

相関を考慮したRBFネットワークを用いたソフトウェアメトリクスデータの分析

後藤, 一成 / GOTOH, Kazunari

(出版者 / Publisher)

法政大学大学院理工学・工学研究科

(雑誌名 / Journal or Publication Title)

法政大学大学院紀要. 理工学・工学研究科編

(巻 / Volume)

56

(開始ページ / Start Page)

1

(終了ページ / End Page)

6

(発行年 / Year)

2015-03-24

(URL)

<https://doi.org/10.15002/00011348>

相関を考慮したRBFネットワークを用いた ソフトウェアメトリクスデータの分析

ANALYSIS OF SOFTWARE METRICS DATA BASED ON RBF NETWORKS CONSIDERING
CORRELATION

後藤 一成

Kazunari GOTOH

指導教員 木村 光宏

法政大学大学院理工学研究科システム工学専攻修士課程

It is known that an RBF (Radial Basis Function) network model can be used to predict the number of inherent software faults from the data sets concerning software development data in the literature. To grasp the information on the number of software faults before beginning the testing process, can be the great help for the software development managers. The aim of this study is to provide the improved RBF network model in terms of the prediction accuracy. Therefore we propose a new RBF network model considering the correlation among the input data. In particular, we investigate the usefulness of introducing the correlation factors into the RBF networks and compare the performances between the proposed model and the previous one by using the actual data sets.

Key Words : software reliability, multivariate analysis, RBF network, neural network

1. はじめに

(1) 研究背景

現代の社会はコンピュータの品質の上に成り立っており、特にソフトウェアの品質の良し悪しが社会を支えるシステムの技術的側面の質を左右する。現在、自動車や家電等の製品はリコール等の制度が整っており、故障についてあまり考えることなく快適性や利便性を享受できるようになった。しかし、ソフトウェアに関して言えば、製品販売後の重大なフォールト検出等の事例は多くの利用者に影響を及ぼし、重大な社会問題に発展する恐れもある。このような障害を未然に防ぐことがソフトウェアを開発する企業に求められている時代であり、ソフトウェアの信頼性について考え、対応することが必要である。

(2) 研究目的

稼働しているソフトウェアのプログラム内にフォールトが存在しているとき、そのフォールトに気が付くのはソフトウェアが故障した時であり、様々なトラブルを引き起こす可能性がある。そのため、ソフトウェアの開発段階で潜在しているフォールトをテスト工程で全て取り除くことができれば、品質の高いソフトウェアを作ることができる。しかし、プログラム内にフォールトが存在しないことを証明するのは困難であるため、ソフトウェア開発のテスト工程開始時に、プログラム内にどれだけフォールトが存在しているのかを予測することが重要である。そうすることで、テスト工程終了時に発見したフォールト数が予測値に達しなかった場合、テストの手法を変えることで未発見のフォールトを取り除くことが可能になる。

本研究では、文献 [3] のモデルを基に、RBF ネットワークで入力データの相関を考慮したモデルを提案する。いかなるデータ間においても、相関係数の大小に関わらず必ず相関が

ある。それを考慮するモデルを構築することで、先行研究 [3] で提案されたモデルの予測精度を上回るのではないかと考える。これは先行研究にないモデルであり、開発途上にあるソフトウェアに含まれる潜在フォールト数の予測精度が過去の研究に対してどの程度向上するのか、RBF ネットワークで相関を考慮することの有用性を検討する。

2. RBF ネットワーク

(1) 先行研究のモデル

以下に RBF ネットワークの概要と未知パラメータの推定方法を示す。これらは先行研究である文献 [3] の記述と概ね同様であることを付記しておく。

(a) RBF ネットワークの概要 放射基底関数 (Radial Basis Function, 以下 RBF) ネットワークは、放射基底関数と呼ばれる局在化した基底関数の重ね合わせにより、任意関数の補間を行う手法である [4]。RBF ネットワークは、中間層に放射基底関数を用いた 3 層ニューラルネットワークの一種であり、各ニューロンからの出力の重み付き線形和が出力となる。シグモイド関数を用いたニューラルネットワークに比べ中間層の数が少なく構造が簡単であり、関数近似能力や学習効率に優れていると報告されている [5]。図 1 に RBF ネットワークの構成図を示す。

入力を $x \in R^m$ 、出力を $y \in R$ とすると、RBF ネットワークは、 N 個の入出力データの組、

$$(x^1, y^1), \dots, (x^N, y^N) \quad (1)$$

が与えられたとき、これらの条件を満たす関数 $y = f(x)$ を同定することが目的である。また本研究では、中間層の放射基底関数にガウシアン関数を採用した。

入力 x に対する j 番目の RBF 素子の出力 G_j は、

$$G_j = \exp\left(-\sum_i \frac{(x_i - \mu_{ij})^2}{2\sigma_j^2}\right), \quad (2)$$

である。ここで、 μ_{ij} は第 i 次元入力の j 番目の RBF 素子の平均値に相当する量、 σ_j^2 は j 番目の RBF 素子の標準偏差に相当する量である。さらにネットワークの出力 y は、

$$y = \sum_j w_j G_j, \quad (3)$$

で与えられる。ここで、 w_j は j 番目の RBF 素子の重みであるシナプス結合係数、 m は中間層の素子数、 n は入力数である。以降、RBF パラメータの集合を、簡単のため (m, μ, σ^2, w) と表す。

RBF ネットワークは、与えられた入出力関係を満たす関数を近似することが目的であるため、評価関数として RBF ネットワークの出力と教師信号との 2 乗誤差が多く用いられる。 x^p を入力したときの RBF ネットワークの出力を y とすると、

$$E(m, \mu, \sigma^2, w) = \frac{1}{2} \sum (y - y^p)^2, \quad (4)$$

により評価関数 E を定義する。ここで、 y^p は教師信号で所与の値である。このとき、解くべき関数同定問題は、必要な同定の精度 ϵ を用いて $E < \epsilon$ を満たすパラメータの組 (m, μ, σ^2, w) を見つける問題となる [6]。

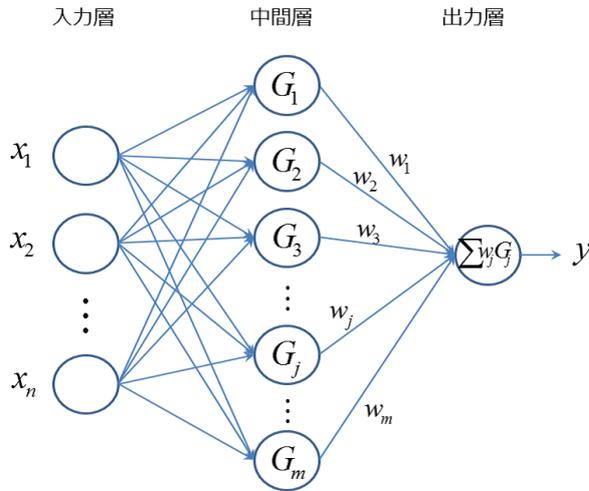


図 1 RBF ネットワークの構成図。

(b) 未知パラメータの推定 RBF ネットワークのアルゴリズムを以下に示す。

1. ネットワークの初期化乱数を用いてシナプス結合係数の初期値を小さな値に設定し、ガウシアン関数のパラメータである μ と σ^2 の初期値を与える。
2. 入力ベクトルの入力まず、入力層に入力ベクトルが入力される。入力ベクトルは中間層に伝播する。各ニューロンは中間層のガウシアン関数によって出力値が決定される。
3. 中間層から出力層への出力中間層の素子の出力と各シナプスの結合係数との積和を求め、出力層へ出力する。

4. 出力層への教師信号の入力出力層へは、入力ベクトルに対応した教師ベクトルが与えられる。
5. 誤差逆伝播によるシナプス結合係数とガウシアン関数のパラメータの学習シナプスの結合係数とガウシアン関数のパラメータである μ と σ^2 を更新する。
6. 2 へ戻る 2 から 4 の動作を繰り返して、シナプス結合係数とガウシアン関数のパラメータである μ と σ^2 を学習していく。2 から 4 の動作を任意の回数行って終了する。

RBF ネットワークの未知パラメータを最急降下法により求める。最急降下法による学習則は、

$$w_j^{new} = w_j^{old} - \eta \frac{\partial E}{\partial w_j}, \quad (5)$$

$$\mu_{ij}^{new} = \mu_{ij}^{old} - \eta \frac{\partial E}{\partial \mu_{ij}}, \quad (6)$$

$$\sigma_j^{2\ new} = \sigma_j^{2\ old} - \eta \frac{\partial E}{\partial \sigma_j^2}, \quad (7)$$

である。ただし、 η は正の学習係数である。上記の学習を繰り返し行い、パラメータの推定を行っていく。

(2) 本研究のモデル

本研究は、RBF ネットワークで入力データの相関を考慮したモデルを提案する。先行研究では、入力 x に対する j 番目の RBF 素子の出力 G_j を

$$G_j = \exp\left(-\sum_i \frac{(x_i - \mu_{ij})^2}{2\sigma_j^2}\right), \quad (8)$$

としている。本研究では、入力データの相関を考慮するため、

$$G_j = \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu}_j)\right), \quad (9)$$

とする。行列 $\boldsymbol{\Sigma}^{-1}$ は分散・共分散行列の逆行列、 $\boldsymbol{\mu}_j$ は μ_{ij} のベクトル表示である [7]。これを用いて、RBF ネットワークの出力 y を求める。

RBF ネットワークの中間層のパラメータとシナプス結合係数の更新は、先行研究と同様に最急降下法により求める。

3. ソフトウェアプロジェクトデータ

本研究の目的の一つは、ウォータフォールモデルにおけるソフトウェア開発の最終段階であるテスト工程に入るまでに得られた多変量データ（ソフトウェアプロジェクトデータと称する）を用いて、RBF ネットワークによる予測モデルを上述のように構築することで、プログラム内に潜在するソフトウェアフォールト数をできるだけ早期に、かつ精度よく予測したいというものである。以下に示すのは、文献 [1] に基づいた、あるソフトウェア開発プロジェクトにおいて観測された 19 個のプロジェクトデータの意味付け、および変数の選択方法である。

(1) 変数の意味付け

本研究では、既存の論文 [1] で使われたソフトウェアプロジェクトのデータを用いて、構築したモデルの予測精度を評価した。各データの意味付けも合わせて以下に列挙する。このデータは文献 [8] により解析されたものでもあり、以下の変数リストはそこでの記述と同様であることを付記しておく。

- x_1 プロジェクト初期のリスク率 (0~100)
- x_2 開発の大きさに対する契約内容の問題数 (数/KLOC)
- x_3 開発の大きさに対する日数 (日/KLOC)
- x_4 開発の大きさに対する計画の問題の数 (数/KLOC)
- x_5 開発の大きさに対する日数 (日/KLOC)
- x_6 設計品質評価の層別因子 (1=評価あり, 0=評価なし)
- Y_q 開発中及びテスト中に発見されたフォールト数
- Y_c 費用超過率
- Y_d 納期延滞率

- Y_c は x_1 と x_6 に強い相関を示す.
- Y_d は x_2 と x_5 に強い相関を示す.
- Y_d は x_1 と x_4 に強い相関を示す.
- x_2 は x_3 と x_4 に強い相関を示す.
- x_2 と x_4 は x_5 に強い相関を示す.

以上の相関関係および、全てのパターンの重回帰分析による適合度を考慮して説明変数の選択を行った。その結果として各目的変数に対して選ばれた説明変数を以下に示す。

Y_q : x_1, x_2, x_6

Y_c : x_1, x_2, x_4, x_6

Y_d : x_1, x_2

表 1 に文献 [1] に掲載されたデータセットを示す¹。

表 1 実測データ [1] (No. はプロジェクト番号)。

No.	x_1	x_2	x_3	x_4	x_5	x_6	Y_q	Y_c	Y_d
1	19	0.323	8.065	0.473	10.430	0	1	1.02	1.04
2	13	0.690	4.483	0.259	2.414	0	0	1.02	1.04
3	11	0.127	4.322	0.059	1.186	0	2	0.95	1.00
4	24	0.150	3.083	0.293	11.880	0	5	1.00	1.00
5	17	0.131	1.343	0.082	4.552	0	0	1.00	0.94
6	29	0.638	2.553	0.979	40.000	0	1	1.03	1.08
7	35	0.241	5.060	0.494	19.398	0	4	1.08	1.04
8	25	0.076	0.138	0.069	0.519	0	0	0.89	1.00
9	28	0.057	0.573	0.046	0.573	0	3	1.08	1.05
10	38	0.323	2.177	0.258	3.226	0	5	1.10	1.05
11	42	0.329	2.336	0.292	3.723	0	6	1.14	1.05
12	30	0.256	1.923	0.321	3.205	0	3	1.08	1.05
13	28	0.149	1.940	0.224	2.985	0	2	1.00	1.00
14	35	1.250	7.500	1.563	62.500	0	0	1.10	1.22
15	23	0.331	7.868	0.184	7.647	1	0	0.97	1.06
16	29	0.345	10.345	0.379	14.483	1	0	0.89	1.00
17	25	0.047	2.617	0.005	0.094	1	0	0.99	1.00
18	18	0.000	0.000	1.167	10.000	1	0	0.78	0.99
19	30	0.059	0.118	0.188	0.941	1	0	0.92	1.00

(2) 説明変数の選択

表 1 のデータを相関分析することによって得られた各変数間の相関関係を表 2 に示す。

表 2 相関行列。

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	1					
x_2	0.226	1				
x_3	-0.029	0.517	1			
x_4	0.198	0.659	0.224	1		
x_5	0.291	0.815	0.409	0.861	1	
x_6	-0.094	-0.276	0.140	-0.002	-0.151	1
Y_q	0.530	-0.140	-0.185	-0.180	-0.156	-0.503
Y_c	0.552	0.434	0.092	-0.014	0.228	-0.635
Y_d	0.449	0.855	0.380	0.656	0.777	-0.241

表 2 より、相関関係について以下のことが考えられる。

- Y_q は x_1 と x_6 に強い相関を示す。

¹ x_2 から x_5 については小数点以下第 4 位を四捨五入して掲載した。

このデータを解析した文献 [1] では、各 QCD に対する予測を重回帰モデルによる手法で行っている。本研究では目的変数 Y_q において、上記の重回帰モデル、文献 [2] によって提案されたニューラルネットワークモデル、文献 [3] によって提案された RBF ネットワークモデルと本研究の RBF ネットワークモデルを比較するため、先行研究と同様 3 つの変数、 x_1, x_2 , および x_6 を選び、RBF ネットワークの入力としてモデルの構築を行った。

4. 適用例

RBF ネットワークを用いる際、いくつかパラメータの値を設定する必要がある。その値は対象とする問題によって異なるが、本研究の実験結果を再現できるよう、以下に各パラメータの値を示す。

[各層の素子数]

本研究では先行研究との予測精度を比較するため、先行研究と同様の変数を選択している。従って、入力層の素子数を 3 個、出力層の素子数を 1 個としている。中間層の素子数については、その決定的方法は知られていないため、試行錯誤的に決めるのが一般的である。中間層の素子数の決定法については、必要な素子数が入力および教師データの情報量に依存しており、素子数を必要以上多くしてもネットワークの機能は大きく変化しないことが判明している。

また、中間層の素子数が必要最小限の数であれば学習は容易に収束するが、それよりも 1 個でも少ないと収束しなくなる。そこで原始的ではあるが、中間層の素子数の初期値を入力層の素子数の 2 倍程度として、そこから学習が不可能になるまで減少させていく。この方法が確実に早いことが経験的に知られており、本研究もそれに習った。

文献 [3] では、中間層の素子数を 3 個、5 個、7 個とした 3 つの予測モデルを構築して推定を行っている。本研究の目的の一つは、先行研究の予測モデルと本研究の予測モデルを比較することである。そこで文献 [3] と同様に、本研究も中間層の素子数を 3 個、5 個、7 個とした 3 つの予測モデルに関して考察した。

[ガウシアン関数のパラメータ]

本研究の RBF ネットワークの中間層に用いるガウシアン関数は式 (9) とする。入力層の素子数が 3 個であるため、入力 x_2 と x_3 の標本相関係数を ρ_1 、入力 x_1 と x_3 の標本相関係数を ρ_2 、入力 x_1 と x_2 のそれを ρ_3 とし、それぞれ $\rho_1 = -0.276$, $\rho_2 = -0.094$, $\rho_3 = 0.226$ であった。また、 μ_{ij} と σ_j^2 は未

知パラメータであり、これらの値を決める際的设计理論は確立されているわけではない。そこで、先行研究ではこれらの未知パラメータの初期値を実験的な経験により決めている。本研究もそれに倣い、未知パラメータの初期値を実験的な経験により決めた。本研究のRBFネットワークモデルにおけるガウシアン関数のパラメータ μ_{ij} と σ_j^2 の初期値を以下に示す。

- 中間層の素子数を3個とした場合の μ_{ij} と σ_j^2 の初期値を以下に示す。

表3 μ_{ij} の初期値.

入力層 (i)	中間層 (j)		
	1	2	3
1	26.1	25	20
2	2.1	0.5	0.5
3	1.9	1	1.2

表4 σ_j^2 の初期値.

中間層 (j)	—
1	21.16
2	9.61
3	5.29

- 中間層の素子数を5個とした場合の μ_{ij} と σ_j^2 の初期値を以下に示す。

表5 μ_{ij} の初期値.

入力層 (i)	中間層 (j)				
	1	2	3	4	5
1	26.1	25	20	30	20.3
2	2.1	0.5	0.5	0.2	1
3	1.9	1	1.2	0.1	1

表6 σ_j^2 の初期値.

中間層 (j)	—
1	21.16
2	9.61
3	5.29
4	5.29
5	4.84

- 中間層の素子数を7個とした場合の μ_{ij} と σ_j^2 の初期値を以下に示す。

表7 μ_{ij} の初期値.

入力層 (i)	中間層 (j)						
	1	2	3	4	5	6	7
1	26.1	25	20	30	20.3	20	20
2	2.1	0.5	0.5	0.2	1	0.5	0.8
3	1.9	1	1.2	0.1	1	0.8	1.2

表8 σ_j^2 の初期値.

中間層 (j)	—
1	21.16
2	9.61
3	5.29
4	5.29
5	4.84
6	6.25
7	4.84

[シナプス結合係数の初期値と更新回数]

シナプス結合係数の初期値は通常0~1または-1~+1の範囲で設定するケースが多く、小さな乱数を初期値として与えることが知られている。本研究でも一般的に行われているように、小さな乱数をシナプス結合係数の初期値として与えた。図2は学習回数による誤差二乗和の振舞いを示しており、これらの結果を踏まえて本研究ではシナプス結合係数の更新回数を2500回とした。

- 中間層の素子数を3個とした場合の w_j の初期値を以下に示す。

表9 w_j の初期値.

中間層 (j)	—
1	0.0013
2	0.5636
3	0.1933

- 中間層の素子数を5個とした場合の w_j の初期値を以下に示す。

表10 w_j の初期値.

中間層 (j)	—
1	0.0013
2	0.5636
3	0.1933
4	0.8087
5	0.5850

- 中間層の素子数を7個とした場合の w_j の初期値を以下に示す。

表11 w_j の初期値.

中間層 (j)	—
1	0.0013
2	0.5636
3	0.1933
4	0.8087
5	0.5850
6	0.4799
7	0.3503

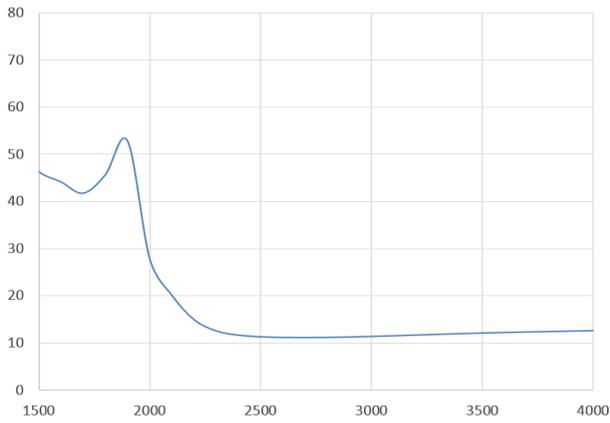


図 2 学習回数による誤差二乗和の振舞い (横軸：学習回数, 縦軸：誤差二乗和)。

[学習係数]

学習係数 η は 1 未満の小さな値とし, 半ば試行錯誤的に決めるのが一般的である。学習係数が大きければ大きいほど学習は早いが, 最急降下法を用いる場合, 学習係数が大きいと大域的な最小値あるいは局所的な最小値を通り過ぎてしまうことがあり, 振動が大きくなり, うまく収束しないことが知られている。図 3 は学習係数の変化に伴う誤差二乗和を示しており, これらの結果を踏まえて本研究では学習係数 η を $\eta = 0.0001$ とした。

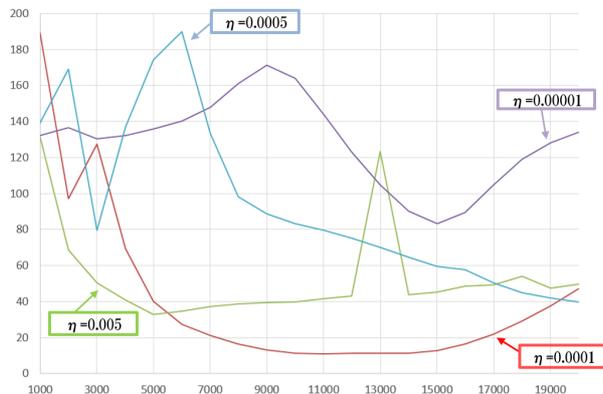


図 3 学習係数の変化による誤差二乗和の変化 (横軸：学習回数, 縦軸：誤差二乗和)。

[中間層のパラメータの学習回数]

中間層のパラメータの学習回数は解くべき問題や学習係数等の他のパラメータとの関係で適正な値が決まるので理論値はないが, 1000~10 万回が一般的である。学習回数の設定値を大きくしすぎると, 過学習により, 学習の進行に伴って学習データの持っている固有の微細構造まで学習してしまう。それにより, 学習データについての近似精度は向上するが, 未学習データに対する精度は悪くなってしまふ。そこで, 本研究では適切な学習回数を模索するためにいくつかの設定値を試した。図 4 は中間層の素子数が 5 個のモデルについての学習回数が 1000~10000 回までの誤差二乗和の変化を示しており, 学習回数が増加するにつれて低減傾向は徐々に弱まるが, 概ね低下することが確認できた。これらの結果を踏まえて本研究では学習回数を 5000~20000 回とした。

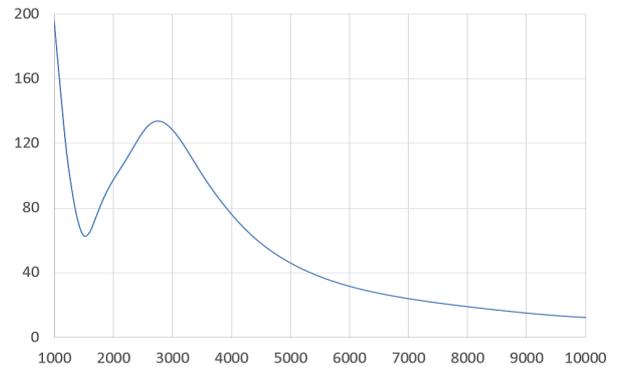


図 4 学習回数による誤差二乗和の振舞い (横軸：学習回数, 縦軸：誤差二乗和)。

(1) 推定結果

パラメータの学習回数を 100 回刻みで 5000 回から 20000 回まで増加させ, 最も残差平方和の値が小さくなった時点の推定結果を表 12 に示す。中間層の素子数が 3 個の予測モデルの学習回数は 20000 回, 5 個の予測モデルの学習回数は 11200 回, 7 個のそれは 11500 回であった。

表 12 推定結果。

No.	Y_q	RBF($m=3$)		RBF($m=5$)		RBF($m=7$)	
		予測値	残差 ²	予測値	残差 ²	予測値	残差 ²
1	1	0.121	0.773	0.762	0.057	1.717	0.513
2	0	0.317	0.101	1.192	1.421	0.769	0.591
3	2	0.093	3.638	0.364	2.676	0.263	3.018
4	5	1.471	12.454	2.864	4.562	2.789	4.887
5	0	0.299	0.090	0.574	0.329	-0.919	0.844
6	1	0.981	0.000	0.558	0.195	0.261	0.547
7	4	4.396	0.157	3.038	0.926	3.090	0.829
8	0	2.915	8.495	0.253	0.064	0.065	0.004
9	3	2.519	0.231	2.420	0.336	2.372	0.394
10	5	5.476	0.227	5.220	0.048	4.909	0.008
11	6	4.670	1.770	5.676	0.105	6.021	0.000
12	3	1.737	1.596	3.292	0.086	3.151	0.023
13	2	2.725	0.525	2.472	0.223	2.555	0.308
14	0	0.587	0.345	0.216	0.047	0.316	0.100
15	0	-0.059	0.004	0.052	0.003	0.132	0.017
16	0	-1.048	1.097	0.168	0.028	0.057	0.003
17	0	0.658	0.433	-0.235	0.055	-0.103	0.011
18	0	0.519	0.269	0.233	0.054	0.122	0.015
19	0	0.406	0.165	-0.151	0.023	-0.034	0.001
合計	—	—	32.369	—	11.238	—	12.114

(2) 推定結果の比較

各研究の予測モデルのうち, 研究内で最も予測精度の高かったモデルの推定結果を表 13 に示す。重回帰分析の推定結果はチューニングによるもの, ニューラルネットワークと RBF ネットワークの推定結果は予測によるものである。推定結果の表中の * は, その行において $(Y_q - \hat{Y}_q)^2$ の値が最も小さく, 最良のモデルであることを示す。ただし, 予測によるニューラルネットワークと RBF ネットワークのモデルの結果を比較しており, チューニングによる重回帰分析の結果は除いて

ある。各研究の予測モデルにおける * の数は、ニューラルネットワークが 4 個、先行研究の RBF ネットワークが 6 個、本研究の RBF ネットワークが 9 個となり、RBF ネットワークのモデルの方がニューラルネットワークのモデルよりも予測精度が高く、最良のモデルの数は本研究のモデルが最も多い結果となった。また、残差平方和という評価基準においては、先行研究で最も予測精度の高い文献 [3] の RBF ネットワークモデルと比べ、本研究のモデルの改善率は 44% 向上した。

表 13 各研究の推定結果。

No.	Y_q	NN($h=2$)		RBF 先 ($m=5$)		RBF 本 ($m=5$)	
		予測値	残差 ²	予測値	残差 ²	予測値	残差 ²
1	1	1.770	0.593	0.917	0.007*	0.762	0.057
2	0	1.072	1.149	0.540	0.292*	1.192	1.421
3	2	1.121	0.772*	0.475	2.325	0.363	2.676
4	5	2.220	7.729	1.727	10.710	2.864	4.562*
5	0	1.434	2.056	0.834	0.696	0.574	0.329*
6	1	1.778	0.605	1.502	0.252	0.558	0.195*
7	4	4.317	0.100	4.030	0.001*	3.038	0.926
8	0	2.763	7.634	2.063	4.255	0.253	0.064*
9	3	3.309	0.096	2.727	0.075*	2.420	0.336
10	5	4.573	0.183	4.752	0.061	5.220	0.048*
11	6	5.906	0.009	5.963	0.001*	5.676	0.105
12	3	2.780	0.049*	2.686	0.099	3.292	0.086
13	2	2.898	0.807	2.524	0.274	2.472	0.223*
14	0	0.958	0.918	0.393	0.155	0.216	0.047*
15	0	-0.094	0.009	-0.054	0.003	0.052	0.003*
16	0	-0.258	0.067	-0.732	0.535	0.168	0.028*
17	0	-0.067	0.004*	0.271	0.074	-0.235	0.055
18	0	0.183	0.033*	0.517	0.267	0.233	0.054
19	0	-0.139	0.019	0.117	0.014*	-0.151	0.023
合計	—	—	22.832	—	20.095	—	11.238

(注：残差²は $(Y_q - \hat{Y}_q)^2$ を意味し、* は予測による 3 つのモデルでの最小値を示す。)

5. おわりに

本研究では、ウォーターフォールモデルにおけるソフトウェア開発の最終段階であるテスト工程に入るまでに得られたソフトウェアプロジェクトデータを用いて、テスト工程の開始時に潜在するフォールト数の予測値を得るために、相関を考慮した RBF ネットワークによる予測モデルを提案した。そして残差平方和という評価基準において、先行研究で提案された予測モデルよりも予測精度の高い結果を得ることができた。チューニングによる重回帰分析による手法、予測によるニューラルネットワークと先行研究 [3] の RBF ネットワークによる手法の中で、最も予測精度の高かった予測モデルと比較し、本研究のモデルの改善率は 44% 向上した。

先行研究で提案された従来のモデルと比較して有用性のあるモデルを構築することができたと思うが、問題点もある。中間層のガウシアン関数のパラメータの初期値の設定である。初期値を決める際的设计理論は確立されているわけではなく、本研究ではこの値を実験的な経験により決めているため、設定した値が最適な値であるという判断が難しい。また、入力データやモデルの構造によって初期値を模索する必要がある。

そのため、構築したモデルごとに最適な初期値を自動で設定する仕組みを作ることで、さらに実用的な予測モデルを構築することができるはずである。これは先行研究でも問題とされていたが、本研究もこの問題を改善することができなかった。この問題に取り組み本手法を展開することで、今後のより良い情報化社会の発展に貢献できることを望んでいる。

参考文献

- [1] S. Yamada and J. Yamakawa, "An Empirical Study on Statistical Analysis Based on Software Process Monitoring Data with Initial Project Risks", *Proc. of 17th ISSAT Intern. Conf. on Reliability and Quality in Design*, Vancouver, Canada, pp. 384-388 (2011).
- [2] 嶋田遼平, 「ソフトウェア開発プロジェクトデータによる潜在フォールト数の予測 - コネクショニストアプローチ - 」, 法政大学大学院工学研究科, 平成 24 年度修士論文 (2013).
- [3] 稲垣寿一, 「RBF ネットワークを用いたソフトウェアメトリクスデータによる信頼性評価に関する研究」, 法政大学大学院工学研究科, 平成 25 年度修士論文 (2014).
- [4] M. J. D. Powell, "Radial basis function for multivariable interpolation: a review", *Algorithm for approximation*, Clarendon, Oxford (1987).
- [5] J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units", *Neural Computation*, Vol. 1, No. 2, pp. 281-294 (1989).
- [6] 淵田孝, 前原正和, 森邦彦, 村島定行「強化学習を用いた RBF ネットワークの学習」, 信学技報, 99-113, pp. 157-163 (2000-03).
- [7] 田口玄一, 真壁肇, 古林隆, 森雅夫, 「確率・統計」, 日本規格協会 (1998).
- [8] 嶋田遼平, 木村光宏, 「開発属性データを用いた GMDH によるソフトウェアの信頼性予測に関する一考察」, 信学技報, R-2012-2, pp. 5-10 (2012).