**1. 算法模型**

**（1）概述**：  
算法模型通过数学公式和固定参数来估算工作量，例如使用项目规模（如代码行数）和生产率计算总工作量。

**（2）代表模型**：  
**COCOMO（Constructive Cost Model）**

**（3）应用场景**：

* 适用于有清晰历史数据的项目。
* 对项目规模变化敏感的大型工程。

**（4）优点**：

* 定量化评估，提供明确的估算公式。
* 能考虑项目规模对工作量的非线性影响。

**（5）缺点**：

* 参数调整困难，需有准确历史数据支持。
* 初期需求模糊时，估算可能偏差大。

**（6）变体**：  
**COCOMO II** 提供三阶段模型：

* **应用构成阶段**：通过对象点计算。
* **早期设计阶段**：通过功能点估算。
* **后期构造阶段**：基于实际代码规模估算。

**2. 专家判断**

**（1）概述**：  
依赖领域专家对项目需求、规模和复杂度的经验判断。

**（2）方法**：

* **Delphi法**：多位专家匿名独立估算并反复讨论，逐步达成共识。
* **Wideband Delphi**：改进版的 Delphi，增加了讨论和信息共享环节。

**（3）应用场景**：

* 初期需求不明确。
* 使用新技术、无历史数据的项目。

**（4）优点**：

* 快速且灵活。
* 可结合团队对特定领域的理解。

**（5）缺点**：

* 结果易受专家主观性和偏见影响。
* 无法提供数学证明，难以量化。

**3. 类比估计**

**(1) 概述**：  
通过寻找与目标项目类似的已完成项目，对工作量进行推断。

**(2) 方法**：  
使用数学工具（如欧几里得距离公式）计算目标项目与参考项目的相似性，评估工作量。

**(3) 应用场景**：

* 有相似项目历史记录。
* 项目需求明确但需快速估算。

**(4) 优点**：

* 简单直观。
* 适用于快速估算。

**(5 )缺点**：

* 依赖参考项目的质量和准确性。
* 缺乏类似项目时难以适用。

**4. 功能点方法**

**(1) 概述**：  
以软件的功能为基础，量化外部输入、输出、查询、逻辑文件和接口文件来计算工作量。

**(2) 流程**：

* 计算五种基本信息量：外部输入（EI）、外部输出（EO）、外部查询（EQ）、内部逻辑文件（ILF）、外部接口文件（EIF）。
* 计算复杂度调整因子（如安全性、接口特性）。
* 将功能点（FP）转换为代码规模（KLOC），估算工作量。

**(3) 扩展**：

* **IFPUG**：传统信息系统。
* **COSMIC**：针对嵌入式系统或实时系统，使用“数据移动”来计算功能点。

**(4) 优点**：

* 平台无关性，与开发人员无关。
* 高度客观，系统间结果具有可比性。

**(5) 缺点**：

* 初期分析复杂度高。
* 对需求拆分的准确性要求高。

**5. 对象点法**

**(1) 概述**：  
基于需要处理的对象（如屏幕、报告、部件）的复杂性来估算。

**(2) 流程**：

* 确定对象的类型（屏幕、报告等）。
* 按复杂度分为简单、中等、复杂三类。
* 排除现有对象，只计算新增或修改的对象。

**(3) 公式**：

NOP=新增对象点数量

**(4) 应用场景**：

* 面向对象开发项目。
* 快速估算新系统或组件。

**(5) 优点**：

* 简单易用。
* 不依赖详细的需求描述。

**(6) 缺点**：

* 不适用于行为复杂的系统。
* 对功能特性考虑不足。

**6. 自顶向下与由低向上估算**

**(1)概述**：

* **自顶向下**：从全局视角估算整个项目的工作量，再逐层细化。
* **由低向上**：将项目分解为任务，对每个任务逐一估算后汇总。

**(2) 应用场景**：

* **自顶向下**：适用于初期规划阶段。
* **由低向上**：适用于详细规划阶段。

**(3) 优点**：

* 灵活性强。
* 易于与其他方法结合。

**(4) 缺点**：

* 自顶向下可能忽略细节。
* 由低向上对任务分解的质量依赖高。

**7. 参数化模型**

**(1) 概述**：  
基于参数化公式和统计数据，结合规模和生产率计算。

**(2) 公式**：

Effort=A×Size^B×EM

**A**：常数。

**Size**：项目规模。

**B**：规模因子。

**EM**：乘法调整因子。

**(3) 特点**：

* 提供详细的参数配置。
* 能处理复杂系统和多阶段开发需求。

**(4) 优点**：

* 支持定量化调整。
* 可适应不同规模的项目。

**(5) 缺点**：

* 参数获取困难。
* 复杂度较高。

**方法比较总结**

| **方法** | **数据依赖** | **精确性** | **适用场景** | **主观性** |
| --- | --- | --- | --- | --- |
| 算法模型 | 高 | 高 | 有历史数据的大项目 | 低 |
| 专家判断 | 无 | 中 | 新技术、新领域项目 | 高 |
| 类比估算 | 中 | 中 | 类似项目参考 | 中 |
| 功能点方法 | 中 | 高 | 信息系统或嵌入式 | 低 |
| 对象点法 | 低 | 中 | 面向对象开发 | 低 |
| 自顶向下/低向上 | 低 | 中 | 项目初期/后期 | 中 |
| 参数化模型 | 高 | 高 | 复杂系统、渐进估算 | 低 |

结合实际需求和数据可用性，可选择一种或多种方法综合应用，以达到更高的估算准确性和实用性。