DOI: 10.3745/KIPSTD.2009.16D.5.729

# 유전 알고리즘 기반의 서포트 벡터 회귀를 이용한 소프트웨어 비용산정

권 기 태[†] · 박 수 권[††]


## 요 약

소프트웨어 공학에서 정확한 개발 비용 예측은 성공적인 개발 프로젝트를 위한 필수적인 요소로, 현재까지 많은 소프트웨어 비용산정을 위한 모델들이 개발되어 왔다. 전통적인 통계적 기법부터 기계학습을 적용한 알고리즘까지 다양한 분야의 아이디어를 접목하고 있다. 본 논문에서는 소프트웨어 개발 비용 예측을 위한 방법으로 유전 알고리즘과 서포트 벡터 머신의 회귀모델인 서포트 벡터 회귀를 결합한 GA-SVR 모델을 제안한다. 제안된 모델은 기존의 연구에 비해 향상된 결과를 보이고 있다.


**키워드 : 서포트 벡터 회귀, 소프트웨어 비용산정, 기계 학습, 유전 알고리즘, 매개변수**

# Estimation of software project effort with genetic algorithm and support vector regression

Ki-Tae Kwon[†] · SooKwon Park[††]


## ABSTRACT

The accurate estimation of software development cost is important to a successful development in software engineering. Until recent days, the model using regression analysis based on statistical algorithm and machine learning method have been used. However, this paper estimates the software cost using support vector regression, a sort of machine learning technique. Also, it finds the best set of optimized parameters applying genetic algorithm. The proposed GA-SVR model outperform some recent results reported in the literature.


**Keywords : Support Vector Regression, Software Cost Estimation, Machine Learning, Gentic Algorithm, Parameters**

## 1. 서 론

프로젝트의 정확한 비용예측은 성공적인 소프트웨어 개발에 중요한 역할을 하게 된다. 그러나 프로젝트 초기단계에는 정보가 한정되어 있어 개발비용을 예측하기가 매우 어렵다. 따라서 프로젝트 초기에 소프트웨어 개발비용을 예측하기 위한 많은 연구가 진행되고 있다[1-3]. 이러한 소프트웨어 비용산정에 대한 연구는 이전의 유사한 프로젝트에 대한 데이터를 기초로 데이터의 개념적 관계를 밝혀 유사 프로젝트의 노력을 예측하는데 있다. 데이터의 관계를 모델링하기 위해서 회귀 모델링과 같은 통계적 기법부터 신경망, 사례기반추론, 퍼지 논리 등을 이용한 기계학습 기법까지 다양한 논리적 접근 방법들이 연구되고 있다. 기계학습 접근법은 프로젝트 비용산정에 영향을 주는 특성들을 이용하여 개발비용과의 연관관계를 정의하고, 연관관계를 기반으로 차기 프로젝트들의 개발비용을 예측하게 된다.

Vapnik에 의해 제안된 서포트 벡터 머신(SVM)[4]은 뛰어난 일반화 능력으로, 패턴인식과 분류문제에 있어서 좋은 성과를 나타내고 있다. SVM은 아주 강력한 분류기로써, 올바른 인자를 얻으면 어떤 분류 기법보다 더 정확하고 더 잘 동작한다. 하지만 성능과 밀접한 관련이 있는 몇몇의 파라미터 값을 사용자 정의에 의존하게 되는데, 파라미터 값에 따른 성능 변화를 예측하기가 어렵다[5].

본 논문에서는 유전 알고리즘을 결합하여 SVM의 회귀모델인 서포트 벡터 회귀(SVR)에서 사용하는 사용자 정의 파라미터들의 최적 값을 예측하고, 실제 사례 데이터들을 이용하여 GA-SVR 결합 모델의 성능을 평가 하도록 한다.

† 종신회원 : 강릉대학교 컴퓨터공학과 교수
†† 준 회 원 : 강릉대학교 컴퓨터공학과 공학석사
논문접수 : 2009년 5월 22일
수 정 일 : 1차 2009년 6월 29일
심사완료 : 2009년 6월 29일

## 2. 연구의 배경 및 관련 연구

### 2.1 서포트 벡터 머신

SVM은 Vapnik에 의해 제안된 통계적 학습이론으로 두 범주를 갖는 객체들을 분류하는 방법이다[4]. 기존 학습 알고리즘은 학습 집단을 이용하여 학습 오류를 최소화하는 경험적 위험 최소화를 하는 반면, SVM은 분류 오류 확률을 최소화하는 구조적 위험 최소화[5] 방법에 기초하고 있다. SVM은 인공 신경망과 비슷한 수준의 높은 예측력을 나타낼 뿐만 아니라 인공신경망의 한계점으로 지적되었던 과대적합, 국소최적화와 같은 한계점들을 완화하는 장점을 가지고 있다[6].

#### 2.1.1 선형 서포트 벡터 머신

벡터$x_i$ 와 목표 값$y_i$ 로 이루어진 n개의 데이터 집합 G가 있다고 가정하자.

$$G=\{x_i, y_i\}^n,\quad x_i\in R^n,\quad y_i\in\{\text{class}: -1, +1\}$$

(그림1) 에서 G의 입력벡터들을 목표 값으로 분류할 수 있는 수많은 분리경계면 중 분리간격을 최대로 하는 초평면을 찾을 수 있다.

$$H(x): w^Tx+b=0,\quad \text{w}: \text{가중치벡터} \tag{1}$$

입력벡터들을 분리하기 위한 결정함수(H)는 (1)과 같다. G의 분리 가능한 모든 입력데이터는 (2)와 (3)을 만족하게 된다.

$$H_1: w\cdot x_i+b\geq+1,\quad \text{for } y_i=+1 \tag{2}$$
$$H_2: w\cdot x_i+b\geq-1,\quad \text{for } y_i=-1 \tag{3}$$

분리 가능한 선형 SVM에서 입력벡터는 결국 (2)와 (3)에 의해 분리된다.

$H_1$ 과 $H_2$ 사이의 최대거리는 다음과 같은 최적화 문제로 표현할 수 있다.

$$\text{Min}(||w||^2/2)$$
$$\text{Subject to } y_i(w\cdot x_i+b)\geq+1 \tag{4}$$

라그랑지 승수 $\alpha_i$ 를 이용하여 라그랑지 함수를 유도하면 다음과 같다.

$$L_P=\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i[y_i(w\cdot x_i+b)-1],$$
$$\alpha_i\geq 0, i=1,\ldots,N \tag{5}$$

(5)의 해는 (5.1)의 최적화 조건을 따른다.

$$\frac{\partial L_p}{\partial w}=0\ \Rightarrow\ w=\sum_{i=1}^{n}\alpha_i x_i y_i$$
$$\frac{\partial L_p}{\partial b}=0\ \Rightarrow\ \sum_{i=1}^{n}\alpha_i y_i=0 \tag{5.1}$$

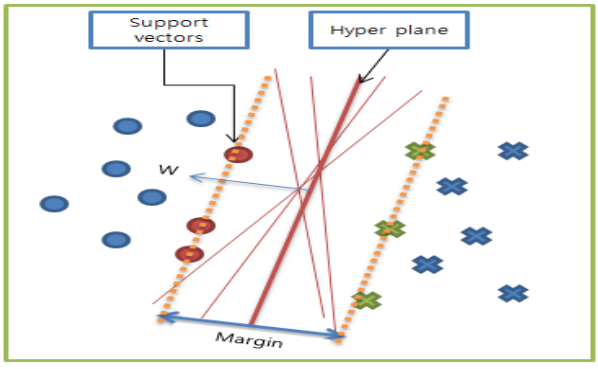
(5)는 쌍대문제를 최대화시키는 최적화 문제로 변환하여 해결할 수 있다.

$$L_P=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i\cdot x_j$$
$$\text{Subject to } \sum_{i=0}^{n}\alpha_i y_i=0,\ \alpha_i\geq 0,\ i=1,\ldots,N \tag{6}$$

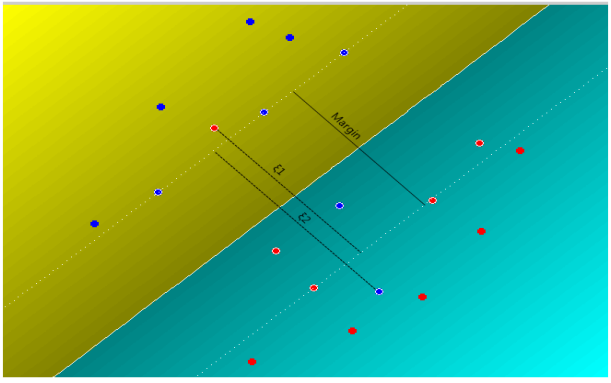
(6)으로부터 $\alpha_i$를 유도하게 되면 (5.1)을 이용하여 초평면 계수들을 결정할 수 있다. 하지만 실제 입력데이터의 경우 명확하게 선형 분류가 되지 않는 경우가 대부분으로 분리 불가능한 데이터가 존재하게 되므로 두 초평면 사이의 오분류를 허용하기 위하여 슬랙변수 $\xi$ 와 페널티 함수를 사용하게 된다.

페널티 함수를 포함한 최적화 문제(4)는 다음과 같다.

$$Min\frac{\|w\|^2}{2}+C\sum_{i=0}^{n}\xi_i$$
$$\text{Subject to } y_i(w\cdot x_i+b)\geq 1-\xi_i \tag{7}$$
$$\xi_i\geq 0$$




(그림 1) 선형 분류기




(그림 2) 노이즈를 갖는 선형 분류기

여기서 C는 최대거리 사이에 데이터가 놓이는 것에 대한 오차정도를 결정하는 역할을 하는 비용이다. 쌍대문제에서 $0 \le \alpha_i \le C$ 의 범위를 갖는다.

슬랙변수 $\xi$ 는 최대거리를 벗어난 오분류 데이터를 허용하기 위하여 사용되는 변수로써 올바른 분류 위치 사이의 간격으로 계산된다.

2.1.2 비선형 서포트 벡터 머신

입력 공간에 대한 비선형 분류경계를 학습하기 위해서 SVM의 결정함수로 매핑함수를 사용하여 정의한다.

$$H(x) : \Phi(x)^T w + b = 0$$

입력데이터 x를 새로운 특성 공간으로 매핑한 후 초평면을 구하게 된다.

비선형 SVM의 쌍대문제는 다음과 같다.

$$L_P = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \Phi(x_i)^T \Phi(x_j)$$
$$\text{Subject to } \sum_{i=0}^{n} \alpha_i y_i = 0 \ , 0 \le \alpha_i \le C \qquad (8)$$

특성 공간으로 매핑을 위해 SVM에서는 커널 함수를 사용하게 된다.

$$K(x, x_i) = \Phi(x)^T \Phi(x_i) \qquad (8.1)$$

2.2 서포트 벡터 회귀

분류를 위한 SVM 이론이 최근에는 회귀분석 및 함수추정으로 확장되어 많이 활용되고 있다[7].

SVM을 회귀 문제에 적용하여 훈련 데이터에 의존한 서포트 벡터 회귀(SVR) 예측모델을 만들 수 있다. SVR에서는 목표치 y와의 오차를 포함하기 위해 손실함수를 사용한다. SVR에서는 손실함수로 ε-Insensitive를 사용한다.

선형 SVR의 초평면은 앞의 2.1.1.절의 (7)을 사용하여 다음과 같은 최적화 문제로 정리된다.

$$Min \frac{\| w \|^2}{2} + C \sum_{i=0}^{n} \xi_i$$
$$\text{Subject to } \| y_i - (w \cdot x_i + b) \| \ge \epsilon + \xi_i \qquad (9)$$
$$\xi_i \ge 0$$

2.3 유전 알고리즘

유전 알고리즘은 John Holland에 의해 개발된 전역적 최적화 알고리즘으로, 자연의 생물유전을 모방한 연산자들을 반복적으로 적용하여 적합한 해를 탐색한다. 유전 알고리즘은 최고의 결과를 선택하기 위해 진화압력 아이디어를 사용한 최적화 알고리즘이다[8]. 유전 알고리즘은 개체군을 생성하여 개체의 적합도가 평가기준에 도달할 때까지 다음세대로의 진화를 반복하게 된다. 다음세대로 이어진 개체를 기준으로 하여 개체군을 생성하게 되는데, 돌연변이 및 교배를 통하여 새로운 유전자를 보유한 자식 개체들이 생성된다.

2.4 SVR을 이용한 소프트웨어 비용산정

SVR을 이용한 소프트웨어 비용산정은 훈련 데이터에 포함된 입력 특성들을 사용하여 공수(Y)를 추정하는 선형 또는 비선형 SVR모델을 만들고, 만들어진 SVR 모델을 이용하여 테스트 데이터의 비용산정에 활용하는 방법이다. 입력 특성으로 사용되는 비용산정 영향 요소로는 프로젝트의 규모, 개발되는 소프트웨어의 종류, 인적 요인, 프로그래밍 언어, 기능 점수 등 비용산정에 영향을 미칠 가능성이 있는 다양한 자료들이 수치화되어 사용되어 진다. SVR을 사용하는 소프트웨어 비용산정 모델의 이슈는 적합한 파라미터 값 선택에 있다. [5]에서는 SVR에 사용되는 사용자 정의 특성값을 선택하기 위해 임의의 파라미터 값으로 이루어진 다양한 순서쌍을 사용하여 SVR 모델을 실험하였고, 실험에 사용된 SVR 모델을 기반으로 훈련 데이터에 가장 적합한 파라미터 값을 선정하는 방법을 사용하였다. [4]에서는 파라미터 값을 더 효율적으로 찾는 방법을 제안하고 있다. 입력 데이터를 이용하여 기준 파라미터 집합을 설정하고 파라미터 집합을 중심으로 구간별 실험 포인트를 선택한다. 선택된 파라미터 집합을 이용한 SVR 결과값을 비교하여 최적의 파라미터 값을 선택하는 방식이다.

## 3. GA-SVR 알고리즘을 이용한 비용산정

SVR은 분류 최적화에 대한 일반화 능력이 뛰어난 반면, 데이터 셋에 따라 적합한 커널 변환함수와 함수의 파라미터 값이 달라 매번 이들을 찾아야 하는 문제가 있다[5].

사용자 정의에 의존하는 파라미터 값은 SVR의 성능에 있어서 매우 중요한 역할을 하게 된다. 정확한 비용예측을 위하여 사용자는 다양한 변경 값을 파라미터 값에 적용하게 되는데, SVR은 신경망과 같은 블랙박스 기법을 사용하고 있어, 파라미터의 증가, 감소에 따른 결과를 미리 예측하기가 어렵다.

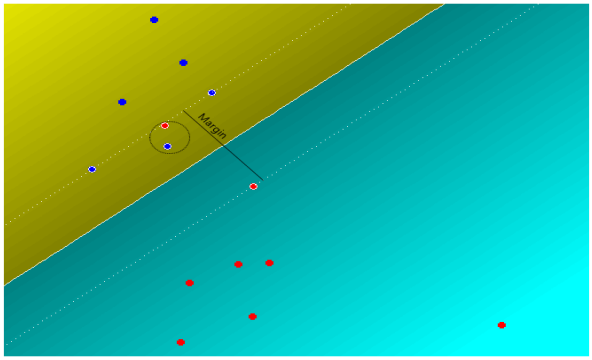
유전 알고리즘은 이러한 문제들 즉, 기존의 전통적 방법으로 좋은 해를 잘 구하지 못하는 경우에 특히 유용하다[8]. 따라서 본 논문에서는 소프트웨어 비용산정을 위해 SVR을 사용하고, SVR에서 사용되는 파라미터의 최적 값을 유전 알고리즘을 이용하여 구하고자 한다.

SVR에서 사용되는 사용자 정의에 의한 파라미터는 다음과 같다.

3.1 파라미터

3.1.1 파라미터 C

실제 사용되는 데이터는 잘못된 측정에 의한 에러 데이터를 포함하고 있는 경우가 일반적일 것이다. 에러 데이터로 인하여 선형분리가 불가능할 경우 가능해가 존재하지 않게




(그림 3) 노이즈를 갖는 최적 선형 분류기

된다. 이러한 경우에 올바른 분리경계면을 기대하기 위해서는 에러 데이터를 허용하는 것이 필요한데, Cortes는 슬랙변수 ξ 와 페널티 함수를 제안 하였다[9]. 이에 따른 최적화 문제는 앞의 2.1.1절의 (7)에서 설명한 바와 같다. 페널티 함수에 사용되는 상수 C는 초평면의 최대거리 사이에 데이터가 존재하는 것을 허용하되 이에 대한 페널티를 주기 위한 상수로서 0 이상의 값을 갖게 되며, 미리 결정되어야 한다. 상수 C에 따라 초평면의 방향과 서포트 벡터와의 최대거리에 영향을 미치게 된다. 또한 C 값에 의해 라그랑지 계수 $\alpha_i$ 는 $0 \le \alpha_i \le C$ 의 범위를 갖게 된다. C 값에 따른 초평면의 변화는 [9]에 잘 나타나 있다.

3.1.2 파라미터 ε

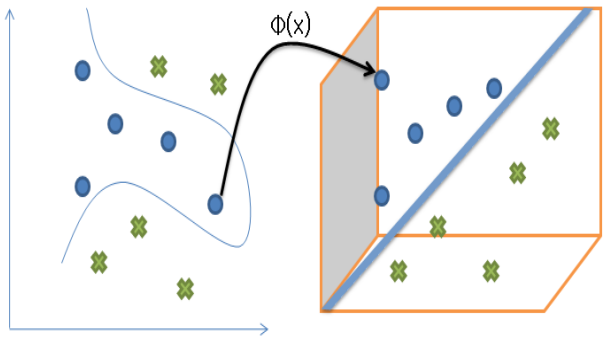
ε은 SVR에 사용되는 손실함수로써, 회귀모델의 경우 목표값 y 와 예측값 y′ 사이의 오차가 발생하게 된다. 손실함수는 이러한 예측오차의 임계치를 설정한다.

SVR에서는 손실함수로 ε-Insensitive를 사용한다[9].

3.1.3 비선형 커널함수의 파라미터

입력 데이터의 표현 공간에서 선형분류가 불가능한 경우, 입력데이터를 고차원의 특성 공간으로 매핑시켜 비선형 분리 문제를 해결한다.

앞장 2.1.2에서 설명한 바와 같이 비선형 분류 데이터를 분리하기 위해서는 비선형 함수를 이용하여 특성 공간으로 데이터를 이동해야 한다. (그림 4)에서 보면 두 데이터 집합은 선형 분리가 불가능하지만, 비선형 사상함수$\Phi(x)$에 의해 특성 공간으로 데이터를 이동하면 분리가 가능하다. 하지만 매핑함수$\Phi(x)$에 의해 이동된 특성 공간에서 데이터간의 내적을 구하는 것은 쉽지 않기 때문에 사상된 후 입력 데이터간의 내적을 구하기 위해 커널 함수를 이용한다.




(그림 4) 비선형 매핑

3.2 GA-SVR 알고리즘

본 논문에서는 소프트웨어 비용산정을 위한 방법으로 SVR을 제안한다. SVR은 손실함수인 ε-Insensitive와 페널티 함수를 사용하여 이상치에 민감하지 않고 그 영향을 최소화시키는 장점을 갖고 있다[7]. 하지만 앞의 3.1절에서 보인바와 같이 SVR의 성능에 중요한 영향을 미치는 파라미터 값은 훈련 데이터에 따라 사용자가 적절한 상수 값으로 설정해야 한다. 사용자가 입력하는 파라미터 값에 따라 그 결과치는 크게 달라지기 때문에 사용자는 훈련 데이터에 맞는 적절한 파라미터 값을 찾기 위해 여러 번의 반복적인 작업을 통해 파라미터 값을 선택하게 된다. 이 과정은 최적의 수치를 얻기 위해서 그리 효율적이지 못하다.

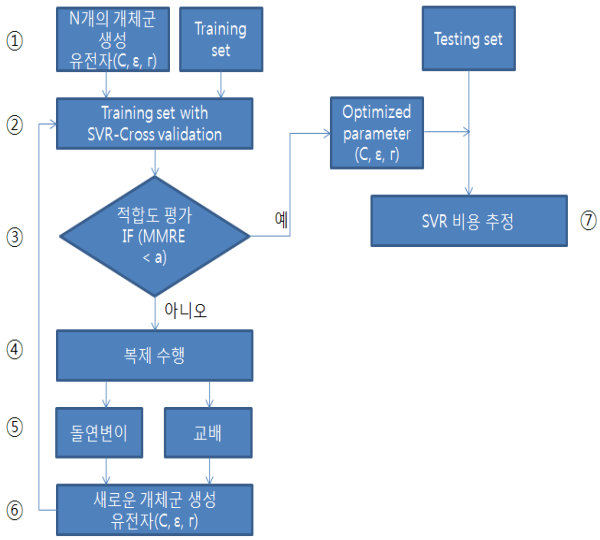
본 논문에서는 파라미터의 최적값을 찾고, 소프트웨어 비용산정의 성능을 높이기 위해서 유전 알고리즘을 사용한다. 유전 알고리즘을 통하여 훈련 데이터에 따른 적합한 SVR 파라미터 값을 최적화하고, 최적화된 파라미터 값을 SVR에 설정하여 소프트웨어 비용산정에 이용한다. 따라서 유전 알고리즘은 여러 가지 소프트웨어 프로젝트 데이터 집합에 맞는 최적의 SVR 파라미터 값을 검색하게 된다.

GA-SVR 알고리즘 수행순서는 다음과 같다.

① SVR 파라미터 개체군 생성 : 유전 알고리즘에 SVR 파라미터를 유전형질로 갖는 N개의 개체군을 생성.

② 개체군을 이용한 SVR 수행 : N개의 개체군에 포함된 개체를 파라미터 값으로 설정하여 SVR 교차검증 프로세서를 수행.

③ 적합도 평가 : 적합도 평가 결과 사용자가 만족하는




(그림 5) GA-SVR 알고리즘

결과가 나오면 해당 개체를 출력하고 그렇지 않으면 교차 검증으로 가장 좋은 결과 값을 나타낸 개체부터 우선순위를 설정.

④ 부모개체 선택 : 우선순위가 높은 N개의 개체는 다음 세대의 부모개체로 선택.

⑤ 자식개체 생성 : 선택된 부모개체를 이용한 돌연변이 및 교배 과정을 통하여 자식개체를 생성.

⑥ 새로운 개체군 생성 : 유전 알고리즘을 이용하여 세대를 진화시킴.

⑦ SVR 비용 추정 : 출력된 파라미터 값을 이용하여 테스트 데이터의 비용을 산정.

```
// GA-SVR 주요 함수
genome =                              // 유전자 생성
 {'gamma':XGeneA, 'eps':XGeneB, 'C':XGeneG }
def QPopulation() :                   // 개체군 생성
def fitness(self):                    // 적합도 평가
      alpha = self['gamma']
      beta  = self['eps']
      gamma = self['C']
      fitness = cross_validation(alpha,beta,gamma)
```

## 4. 실험 및 분석

### 4.1 데이터 수집

본 논문에서 제안된 알고리즘의 유용성을 확인하기 위해 이전의 연구에서 주로 사용된 2개의 데이터 집합을 활용한다.

첫 번째 데이터 집합은 소프트웨어 비용산정과 관련하여 [10-12]의 연구에서 사용된 NASA 소프트웨어 프로젝트 데이터 집합을 사용한다. 이 데이터 집합은 총 18개의 프로젝트로 구성되어 있으며 2개의 특성을 이용하여 공수(Y)를 추정하게 된다.

〈표 1〉 Desharnais 데이터 집합의 특성

| 특성 | 설명 | 모델 사용여부 |
|---|---|---|
| Team exp | 팀 개발 경험 기간 | X |
| Manager Exp | 프로젝트 관리자 경험기간 | X |
| Year End | 개발 기간 | O |
| Length | 규모 | X |
| Effort (Y) | 공수 | O |
| Transactions | 트랜잭션 수 | O |
| Entities | 엔티티 수 | O |
| PointAdjust | 조정 기능점수 | O |
| Adjustment | 조정 인자 | X |
| PointNonadjust | 미조정 기능점수 | O |

두 번째 데이터 집합은 [13-15]에서 사용된 Desharnais 데이터 집합[16]을 사용한다.

Desharnais 데이터 집합은 Canadian Software House-Commercial Projects로 구성된 81개의 프로젝트로 이루어져 있다. 이 중 부정확한 데이터가 포함된 4개의 프로젝트를 제외한 77개의 프로젝트를 사용하고 있으며 개발환경에 따라 3개의 부분집합들로 분리하여 비용산정에 활용하였다. 본 논문에서는 Desharnais 데이터 집합(77 프로젝트), Desharnais 데이터 집합 1(44 프로젝트), Desharnais 데이터 집합 2(23 프로젝트), Desharnais 데이터 집합 3(10 프로젝트)으로 총 4개의 데이터 집합을 이용하여 결과 측정을 한다.

### 4.2 구현 방법

GA-SVR 알고리즘은 Python을 이용하여 구현한다. SVR과 유전 알고리즘의 라이브러리는 Python 오픈 소프트웨어인 Libsvm과 Pygene를 사용하였다. 유전 알고리즘의 개체군은 20개를 생성하며, 개체군에는 100개의 자식을 만들고 그 중 20개를 선별한다. 돌연변이율은 0.1~0.5로 설정하였다.

### 4.3 평가 분석

본 논문에서 제안한 GA-SVR 비용산정 모델을 평가하기 위한 척도로는 MMRE(Mean Magnitude of Relative Error)와 PRED(25)를 사용하였다.

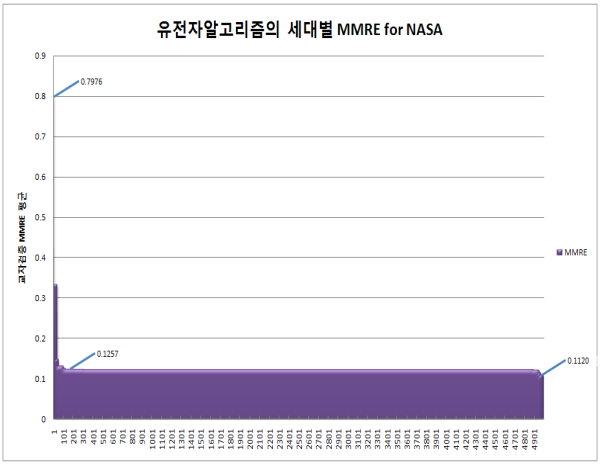
MMRE는 실제값과 추정값 사이의 상대적 오차의 평균 크기를 나타내며, 값이 작은 모델이 일반적으로 좋은 모델이다.

PRED(25)는 추정값이 실제값의 25% 범위내에 있는 비율이다. 좋은 모델 일수록 PRED(25) 수치는 높게 나타나게 된다.

SVR의 평가방법은 교차 검증 방법인 LOOCV(Leave One Out Cross Validation)를 사용한다. N개의 데이터셋을 이용할 경우, 이를 N개의 부분집합으로 나눈 후 (N-1)개의 부분집합을 훈련 데이터로 사용하고, 나머지 1개를 테스트 데이터로 사용하여 SVR 성능 평가를 수행한다. 이를 N번 반복하여 평균 MMRE와 PRED(25)를 구한다.

#### 4.3.1 NASA 데이터 집합

(그림 6)은 NASA 소프트웨어 프로젝트 집합을 이용한 GA-SVR의 5000세대까지의 진화과정을 나타낸 그림이다. SVR 커널함수는 RBF를 사용하였다. 132세대에서 베스트 개체는 MMRE=0.125, r=0.000433, ε=7.594912, C=78.174381의 수치를 나타냈지만, 이후 4895세대까지 진화하면서 MMRE는 0.001의 차이만을 줄였다. 5000세대에서는 MMRE=0.112, 파라미터는 r=0.000155, ε=1.794135, C=140.000854 로 나타났다. 커널함수를 Polynomial이나 Sigmoid로 교체하게 되면, 관련 파라미터에 따라 유전인자를 변경하게 된다. 결과는 RBF와 유사하게 나타났다.




(그림 6) NASA 데이터 집합의 GA-SVR 진화과정

<표 2>는 GA-SVR를 이용하여 5000세대까지의 진화과정을 거쳐 나타난 파라미터 값을 이용하여 SVR 비용산정을 수행한 결과이다.

NASA 소프트웨어 프로젝트 집합을 이용한 기존 연구 [9, 10]은 MMRE와 PRED(25)를 성능 실험에 사용하였다. <표 3>의 기존 연구와 <표 2>의 GA-SVR 결과 데이터를 비교하여 보면 제안된 모델이 0.053 이상 향상된 MMRE 수치를 확인할 수 있다.

〈표 2〉 NASA 데이터 집합의 추정

| 프로젝트 번호 | 실제값(Y) | 추정값(Y′) | MRE | PRED(25) |
|---|---|---|---|---|
| 1 | 115.8 | 116.7156 | 0.007907003 | O |
| 2 | 96 | 76.60652 | 0.202015417 | O |
| 3 | 79 | 77.03895 | 0.024823456 | O |
| 4 | 90.8 | 90.13819 | 0.007288689 | O |
| 5 | 39.6 | 41.0063 | 0.035512601 | O |
| 6 | 98.4 | 99.28492 | 0.0089931 | O |
| 7 | 18.9 | 19.50862 | 0.032201852 | O |
| 8 | 10.3 | 11.02801 | 0.070680388 | O |
| 9 | 28.5 | 28.26909 | 0.008102 | O |
| 10 | 7 | 7.540882 | 0.077268857 | O |
| 11 | 9 | 14.99179 | 0.665754667 | X |
| 12 | 7.3 | 9.379885 | 0.284915753 | X |
| 13 | 5 | 5.007066 | 0.0014132 | O |
| 14 | 8.4 | 8.083406 | 0.037689762 | O |
| 15 | 98.7 | 105.4985 | 0.0688808 | O |
| 16 | 15.6 | 14.75111 | 0.054416282 | O |
| 17 | 23.9 | 18.18695 | 0.239039749 | O |
| 18 | 138.3 | 112.2444 | 0.188399176 | O |
| RBF 파라미터(r=0.000155,ε=1.794135,C=140.000854) MMRE:0.11196126 PRED(25):88.89% | | | | |

〈표 3〉 기존 연구와의 비교

| 기존 연구 | 사용 모델 | MMRE (LOOCV) | PRED (25) |
|---|---|---|---|
| [11] | Radial Basis Function | 0.187 | 72.2% |
| [12] | SVR | 0.165 | 88.89% |

4.3.2 Desharnais 데이터 집합

<표 4>는 GA-SVR을 사용하여 Desharnais 데이터 집합의 비용산정에 적합한 커널함수와 파라미터 값을 유도하고 비용산정을 실시한 결과이다. 공통적으로 스케일을 적용한 입력 특성을 사용할 때 MMRE 수치는 좋아졌으며 SIGMOID 커널 함수를 사용할 때 좋은 결과를 나타냈다.

동일한 Desharnais 데이터 집합을 이용한 기존의 연구 결과 [13, 14]인 <표 5>와 <표 4>의 GA-SVR 결과 데이터를 비교하여 보면 본 연구에서 제안된 모델이 향상된 MMRE 및 PRED 수치를 확인할 수 있다.

〈표 4〉 GA-SVR 결과

| 데이터 집합 | 커널 함수 | 파라미터 | 세대 | MMRE (LOOCV) | PRED (25) |
|---|---|---|---|---|---|
| Desharnais Data Set (77 projects) | GAUSSIAN (Optimizing the Scale) | σ=0.6989, ε=1.8031, C=48642.22 | 65 | **0.5738** | 38.96% |
| | SIGMOID (Optimizing the Scale) | K=0.1759, ε=6.5486, C=11124.58, δ=0 | 61 | 0.5968 | **44.16%** |
| Desharnais Data Set 1 (44 projects) | GAUSSIAN | σ=61.7693, ε=4.7098, C=1224.39 | 32 | 0.3617 | **56.81%** |
| | SIGMOID (Optimizing the Scale) | K=0.1580, ε=1.4027, C=11671.92, δ=0 | 40 | **0.3184** | 50% |
| Desharnais Data Set 2 (23 projects) | RBF | r=0.000006, ε=0.3495, C=1230.312 | 1305 | 0.355 | 47.82% |
| | SIGMOID (Optimizing the Scale) | K=0.4956, ε=18.2388, C=1650.1457, δ=0 | 153 | **0.2590** | **65.22%** |
| Desharnais Data Set 3 (10 projects) | RBF | r=0.000003, ε=0.5985, C=1098.475 | 260 | 0.2466 | 70% |
| | SIGMOID (Optimizing the Scale) | **K=0.5041**, ε=15.7797, C=502.0405, δ=0 | 195 | **0.2234** | **70%** |

〈표 5〉 기존 연구와의 비교

| 데이터 집합 | MMRE (LOOCV) | PRED (25) |
|---|---|---|
| Desharnais 데이터 집합 (77 프로젝트) | 0.8348 | 28.57% |
| Desharnais 데이터 집합 1 (44 프로젝트) | 0.5627 | 31.82% |
| Desharnais 데이터 집합 2 (23 프로젝트) | 0.5977 | 26.09% |
| Desharnais 데이터 집합 3 (10 프로젝트) | 0.8644 | 0% |

## 5. 결론 및 향후 연구과제

본 논문에서는 소프트웨어 비용예측을 위해 서포트 벡터 회귀를 사용하였다. 서포트 벡터 회귀는 분류 문제에 있어서 뛰어난 일반화 능력을 보이는 반면, 데이터 집합에 따라 적합한 커널함수와 파라미터 값을 매번 찾아야 하는 단점이 있다. 이러한 단점을 보완하기 위해 본 논문에서는 유전 알고리즘을 이용하여 서포트 벡터 회귀에서 사용되는 파라미터 값을 최적화 시키는 방법을 제안하였다.

향후 연구과제로는 데이터 집합의 유형을 분석하여 알맞은 커널함수를 선택하도록 유도하는 방법을 연구하고, 입력 특성의 수가 많아지면 수행 속도에 영향을 많이 받게 되는데, 이러한 수행 속도 개선에 대한 연구 등이 필요하다. 또한 파라미터에 따른 MMRE 척도와 PRED(25) 척도를 기준으로 하는 정확도가 반드시 일치하지 않기 때문에 두 측정방식의 적합도를 효과적으로 만족시키는 파라미터를 찾기 위한 연구가 진행되어야 하겠다.

## 참 고 문 헌


[1] B. W. Boehm et al., "Software Development Cost Estimation Applications - a Survey", Annals of Software Engineering, Vol.10, No.1, pp.177-205, 2000.

[2] 권기태, 변분희, "소프트웨어 개발비 대가기준 개선에 관한 연구", 정보처리학회논문지D, 제13-D권, 제6호, pp.815-822, 2006.

[3] 변분희, 권기태, "소프트웨어 사업대가기준 보정계수의 유도 및 민감도 분석", 정보처리학회논문지D, 제15-D권, 제1호, pp.61-72, 2008.

[4] V.N.Vapnik, "The Nature of Statistical Learning theory", Springer-Verlag, 1995.

[5] Toby Segaran, "Programming Collective Intelligence", O'relly, 2007.

[6] Chih-Wei Hsu. Chih-Chung Chang and Chih-Jen Lin, "A Practical Guide to Support Vector Classication", http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2008.

[7] Changha Hwang, "Support Vector Median Regression", Data&Information Science Society, Vol.14, No.1, pp.67-74, 2003.

[8] 문병로, "Genetic Algorithm", 두양사, 2003.

[9] Steve Gunn, "Support Vector Machines for Classification and Regression", ISIS Technical Report, 1998.

[10] Alaa F. Sheta, "Estimation of the COCOMO Model Parameters Using Genetic Algorithms for NASA Software Projects", Journal of Computer Science 2, pp.118-123, 2006.

[11] Miyoung Shin and Amrit L. "Goel, Empirical Data Modeling in Software Engineering Using Radial Basis Functions", IEEE TSE, Vol.26, No.6, pp.567-576, 2000.

[12] Adriano L.I. Oliveira, "Estimation of Software Project Effort with Support Vector Regression", Neurocomputing, Vol.69, pp.1749-1753, 2006.

[13] Hojung Lim, "Support Vector Parameter Selection using Experimental Design Based Generating Set Search with Application to Predictive Software Data Modeling", PhD dissertation, Syracuse University, 2004.

[14] Hojung Lim and Amrit L. Goel, "Support Vector Machines for Data Modeling with Software Engineering Applications", Springer Handbook of Engineering Statistics, pp.1023-1037, 2006.

[15] Martin Shepperd and Chris Schofield, "Estimating Software Project Effort using Analogies", IEEE TSE, Vol.23, No.12, pp.736-743, 1997.

[16] J.M. Desharnais, "Analyse Statistique de la Productivitie des Projets Informatique a Partie de la Technique des Point des Fonction", Masters Thesis, Univ. of Montreal, 1989.



권 기 태

e-mail : ktkwon@kangnung.ac.kr

1986년 서울대학교 계산통계학과(학사)

1988년 서울대학교 계산통계학과(이학석사)

1993년 서울대학교 계산통계학과(이학박사)

1996년 Univ. of Southern California, Post-Doc.

1990년 9월~현 재 강릉대학교 컴퓨터공학과 교수

관심분야 : 소프트웨어 비용산정, 소프트웨어 메트릭스, 소프트웨어 아키텍처 등

박 수 권

e-mail : kokoo@kangnung.ac.kr
1999년 강릉대학교 수학과(학사)
2009년 강릉대학교 컴퓨터공학과 공학석사
관심분야 : 소프트웨어 비용산정, 소프트웨어 메트릭스, 소프트웨어 아키텍처 등