00	20.00	0.10	10.00
4	10.00	2.00	5.00	0.20
平均数	(a) 7.25	(a) 6.75	(h) 0.37	(h) 0.33

也产生了矛盾！

加速比的汇总和比较：几何平均数

实验	T_X	T_Y	T_X/T_Y	T_Y/T_X
1	9.00	3.00	3.00	0.33
2	8.00	2.00	4.00	0.25
3	2.00	20.00	0.10	10.00
4	10.00	2.00	5.00	0.20
平均数	(a) 7.25	(a) 6.75	(g) 1.57	(h) 0.64

$$\overline{\left(\frac{T_Y}{T_X}\right)}_g = 0.64 \approx \frac{1}{1.57} = \frac{1}{\overline{\left(\frac{T_X}{T_Y}\right)}_g}$$

重要性质：几何平均数的比值与比值的几何平均数是相等的。

多个系统的性能比较

实验	T_X	T_Y	T_Z	T_Y/T_X	T_Z/T_X	T_Z/T_Y
1	9.00	3.00	4.00	0.33	0.44	1.33
2	8.00	2.00	4.00	0.25	0.50	2.00
3	2.00	20.00	1.00	10.00	0.50	0.05
4	10.00	2.00	3.00	0.20	0.30	1.50
平均数	(a) 7.25	(a) 6.75	(a) 3.00	(g) 0.64	(g) 0.43	(g) 0.67

$$\overline{\left(\frac{T_Z}{T_X}\right)}_g \Big/ \overline{\left(\frac{T_Y}{T_X}\right)}_g = \overline{\left(\frac{T_Z}{T_Y}\right)}_g = 0.43/0.64 \approx 0.67$$

以 X 为参照系统，分别计算 Y 和 Z 相对于 X 的加速比，所得两个加速比是可比的，可直接计算 Y 与 Z 的加速比。

SPEC ratios (SPECspeed & SPECrate)

SPEC ratios normalize the execution times to a reference computer by dividing the time on the reference computer by the time on the computer being rated, yielding a ratio proportional to performance.

Benchmarks	Sun Ultra Enterprise 2 time (seconds)	AMD A10-6800K time (seconds)	SPEC 2006Cint ratio	Intel Xeon E5-2690 time (seconds)	SPEC 2006Cint ratio	AMD/Intel times (seconds)	Intel/AMD SPEC ratios
perlbench	9770	401	24.36	261	37.43	1.54	1.54
bzip2	9650	505	19.11	422	22.87	1.20	1.20
gcc	8050	490	16.43	227	35.46	2.16	2.16
mcf	9120	249	36.63	153	59.61	1.63	1.63
gobmk	10,490	418	25.10	382	27.46	1.09	1.09
hmmer	9330	182	51.26	120	77.75	1.52	1.52
sjeng	12,100	517	23.40	383	31.59	1.35	1.35
libquantum	20,720	84	246.08	3	7295.77	29.65	29.65
h264ref	22,130	611	36.22	425	52.07	1.44	1.44
omnetpp	6250	313	19.97	153	40.85	2.05	2.05
astar	7020	303	23.17	209	33.59	1.45	1.45
xalancbmk	6900	215	32.09	98	70.41	2.19	2.19
Geometric mean			31.91		63.72	2.00	2.00

Figure 1.19 SPEC2006Cint execution times (in seconds) for the Sun Ultra 5—the reference computer of SPEC2006—and execution times and SPECRatios for the AMD A10 and Intel Xeon E5-2690. The final two columns show the ratios of execution times and SPEC ratios. This figure demonstrates the irrelevance of the reference computer in relative performance. The ratio of the execution times is identical to the ratio of the SPEC ratios, and the ratio of the geometric means (63.7231.91/20.86 = 2.00) is identical to the geometric mean of the ratios (2.00). Section 1.11 discusses libquantum, whose performance is orders of magnitude higher than the other SPEC benchmarks.

[John L. Hennessy, David A. Patterson: Computer Architecture - A Quantitative Approach, 6th Edition. Morgan Kaufmann 2017.]

常见错误与规避

可能存在误导性的描述	批判性分析	规避方法
该程序已运行了十亿次循环。	是不是程序真正地在待测机器上执行了十亿次循环？部分编译器在代码生成阶段通常会执行“死码删除(dead code elimination)”，即删去对程序输出无影响的代码，实际可能没有执行那么多次循环。	在循环执行的过程中输出某些内容进行验证。
程序输出了结果，但是没有校验，因为浮点数结果有微小差异是可以预期的。	即便是微小的浮点数差异，也有可能是执行时指令路径不同导致的，如果该程序是由于异常而退出的，程序并没有完成预期的工作量。	应当校验程序的结果，并且规定结果容许的误差范围。
基准评测程序已经是预编译好的，下载之后即可直接运行，不需要在待测系统上进行编译。	可能无法比较新硬件、新操作系统、新编译器对性能的影响。	基准评测应当提供源代码，以拓宽待测系统的范围，并且应当提前在多种编译器与操作系统下进行测试。
基准评测程序测量了某方面的性能。	测量结果可能包含了程序启动的部分，如果这部分占据主导地位，那么可能会产生误导性的结果。	检查程序的性能剖析数据，检查真正被测量的是什么。
在一个知名的基准评测程序上进行了微小的改造，得到了一个新的基准评测程序。	是否有改造的确切记录？这样的改造是否破坏了可比较性？	应当由第三方进行检查。
基准评测没有定义运行规则，因为如何正确运行该基准评测程序“显而易见”。	尽管可能运行程序的正确方法是“显而易见的”，但是仍有可能出现问题。即便是一个微小的改动也可能得到完全不同的结果。	为了保证基准评测结果能够进行有意义的比较，需要定义明确的运行规则。
该基准评测是代表某一应用的一系列的低级操作。	如何说明这些操作的代表性？	基准评测程序应当优先从真实应用抽取。

批判性思维 Critical Thinking

- 明确定义和识别问题：充分理解需求，提出关键问题，挖掘隐含的假设，以及识别关联问题等。
- 收集可靠的数据：寻找可信的数据源，采用可靠的方法，理清数据关联并交叉验证，保障数据可靠性。
- 合理的分析和推理：综合分析不同的观点和解释，评估优点和缺点，使用归纳推理（从特定观察中得出一般性结论）和演绎推理（从一般原则得出特定结论）等进行合理的逻辑推理。
- 切实解决问题：复杂问题的求解通常都涉及优劣取舍，在充分分析后，最后还是需要得到切实可行的解决方案，并做出恰当的决策。
- 清晰有效的沟通：能清晰有效地将分析结果传递给相关人员也是关键，要能明确表达自己的观点和推理，与相关人员开展有意义的讨论和辩论。

小结

- 性能评价应当遵循系统化的过程
- 评价方法应根据需要进行选择，并结合使用、交叉验证
- 选择和设计合适的评价指标，通过量纲分析和合理性检查进行验证
- 数据的汇总和比较应选择合适的统计量，防止常见的数据统计陷阱
- 加强批判性思维能力的锻炼和培养，避免误导性的分析和评价