

基于灰色关联分析的软件工作量估算方法

付雅芳^{1,2}, 杨任农¹, 刘晓东¹, 王琳³

(1. 空军工程大学航空航天工程学院, 陕西 西安 710038; 2. 中国人民解放军 93792 部队, 河北 廊坊 065000; 3. 总参陆航研究所, 北京 101121)

摘要: 针对历史数据少、信息贫乏的软件估算问题, 提出了一种基于灰色关联分析的工作量估算方法。首先结合灰色关联理论分析了项目特征与工作量的关联度, 然后利用回归技术选取最优特征集; 在此基础上, 计算新项目和历史项目的关联度, 并根据关联度大的项目估算软件工作量; 最后通过 4 个典型的软件数据集对估算方法的性能进行分析。实验结果表明, 该方法能够在历史数据少的情况下准确估算出软件工作量, 其性能优于基于回归分析、BP 神经网络和类比估算等方法; 而选取最优特征后的灰色关联方法, 由于剔除了与工作量相关程度低的特征因子, 进一步减小了工作量估算的平均误差率。

关键词: 灰色关联分析; 软件工作量; 最优特征集; 平均误差率

中图分类号: TP 311

文献标志码: A

DOI:10.3969/j.issn.1001-506X.2012.11.34

Software effort estimation method based on grey relational analysis

FU Ya-fang^{1,2}, YANG Ren-nong¹, LIU Xiao-dong¹, WANG Lin³

(1. The Aerospace Engineering College, Air Force Engineering University, Xi'an 710038, China;

2. Unit 93792 of PLA, Langfang 065000, China;

3. The Army Aviation Institute, the Headquarters of the General Staff, Beijing 101121, China)

Abstract: Under some small history data condition, a grey relational analysis method for software effort estimation is proposed. Firstly, the relationship grade between features and effort is analyzed according to the grey relational analysis theory. At the same time, the optimal feature set is chose by regression technique. On the basis of which, the grey relationship grades between new software and history dataset are computed, and some high relationship grade software data are used for estimating the effort of new software. Finally, four history datasets are used for analyzing the performance of the estimation method. The results of experiment show that the proposed method could precisely estimate software effort, and the method is superior to the linear regression, back propagation network, and analogy method. After some low relationship grade features are deleted from the optimal feature set, the mean magnitude of relative errors is further reduced.

Keywords: grey relational analysis; software effort; optimal feature set; mean magnitude of relative error

0 引言

软件工作量估算是软件完成所需的人力资源成本(工作量), 一般以人时或人日表示进行预先估计^[1]。软件工作量估算费用估算的基础, 它不仅关系到项目实施的成本和计划, 而且也对项目的管理控制起到重要作用。

如何进行精确可靠的工作量估算一直以来都是学术界和企业界普遍关注的问题。目前, 软件工作量估算方法主要分为 3 类: ① 基于回归模型; ② 基于专家经验; ③ 基于既有理论。基于回归模型的方法需要大量的与当前项目相关的历史数据的支持, 它通过对样本数据集进行回归分析,

得到项目特征与工作量的数学关系^[2]。根据问题的需要, 可以向模型添加影响工作量的驱动因子, 以提高其估算精度。比较典型的回归模型有 COCOMO、SLIM、PRICE-S 等^[3-4]。基于专家经验的方法是依赖专家知识及其主观判断来预测工作量, 包括类比法、自顶向下方法、自底向上方法和 Delphi 方法等^[5], 此类方法可以在不具备领域知识的前提下, 利用历史项目经验来确定目标项目的预测值, 但估算精度相对较低。基于既有理论的方法是指将其他领域的估算理论应用于工作量估算, 如利用雷利方程、贝叶斯网络^[6]、神经网络^[7]等估算工作量。其估算效果取决于是否能够充分将既有方法的特点和工作量估算的需要相结合。

收稿日期: 2012-01-19; 修回日期: 2012-05-08。

基金项目: 国家自然科学基金(60904089); 航空科学基金(20095196012)资助课题

作者简介: 付雅芳(1982-), 女, 工程师, 博士, 主要研究方向为软件费用分析、装备发展战略与管理决策。E-mail: huokong12@163.com

上述方法主要适用于有大量历史数据,且数据分布规律较好的情形,但在实际的软件估算过程中,经常遇到的是历史数据少且信息贫乏的问题,此时如果利用传统方法进行估算,将会产生较大误差,其估算精度无法满足实际需要。

针对上述问题,本文提出了一种基于灰色关联分析的软件工作量估算方法。该方法首先分析了项目特征与工作量的灰色关联度,并采用回归技术选取最优特征集,然后计算新项目和历史项目的关联度,并利用关联度大的项目来估算软件工作量,最后通过 4 个典型的软件数据集对提出方法的性能进行了分析。实验结果表明,该方法能够在历史数据少的情况下准确估算出软件工作量,其估算性能优于基于回归分析、BP 神经网络和类比估算等方法。

1 灰色关联分析

1.1 灰色关联分析概述

灰色关联分析(grey relational analysis, GRA)是灰色系统理论的重要组成部分,是挖掘数据内部规律的有效方法,自邓聚龙教授提出以来,已经在经济、管理等领域得到广泛应用。灰色关联分析基于灰色关联度,它通过对数据序列几何关系和曲线几何形状的相似程度进行比较,来分析系统各因素之间的关联程度^[8]。序列曲线之间的几何形状越接近,它们之间的关联度就越大。

灰色关联分析是对运行机制与物理原型不清晰或者根本缺乏物理原型的灰关系序列化、模式化,进而建立灰关联分析模型,使灰关系量化、序化、显化,为复杂系统的建模提供重要的技术分析手段。在灰色关联分析中,各因素关联度数值的实际意义并不大,它只是反映因素间密切程度的相对大小。将多个比较序列对参考序列的关联度按大小顺序排列起来,便组成灰色关联序,关联序可以直接反映各个比较序列对于同一参考序列的“优劣”或“主次”关系。

1.2 灰色关联公理

令 $@_{GRF}$ 为灰关联因子集, Δ_{GR} 为灰关联差异信息空间 $@_{GRF} \Rightarrow \Delta_{GR}$, 其中

$$\Delta_{GR} = \{\Delta, \xi, \Delta_{\max}(\max), \Delta_{\min}(\min)\} \quad (1)$$

$$\Delta_{\min}(k) = |x_0(k) - x_i(k)|, k \in K = \{1, 2, \dots, n\} \quad (2)$$

令 $\gamma(x_0(k), x_i(k))$ 为 Δ_{GR} 上 x_i 与 x_0 在第 k 点比较测度, x_0 为参考序列, x_i 为比较序列, $\gamma(x_0, x_i)$ 为 $\gamma(x_0(k), x_i(k))$ 在 $k \in K$ 上的平均值。

若满足以下公理^[9]:

(1) 规范性

$$0 \leq \gamma(x_i(k), x_j(k)) \leq 1, \forall i, \forall j, \forall k$$

$$\gamma(x_i, x_j) = 1 \Leftrightarrow x_i \equiv x_j, \text{ or } x_0 \text{ 与 } x_i \text{ 同构}$$

$$\gamma(x_i, x_j) = 0 \Leftrightarrow x_i \cap x_j \in \emptyset (\emptyset \text{ 为空集})$$

(2) 偶对称性

$$\forall x_i, \forall x_j \in X,$$

$$\gamma(x_i, x_j) = \gamma(x_j, x_i) \Leftrightarrow X = \{x_i, x_j\}$$

(3) 整体性

$$\forall x_i, \forall x_j \in X = \{x_\sigma \mid \sigma = 0, 1, \dots, n\}, n \geq 2,$$

$$\gamma(x_i, x_j) \stackrel{\text{often}}{\neq} \gamma(x_j, x_i)$$

(4) 接近性

差异信息 $\Delta_{\min}(k) = |x_i(k) - x_j(k)|$ 越小,则 $\gamma(x_0(k), x_i(k))$ 越大,记为 $\Delta_{\min}(k) \downarrow = \gamma(x_0(k), x_i(k)) \uparrow$ 。

满足上述公理时,则称 $\gamma(x_0(k), x_i(k))$ 为 k 点 x_i 对于 x_0 的灰关联系数,称 $\gamma(x_0, x_i)$ 为 x_i 对于 x_0 的灰关联度, γ 为灰关联映射。

$$\gamma(x_0, x_i) = \frac{1}{n} \sum_{k=1}^n \gamma(x_0(k), x_i(k)) \quad (3)$$

1.3 灰色关联分析步骤

灰色关联分析作为一种因素分析方法,它采用量化方式来获得灰色关联度,并以此区分系统变量之间关系的密切程度。灰色关联分析的步骤如下:

步骤 1 确定比较数列和参考数列。设评价对象为 m 个,评价指标为 n 个。

比较数列(即评价对象)为

$$x_i = \{x_i(k) \mid k = 1, 2, \dots, n\}, i = 1, 2, \dots, m$$

参考序列(即评价标准)为

$$x_0 = \{x_0(k) \mid k = 1, 2, \dots, n\}$$

式中, $x_0(k)$ 为第 k 个指标的最优值。

步骤 2 计算灰色关联系数。关联度是反映曲线间几何形状的差别,在灰色关联分析中,常用的关联度有邓氏关联度、绝对关联度、相对关联度、点关联度、斜率关联度等。这里采用邓氏关联度^[9],具体形式如下:

$$\gamma(x_0(k), x_i(k)) = \frac{\Delta_{\min} + \rho \Delta_{\max}}{\Delta_{ik} + \rho \Delta_{\max}} \quad (4)$$

$$\Delta_{\min} = \min_i \min_k |x_0(k) - x_i(k)| \quad (5)$$

$$\Delta_{\max} = \max_i \max_k |x_0(k) - x_i(k)| \quad (6)$$

$$\Delta_{ik} = |x_0(k) - x_i(k)| \quad (7)$$

式中, Δ_{ik} 为绝对差; Δ_{\min} 为两级最小差; Δ_{\max} 为两级最大差; $\rho \in (0, 1)$ 为分辨系数,它直接影响分析分辨率的因子,其取值直接决定灰色关联系数的分布情况。可以看出,关联系数 $\gamma(x_0(k), x_i(k))$ 是分辨系数 ρ 的单调增函数, ρ 越大,表明对 Δ_{\max} 越重视,各因子对关联度的影响越大; ρ 越小,表明对 Δ_{\max} 越不重视,各因子对关联度的影响越小。在实际使用时,应根据序列间的关联程度选择分辨系数,一般取 $\rho \leq 0.5$ 。

步骤 3 计算灰色加权关联度。综合各点的关联度系数,得到比较序列 x_i 的曲线与参考序列 x_0 的曲线之间的关联度 $\gamma(x_0, x_i)$ 。

$$\gamma(x_0, x_i) = \frac{1}{n} \sum_{k=1}^n \gamma(x_0(k), x_i(k)) \quad (8)$$

步骤 4 确定权重集。上述计算没有考虑工作量驱动因子的权重因素,但在实际软件系统的设计和选择中,必须体现软件工作量因子的权重差异。软件工作量的权重值在理论上可用区间灰数 $\otimes \beta_k$ 表示, β_k 为区间灰数 $\otimes \beta_k$ 的白化值。在工程实际中为便于计算,权重值可直接赋予其白化值 $\beta = (\beta_1, \beta_2, \dots, \beta_n)$, 且有

$$\gamma(x_0, x_i) = \sum_{k=1}^n \beta_k \cdot \gamma(x_0(k), x_i(k)),$$

$$\sum_{k=1}^n \beta_k = 1, \forall \beta_k \in [0, 1] \quad (9)$$

步骤 5 分析计算结果。根据灰色加权关联度的大

3 实验分析

为了验证本文方法的有效性, 分别对软件估算领域 4 个典型数据集 Albrecht/Desharnais/Kermere/Mermaid^[12] 进行研究, 以此分析估算方法的性能, 这些数据集的项目特征描述参见表 1。

表 1 实验数据集描述

数据集	项目数	项目特征
Albrecht	24	8 (input, output, query, file, function point, source loc)
Desharnais	77	8 (TeamExp, ManagerExp, YearEnd, Length, Language, Transactions, Entities, PointsAdjust, Envergure, PointsNonAdjust)
Kermere	15	5 (Software, Hardware, Months, KSLOC, SLOC/MM)
Mermaid	26	4 (Adjusted FP, Raw FP, Total Duration, Project Type)

3.1 评价标准

评价标准 1 MMRE

$$MMRE = \frac{\sum_{i=1}^m MRE(i)}{m} = \frac{\sum_{i=1}^m \left| \frac{Estimated_i - Actual_i}{Actual_i} \right|}{m} \quad (19)$$

式中, m 是待估算项目数; MRE 为相对误差量; $Actual_i$ 表示项目 i 的真实工作量; $Estimated_i$ 表示项目 i 的估算工作量。

评价标准 2 PRED(ℓ)

$$PRED(\ell) = \frac{\lambda}{n} \times 100 \quad (20)$$

式中, λ 是估计误差 $MRE \leq \eta\%$ 的项目数; n 是所有估算项目数, 通常取 $\eta=25$ 。

3.2 实验结果分析

下面以收集到的软件项目数据库为例, 利用提出的 GRA 方法估算软件项目工作量。在数据分析过程中, 采用 Jackknife 方法^[13] 进行验证, 每次选取一个项目作为测试集, 其余项目作为训练集。为了充分反映 GRA 方法的估算性能, 分别将其与基于回归分析、BP 神经网络、类比估算等方法进行比较。

(1) 回归分析方法

这里分别建立各数据库的多元线性回归模型, 其中模型参数利用最大似然估计法^[14] 进行估计, 具体如下:

Albrecht 数据集的多元线性回归模型为

$$Effort = -0.226\,1 \times Input - 0.131\,2 \times Output + 0.290\,3 \times Query - 0.680\,0 \times File - 65.885\,6 \times FP + 0.090\,8 \times SLOC + 50.500\,4 \quad (21)$$

Desharnais 数据集的多元线性回归模型为

$$Effort = -433.25 \times TeamExp + 408.805\,7 \times ManagerExp + 201.270\,1 \times Length + 4.236\,1 \times Transactions + 7.905\,6 \times Entities + 4.459\,4 \times PointsAdjust + 92.138\,9 \times Envergure - 1\,777.457\,9 \times Langage - 278.078\,6 \quad (22)$$

Kermere 数据集的多元线性回归模型为

$$Effort = -0.890\,7 \times Months + 1.582\,5 \times KSLOC - 0.190\,1 \times SLOC/MM + 148.341\,3 \quad (23)$$

Mermaid 数据集的多元线性回归模型为

$$Effort = 4.365\,6 \times Adjusted\,FP + 1.988\,9 \times Raw\,FP + 858.266\,2 \times Total\,Duration - 3.615\,1 \quad (24)$$

(2) BP 神经网络方法

BP 网络的构建、训练和测试利用 Matlab 实现。网络输入为各数据集的项目特征, 网络输出为软件工作量, 网络隐层节点数设置为 30, 网络各层间的传递函数分别设置为 purelin/ purelin/ logsig, 并利用 traingda 函数训练网络。BP 网络训练精度设置为 10^{-4} , 最大迭代次数为 5 000 次, 训练步长为 0.000 5, 学习率为 0.25。

(3) 类比估算方法

类比估算采用多项目综合的策略^[15] 进行, 设类比项目数 $K=3$, 则估算的软件工作量为

$$\hat{E}_p = \sum_{k=1}^K \frac{sim(p, p_k)}{\sum_{k=1}^K sim(p, p_k)} E_{p_k} \quad (25)$$

式中, p 是新项目; p_k 是与新项目相似的第 k 个历史项目; E_{p_k} 是历史项目 k 的工作量; $sim(p, p_k)$ 是新项目 p 与历史项目 p_k 之间的相似度。实验选取 Euclidean 函数来度量不同软件项目的相似度:

$$sim(p, p_k) = 1 / \left[\sqrt{\sum_{i=1}^n \omega_i (p(i) - p_k(i))^2} + \delta \right] \quad (26)$$

式中, $p(i)$, $p_k(i)$ 分别表示第 i 个特征值; ω_i 表示特征 i 的权重; δ 是一个常数, 取 $\delta=0.001$ 。

(4) GRA 方法

根据 2.3 节的步骤估算软件工作量, 设置分辨系数 $\rho=0.5$, 在 GRA 过程中不筛选特征, 并认为各特征具有相同权重, 设置估算工作量的项目数 $\ell=3$ 。

(5) 特征选取+GRA 方法

该方法的参数设置同上, 具体过程是: 首先计算各特征与软件工作量的关联程度, 并从中选取关联度最大的特征; 然后采用回归技术进行特征增加或删除, 选取出最优特征集 *Optimal_Feature_Set*; 最后在最优特征集的基础上, 利用 2.3 节的步骤估算软件项目工作量。各数据集的最优特征选取结果见表 2 (表中“斜体下划线”为灰色关联度最大的特征)。

表 2 各数据集的最优特征选取结果

数据集	最优特征集(<i>Optimal_Feature_Set</i>)
Albrecht	{ <i>Output</i> , Query, ADJ, Function Point}
Desharnais	{YearEnd, <i>PointsAdjust</i> , Langage}
Kermere	{ <i>KSLOC</i> , SLOC/MM}
Mermaid	{ <i>Adjusted FP</i> , Total Duration}

利用上述 5 种方法对 4 个软件数据集进行估算, 工作量估算结果如图 1、图 3、图 5 和图 7 所示; 这些估算方法的平均误差率如图 2、图 4、图 6 和图 8 所示; 不同估算方法的性能对照参见表 3。

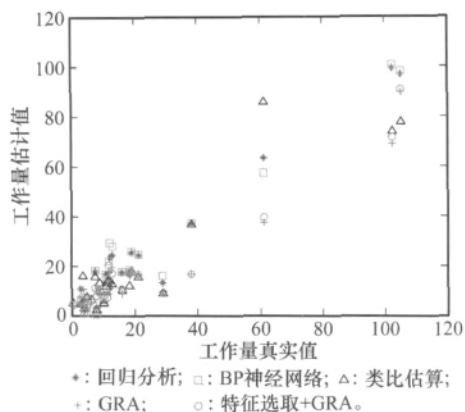


图1 Albrecht数据集的工作量估算结果

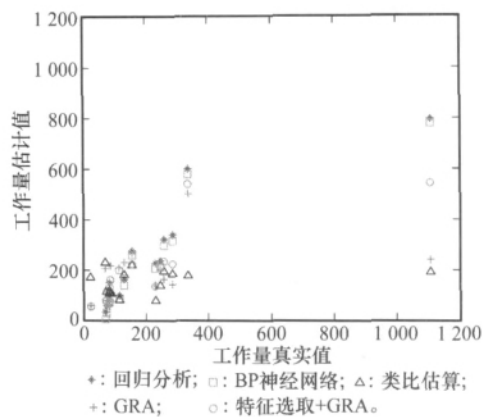


图5 Kermere数据集的工作量估算结果

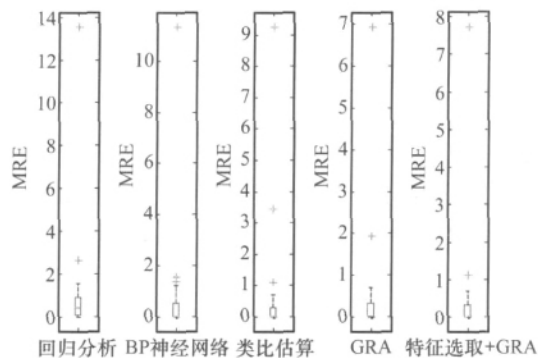


图2 不同方法估算 Albrecht 数据集的误差

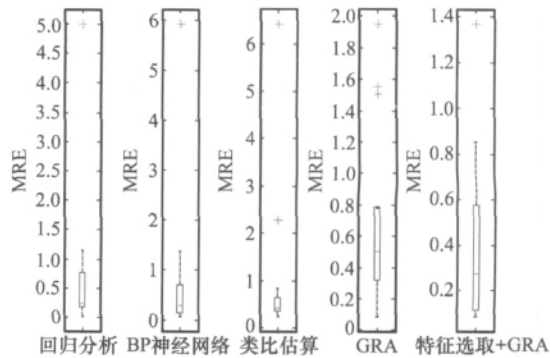


图6 不同方法估算 Kermere 数据集的误差

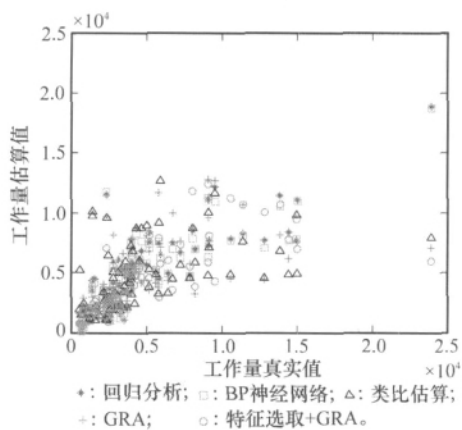


图3 Desharnais数据集的工作量估算结果

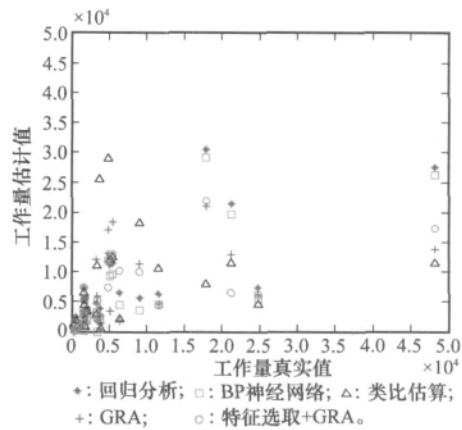


图7 Mermaid数据集的工作量估算结果

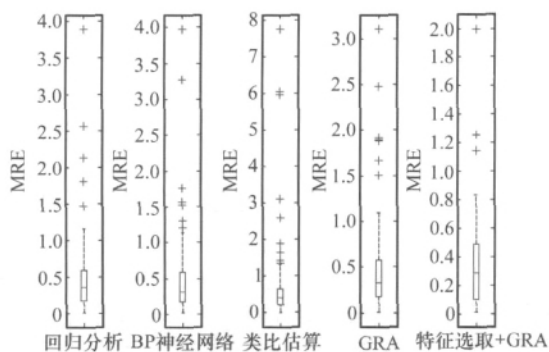


图4 不同方法估算 Desharnais 数据集的误差

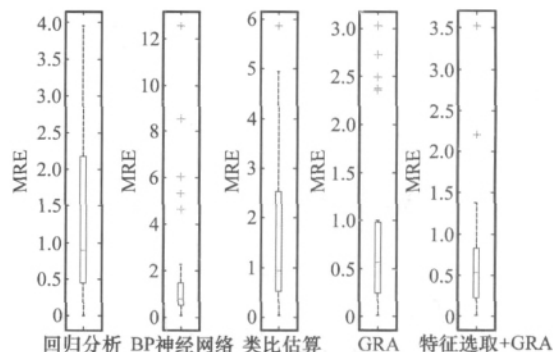


图8 不同方法估算 Mermaid 数据集的误差

表 3 几种估算方法的性能对比

工作量估算方法	Albrecht Database		Desharnais Database		Kermere Database		Mermaid Database	
	MMRE	PRED	MMRE	PRED	MMRE	PRED	MMRE	PRED
基于回归分析	1.097	41.67	0.522 3	37.66	0.746 0	33.33	1.276 4	11.54
基于 BP 神经网络	1.085 8	37.50	0.517 4	37.66	0.777 2	40.00	1.444 2	11.54
基于类比估算	0.848 2	33.33	0.776 0	29.87	0.957 8	13.33	1.598 2	19.23
基本 GRA	0.659 6	41.67	0.523 7	37.66	0.689 5	20.00	0.868 6	26.92
特征选取+GRA	0.622 1	54.17	0.342 6	50.65	0.401 4	46.67	0.696 4	34.62

从实验结果可以看出,对于历史数据少的软件估算问题,基于 GRA 的方法具有较高的估算精度,其性能明显优于基于回归分析、BP 神经网络和类比估算等方法;而经过最优特征选取后的 GRA 方法,由于剔除了与工作量相关程度低的特征因子,使得该方法获得了更小的 MMRE 和更大的 PRED。

4 结束语

软件工作量估算是软件工程领域重要的研究方向,是可行性研究阶段的重要任务,也是制定计划的依据。为了有效解决历史数据少、信息贫乏的软件工作量估算问题,本文提出了一种基于 GRA 的估算方法。首先结合灰色关联理论分析了项目特征与工作量之间的关联程度;然后利用回归分析技术选取最优特征集,并对不同类别的特征予以处理;最后在选取最优特征集的基础上,利用 GRA 方法计算新项目 and 历史项目之间的关联度,并利用关联程度大的项目估算软件工作量。实验结果表明,所提出的方法能够在历史信息缺乏的情况下,提高软件工作量的估算精度,其性能明显优于基于回归分析、BP 神经网络和类比估算等方法;而经过最优特征选取后的 GRA 方法,由于剔除了与工作量相关程度低的特征因子,使得该方法获得了更小的 MMRE 和更大的 PRED。

将灰色关联分析用于软件工作量估算,不仅是工作量估算方法由定性向定量发展的有益尝试,而且也可作为一种变通的方法用于软件发展生命周期中,具有较强的适应性和推广价值。

参考文献:

- [1] Huang S J, Chiu N H, Chen L W. Integration of the grey relational analysis with genetic algorithm for software effort estimation[J]. *European Journal of Operational Research*, 2008, 188(3): 898-909.
- [2] Mahmoud O E. Improved estimation of software project effort using multiple additive regression trees[J]. *Expert Systems with Application*, 2009, 36(7): 10774-10778.
- [3] Nikolaos M, Lefteris A. LSEbA: least squares regression and estimation by analogy in a semi-parametric model for software cost estimation[J]. *Empirical Software Engineering*, 2010, 15(5): 523-555.
- [4] Benediktsson O, Dalcher D, Reed K, et al. COCOMO-based effort estimation for iterative and incremental software development[J]. *Software Quality Journal*, 2003, 11(4): 265-281.
- [5] Mohammad A, Daniel N, Peter I C. Analogy-based software effort estimation using fuzzy numbers[J]. *Journal of Systems and Software*, 2011, 84(2): 270-284.
- [6] Kotten C V, Gray A R. Bayesian statistical effort prediction models for data-centered 4GL software development[J]. *Information and Software Technology*, 2006, 48(11): 1056-1067.
- [7] Farhad S G. Neural network application in software cost estimation: a case study[C]// *International Symposium on Innovations in Intelligent Systems and Applications*, 2011: 69-73.
- [8] Mohammad A, Daniel N, Peter I C. Fuzzy grey relational analysis for software effort estimation[J]. *Empirical Software Engineering*, 2010, 15(1): 60-90.
- [9] Song Q B, Shepperd M. Predicting software project effort: a grey relational analysis based method[J]. *Expert Systems with Applications*, 2011, 38(6): 7302-7316.
- [10] Wang Y, Song Q B, Stephen M, et al. Integrate the GM(1,1) and verhulst models to predict software stage-effort[J]. *IEEE Trans. on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 2009, 39(6): 647-658.
- [11] 卢纹岱. SPSS for Windows 统计分析[M]. 北京:电子工业出版社,2003. (Lu W D. *SPSS for Windows statistic analysis* [M]. Beijing: Electronic Industry Press, 2003.)
- [12] Michael A K. *A methodology for software cost estimation using machine learning techniques* [M]. Monterey, CA: Naval Postgraduate School, 1993.
- [13] Han J W, Micheline K. *Data mining concepts and techniques* [M]. San Francisco, CA: Mogen Kaufmann Publishers, 2011.
- [14] 盛骤, 谢式千, 潘承毅. 概率论与数理统计[M]. 北京: 高等教育出版社, 2009. (Sheng Z, Xie S Q, Pan C Y. *Probability and mathematical statistics* [M]. Beijing: Higher Education Press, 2009.)
- [15] Li Y F, Xie M, Goh T N. A study of non-linear adjustment for analogy based software cost estimation[J]. *Empirical Software Engineering*, 2009, 14(6): 603-643.