Ensemble learning

목차

목차

Introduction

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Stacking-based ensemble learning of decision trees for interpretable prostate cancer detection

Yuyan Wang [a], Dujuan Wang [b,*], Na Geng [c], Yanzhang Wang [a], Yunqiang Yin [d], Yaochu Jin [a,e]

[a] School of Management Science and Engineering, Dalian University of Technology, Dalian, 116023, China
[b] Business School, Sichuan University, Chengdu, 610064, China
[c] Department of Industrial Engineering & Logistics Management, Shanghai Jiao Tong University, Shanghai, 200240, China
[d] School of Management and Economics, University of Electronic Science and Technology of China, Chengdu, 611731, China
[e] Joint Laboratory for Artificial Intelligence for Precision Medicine, Jiaxing ACCB Diagnostics Ltd, Jiaxing, Zhejiang 314006, China

HIGHLIGHTS

- We propose a stacking-based interpretable selective ensemble learning method.
- We select ensemble models with accuracy and complexity under consideration.
- We combine selected effective models by random forest-based stacking.
- The proposed method is more accurate and interpretable in prostate cancer detection.
- We extract a few of effective diagnostic rules for clinical decision support.

ARTICLE INFO

Article history:
Received 4 June 2018
Received in revised form 3 January 2019
Accepted 16 January 2019
Available online 23 January 2019

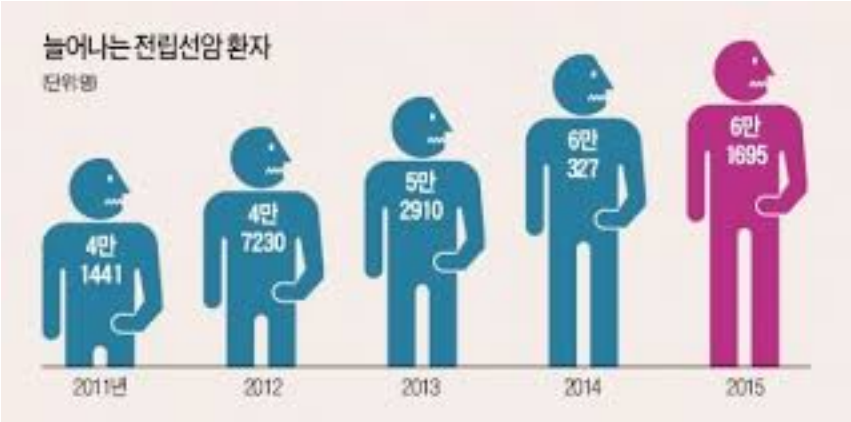
Keywords:
Prostate cancer detection
Ensemble learning
Stacking
Rule extraction
Multi-objective optimization

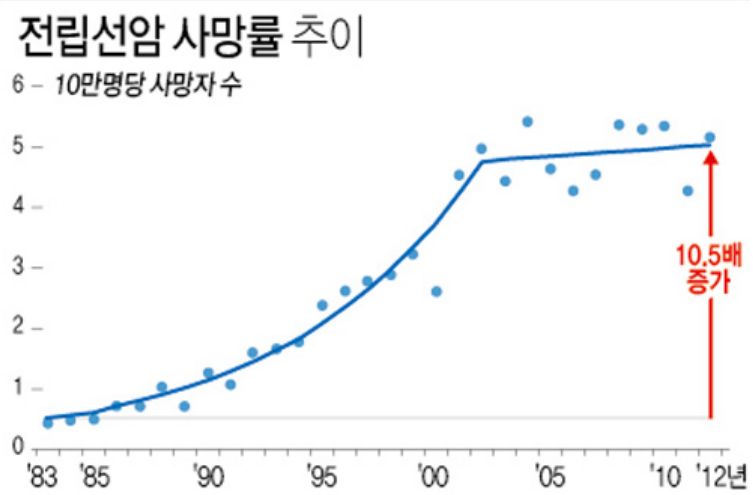
ABSTRACT

Prostate cancer is a highly incident malignant cancer among men. Early detection of prostate cancer is necessary for deciding whether a patient should receive costly and invasive biopsy with possible serious complications. However, existing cancer diagnosis methods based on data mining only focus on diagnostic accuracy, while neglecting the interpretability of the diagnosis model that is necessary for helping doctors make clinical decisions. To take both accuracy and interpretability into consideration, we propose a stacking-based ensemble learning method that simultaneously constructs the diagnostic model and extracts interpretable diagnostic rules. For this purpose, a multi-objective optimization algorithm is devised to maximize the classification accuracy and minimize the ensemble complexity for model selection. As for model combination, a random forest classifier-based stacking technique is explored for the integration of base learners, i.e., decision trees. Empirical results on real-world data from the General Hospital of PLA demonstrate that the classification performance of the proposed method outperforms that of several state-of-the-art methods in terms of the classification accuracy, sensitivity and specificity. Moreover, the results reveal that several diagnostic rules extracted from the constructed ensemble learning model are accurate and interpretable.

Introduction

- Prostate cancer (전립선 암)은 고위험군의 병으로 암 중에서 발병율이 높음
- 높은 사망률을 보이는 사회적으로 중요한 악성종양으로 조기 진단을 위한 노력이 필요함
- 전립선 암은 예측하기 어려운 병의 진행 행태를 보이기 때문에 예후를 통해 사전에 예측하는 것이 필요



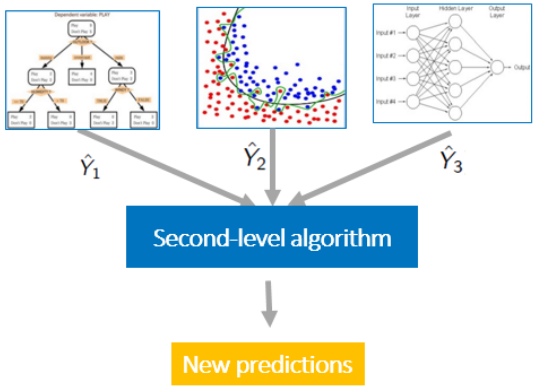


Introduction

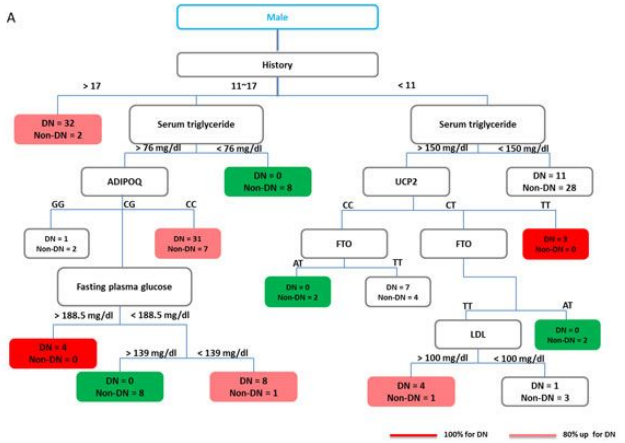
- 우수한 예측 성능을 보임과 동시에 해석이 가능한 기계학습기반의 모델개발은 필수 적임
- 본 논문에서는 이러한 문제 해결을 위해 ensemble learning 기반의 모델을 제안함

Classification performance

Extract effective diagnostic rules



[Ensemble learning – Stacking]



[Rule extraction method]

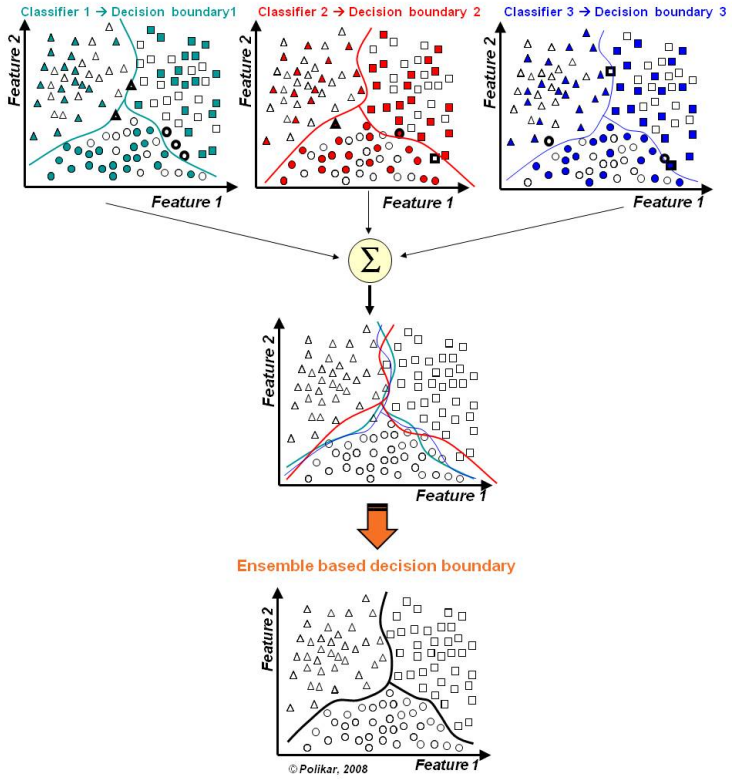
목차

Ensemble learning

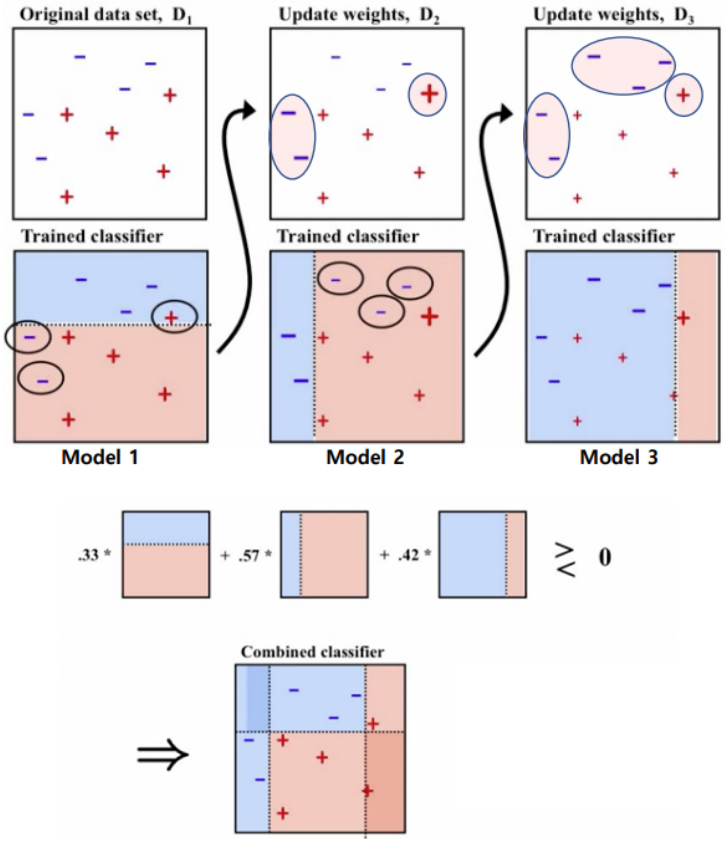
Ensemble learning

- Ensemble learning의 방법은 크게 bagging, boosting으로 구분됨

Bagging (Bootstrap Aggregating)

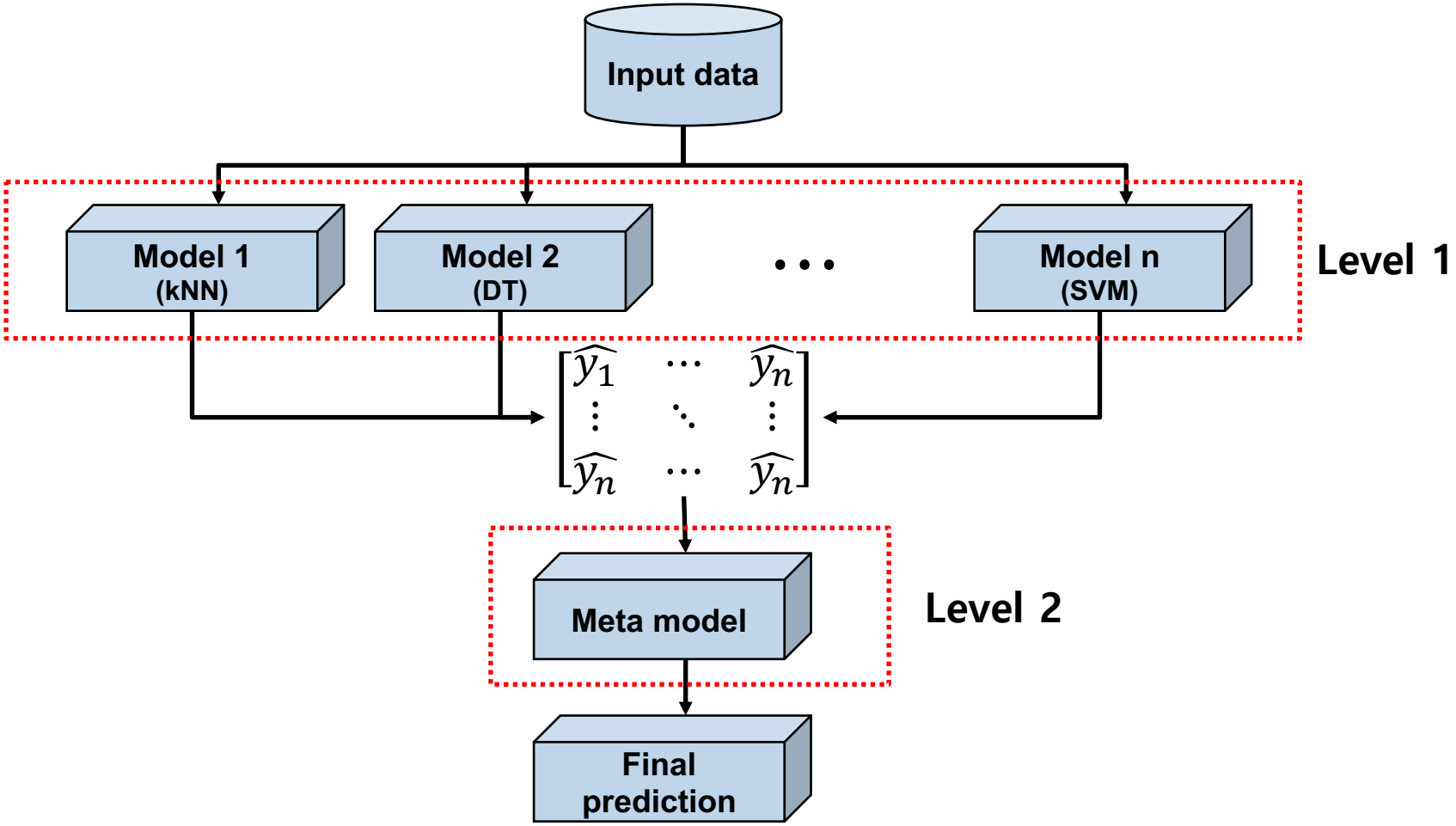


Boosting



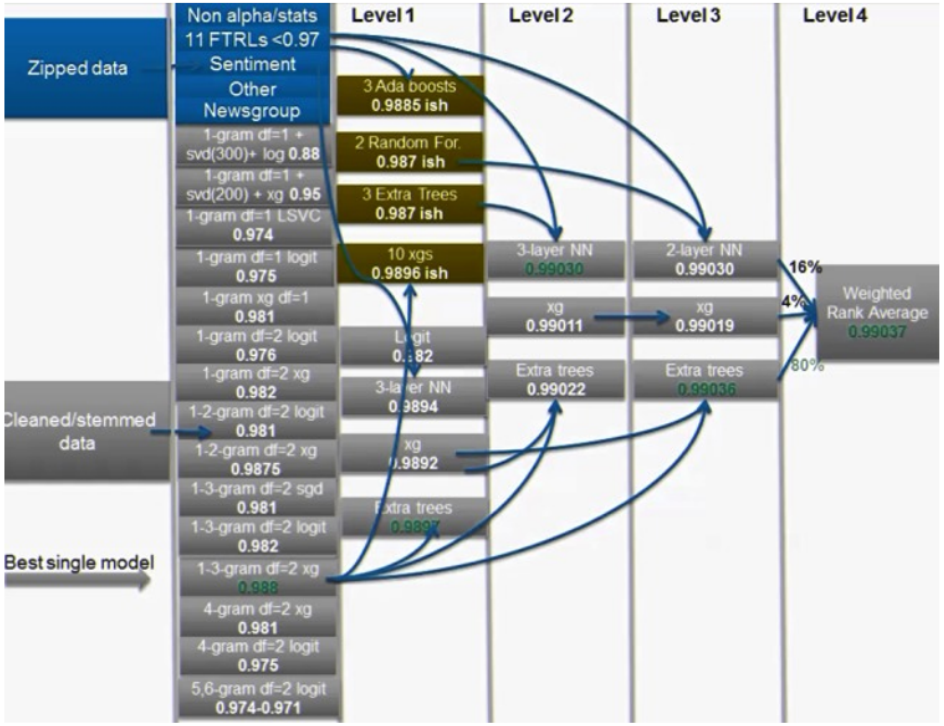
Ensemble learning

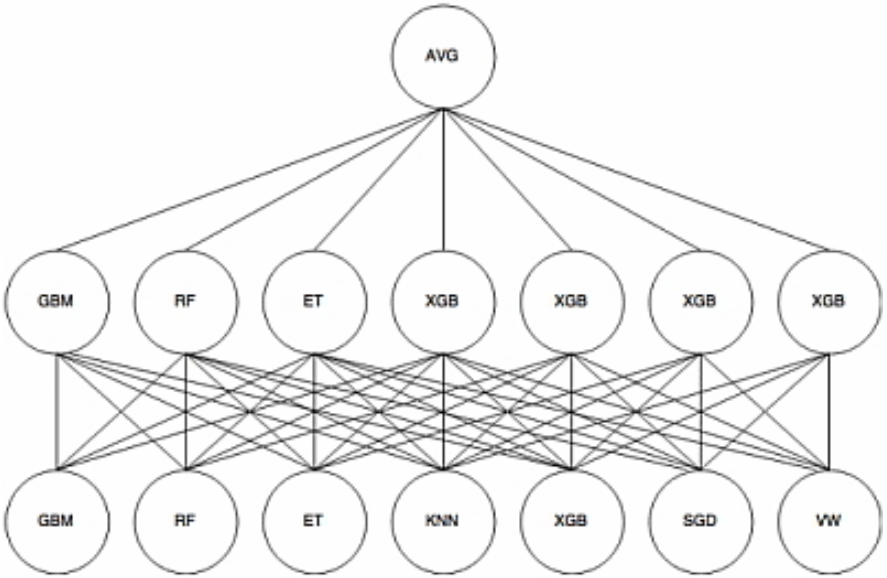
- 추가적으로, meta modeling 이라고 불리는 stacking 방법도 있음



Ensemble learning

- 다양한 형태의 stacking 방법이 존재
- 각 Node는 개별 모델로 구성되며, Node로 구성된 여러 층의 layer를 쌓는 형태





Ensemble learning

비교	Bagging (Bootstrap Aggregation)	Boosting	Stacking
특징	Overfitting 방지 가능 (majority voting, average)	Outlier에 취약	모델 간의 보완이 가능 연산량이 많음
학습 구조	병렬 앙상블 모델 (각 모델을 서로 독립적)	연속 앙상블 모델 (이전 모델의 오류를 고려)	병렬 앙상블 모델 (서로 다른 독립적인 모델)
목적	Variance 감소	Bias 감소	Performance 향상
대표 알고리즘	Random forest	Gradient Boosting Adaboost Xgboost Light GBM	* 학습의 구조가 있으며 특정 알고리즘은 없음
Sampling	Random sampling	Random sampling with weight on error	-

목차

Introduction

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Stacking-based ensemble learning of decision trees for interpretable prostate cancer detection

Yuyan Wang [a], Dujuan Wang [b,*], Na Geng [c], Yanzhang Wang [a], Yunqiang Yin [d], Yaochu Jin [a,e]

[a] School of Management Science and Engineering, Dalian University of Technology, Dalian, 116023, China
[b] Business School, Sichuan University, Chengdu, 610064, China
[c] Department of Industrial Engineering & Logistics Management, Shanghai Jiao Tong University, Shanghai, 200240, China
[d] School of Management and Economics, University of Electronic Science and Technology of China, Chengdu, 611731, China
[e] Joint Laboratory for Artificial Intelligence for Precision Medicine, Jiaxing ACCB Diagnostics Ltd, Jiaxing, Zhejiang 314006, China

HIGHLIGHTS

- We propose a stacking-based interpretable selective ensemble learning method.
- We select ensemble models with accuracy and complexity under consideration.
- We combine selected effective models by random forest-based stacking.
- The proposed method is more accurate and interpretable in prostate cancer detection.
- We extract a few of effective diagnostic rules for clinical decision support.

ARTICLE INFO

Article history:
Received 4 June 2018
Received in revised form 3 January 2019
Accepted 16 January 2019
Available online 23 January 2019

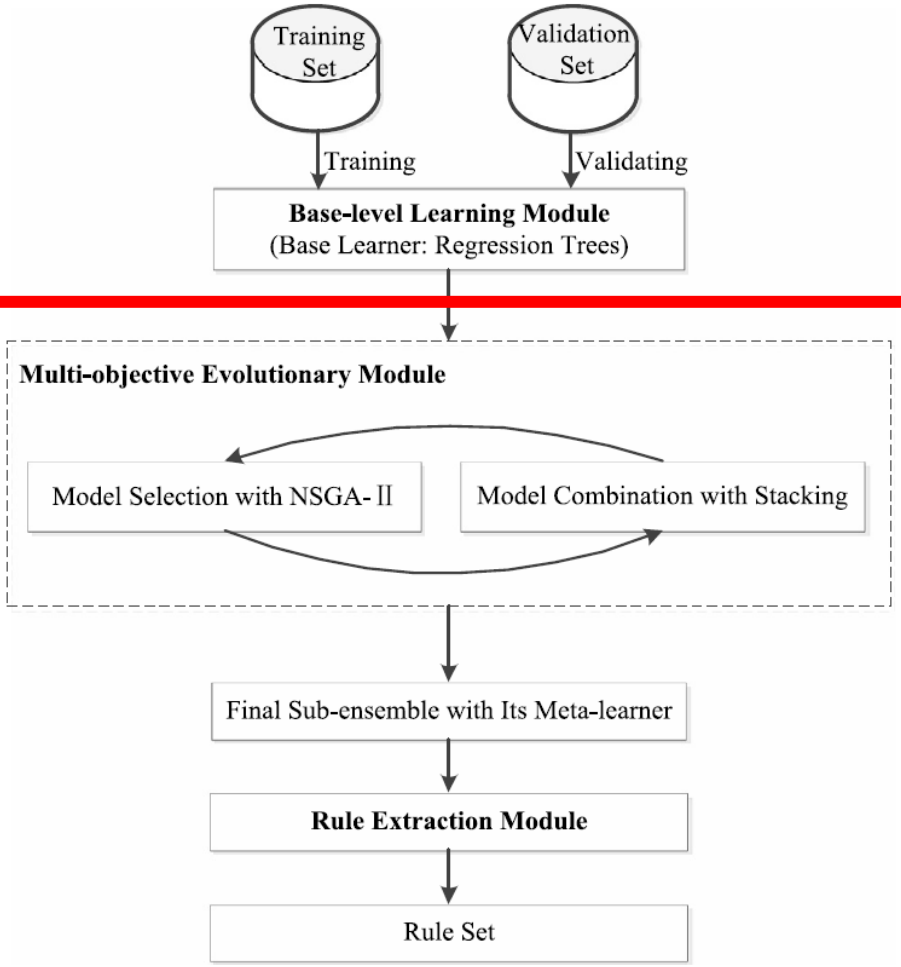
Keywords:
Prostate cancer detection
Ensemble learning
Stacking
Rule extraction
Multi-objective optimization

ABSTRACT

Prostate cancer is a highly incident malignant cancer among men. Early detection of prostate cancer is necessary for deciding whether a patient should receive costly and invasive biopsy with possible serious complications. However, existing cancer diagnosis methods based on data mining only focus on diagnostic accuracy, while neglecting the interpretability of the diagnosis model that is necessary for helping doctors make clinical decisions. To take both accuracy and interpretability into consideration, we propose a stacking-based ensemble learning method that simultaneously constructs the diagnostic model and extracts interpretable diagnostic rules. For this purpose, a multi-objective optimization algorithm is devised to maximize the classification accuracy and minimize the ensemble complexity for model selection. As for model combination, a random forest classifier-based stacking technique is explored for the integration of base learners, i.e., decision trees. Empirical results on real-world data from the General Hospital of PLA demonstrate that the classification performance of the proposed method outperforms that of several state-of-the-art methods in terms of the classification accuracy, sensitivity and specificity. Moreover, the results reveal that several diagnostic rules extracted from the constructed ensemble learning model are accurate and interpretable.

Proposed methodology

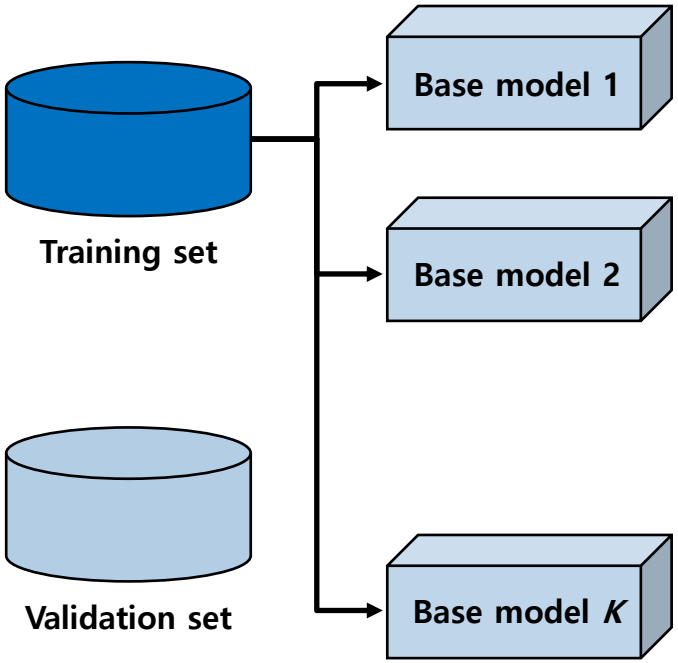
Step1



Proposed methodology

Step1 base learner

- Base leaner는 meta-data를 생성하기 위한 절차
- Training set을 활용하여 다수의 Tree를 생성하고 이를 base leaner로 활용
- 학습의 다양성을 확보 **Bootstrap sampling / Random feature selection**

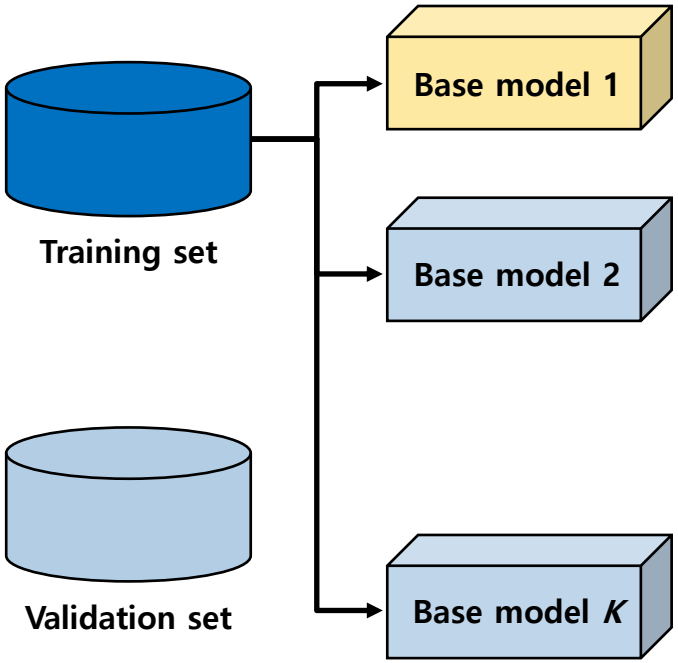


	V1	V2	V3	…	…	Vm-1	Vm
1							
2							
⋮							
n-2							
n-1							
n							

Proposed methodology

Step1 base learner

- Base leaner는 meta-data를 생성하기 위한 절차
- Training set을 활용하여 다수의 Tree를 생성하고 이를 base leaner로 활용
- 학습의 다양성을 확보 **Bootstrap sampling / Random feature selection**

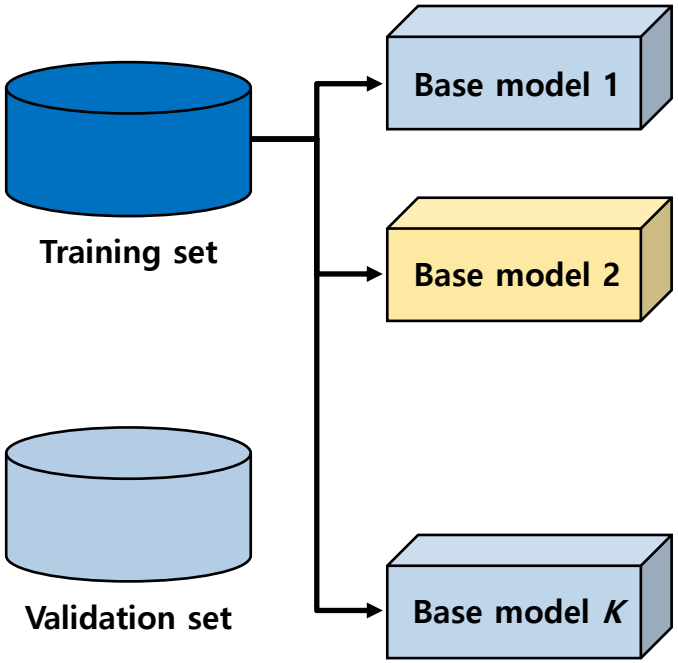


	V1	V2	V3	…	…	Vm-1	Vm
1							
2							
⋮							
n-2							
n-1							
n							

Proposed methodology

Step1 base learner

- Base leaner는 meta-data를 생성하기 위한 절차
- Training set을 활용하여 다수의 Tree를 생성하고 이를 base leaner로 활용
- 학습의 다양성을 확보 **Bootstrap sampling / Random feature selection**

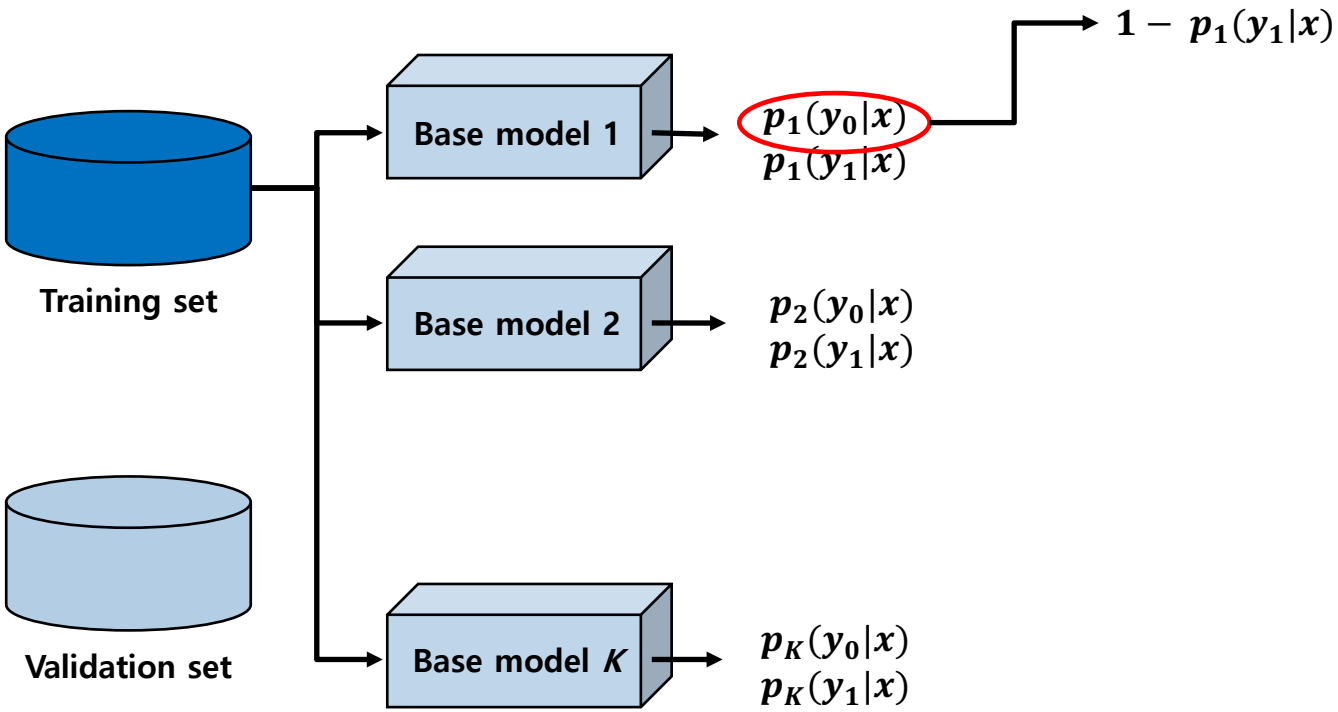


	V1	V2	V3	…	…	Vm-1	Vm
1							
2							
⋮							
n-2							
n-1							
n							

Proposed methodology

Step1 base learner

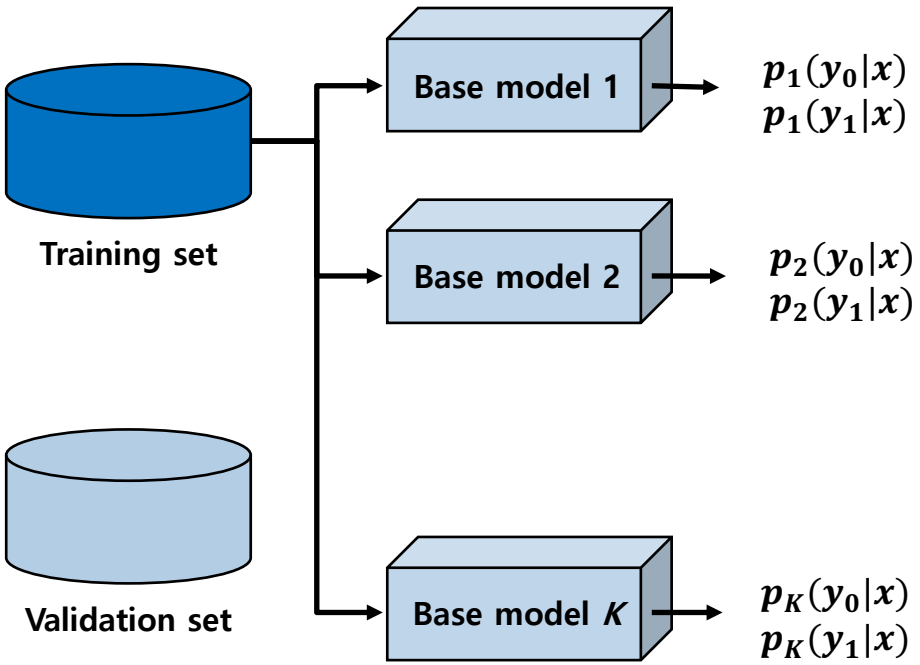
- Base leaner는 meta-data를 생성하기 위한 절차
- Regression tree를 활용하여 base leaner를 구성
- Target variable은 binary classification이므로, 예측된 값은 [0, 1]



Proposed methodology

Step1 base learner

- Base leaner는 meta-data를 생성하기 위한 절차
- Regression tree를 활용하여 base leaner를 구성
- Target variable은 binary classification이므로, 예측된 값은 [0, 1]



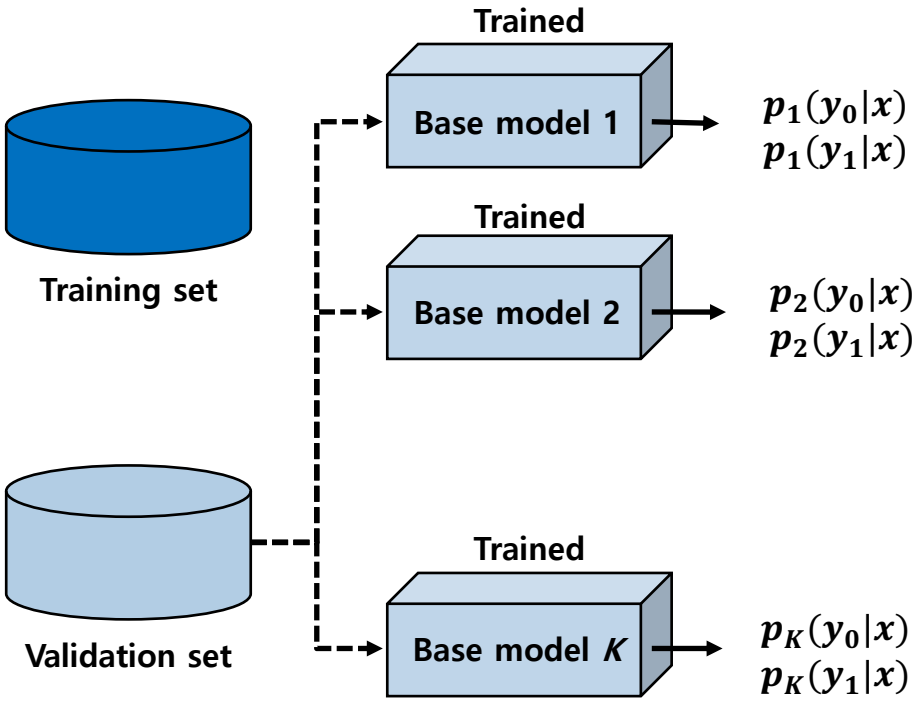
Meta-data of training set

$$\mathbf{L} := \begin{bmatrix} p_1(y_1|\boldsymbol{x}_1) & p_1(y_2|\boldsymbol{x}_1) & \cdots & p_K(y_1|\boldsymbol{x}_1) & p_K(y_2|\boldsymbol{x}_1) \\ p_1(y_1|\boldsymbol{x}_2) & p_1(y_2|\boldsymbol{x}_2) & \cdots & p_K(y_1|\boldsymbol{x}_2) & p_K(y_2|\boldsymbol{x}_2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ p_1(y_1|\boldsymbol{x}_{N_{tr}}) & p_1(y_2|\boldsymbol{x}_{N_{tr}}) & \cdots & p_K(y_1|\boldsymbol{x}_{N_{tr}}) & p_K(y_2|\boldsymbol{x}_{N_{tr}}) \end{bmatrix}$$

Proposed methodology

Step1 base learner

- Base leaner는 meta-data를 생성하기 위한 절차
- Regression tree를 활용하여 base leaner를 구성
- Target variable은 binary classification이므로, 예측된 값은 [0, 1]



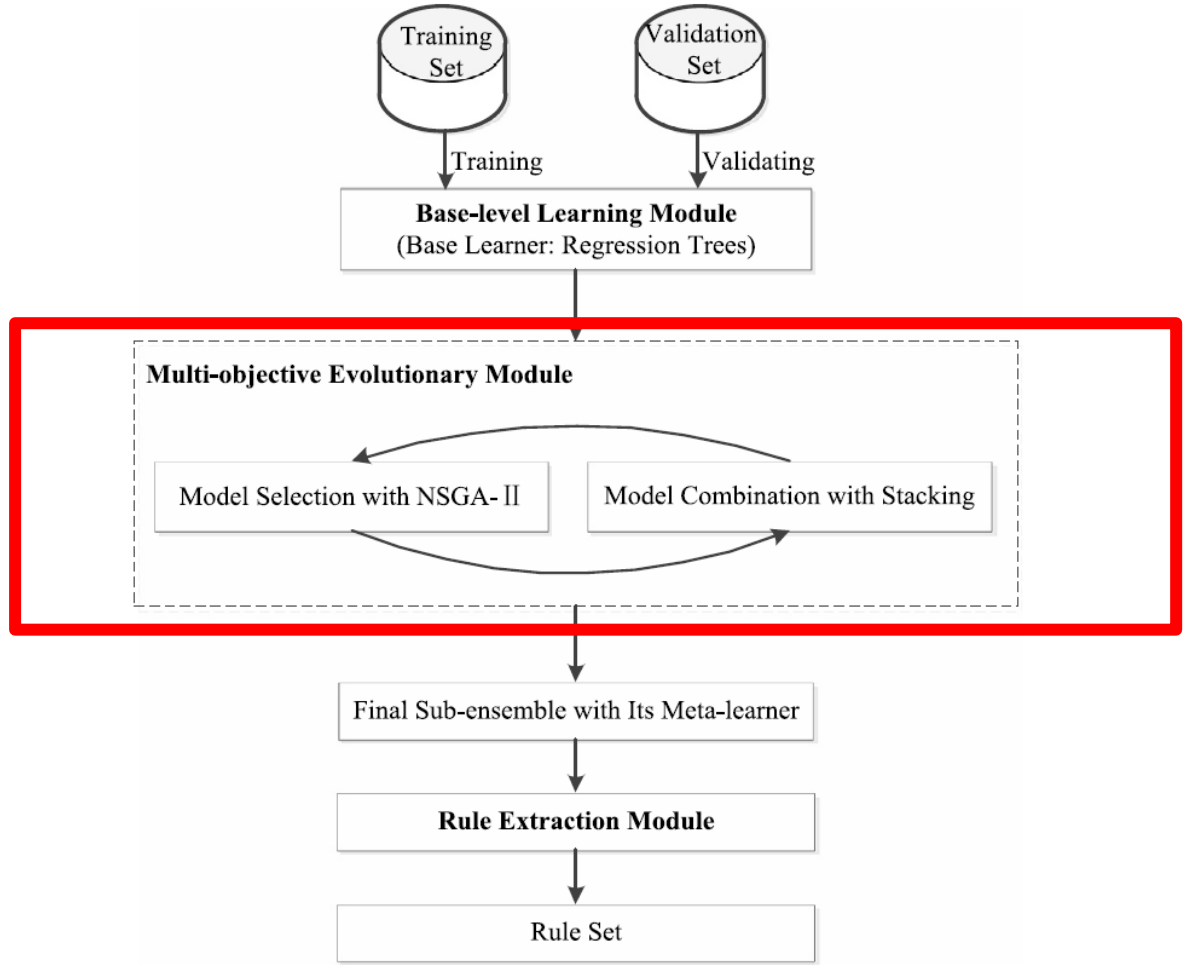
Meta-data of training set

$$\mathbf{L} := \begin{bmatrix} p_1(y_1|\boldsymbol{x}_1) & p_1(y_2|\boldsymbol{x}_1) & \cdots & p_K(y_1|\boldsymbol{x}_1) & p_K(y_2|\boldsymbol{x}_1) \\ p_1(y_1|\boldsymbol{x}_2) & p_1(y_2|\boldsymbol{x}_2) & \cdots & p_K(y_1|\boldsymbol{x}_2) & p_K(y_2|\boldsymbol{x}_2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ p_1(y_1|\boldsymbol{x}_{N_{tr}}) & p_1(y_2|\boldsymbol{x}_{N_{tr}}) & \cdots & p_K(y_1|\boldsymbol{x}_{N_{tr}}) & p_K(y_2|\boldsymbol{x}_{N_{tr}}) \end{bmatrix}$$

Meta-data of validation set

$$\mathbf{L} := \begin{bmatrix} p_1(y_1|\boldsymbol{x}_1) & p_1(y_2|\boldsymbol{x}_1) & \cdots & p_K(y_1|\boldsymbol{x}_1) & p_K(y_2|\boldsymbol{x}_1) \\ p_1(y_1|\boldsymbol{x}_2) & p_1(y_2|\boldsymbol{x}_2) & \cdots & p_K(y_1|\boldsymbol{x}_2) & p_K(y_2|\boldsymbol{x}_2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ p_1(y_1|\boldsymbol{x}_{N_{tr}}) & p_1(y_2|\boldsymbol{x}_{N_{tr}}) & \cdots & p_K(y_1|\boldsymbol{x}_{N_{tr}}) & p_K(y_2|\boldsymbol{x}_{N_{tr}}) \end{bmatrix}$$

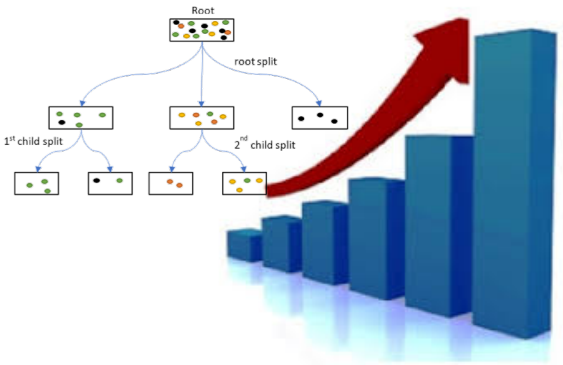
Proposed methodology



Step2

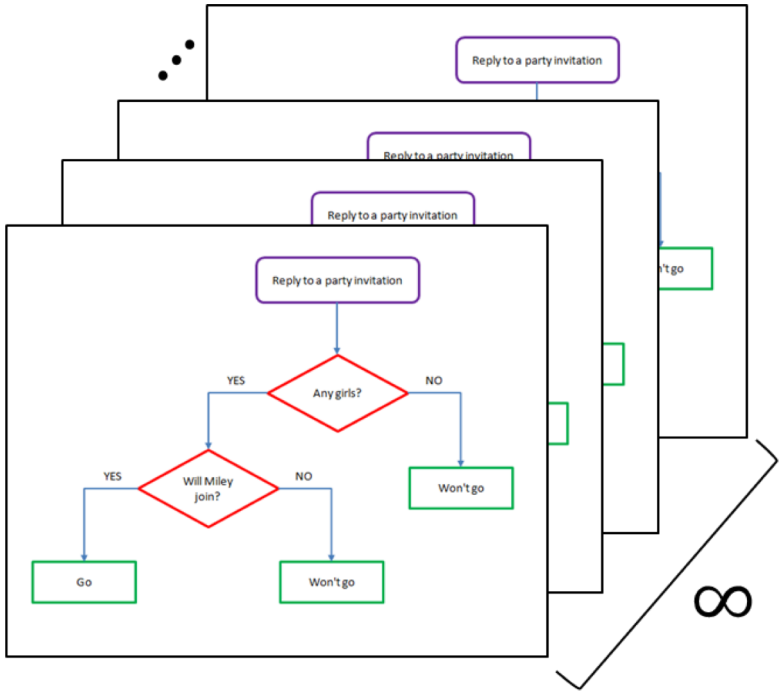
Proposed methodology

Step2 optimal sub-ensemble selection



Classification performance

≠



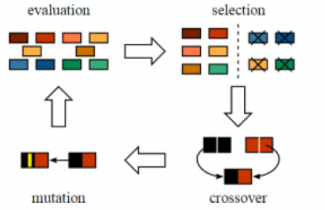
Robustness

Proposed methodology

Step2 optimal sub-ensemble selection

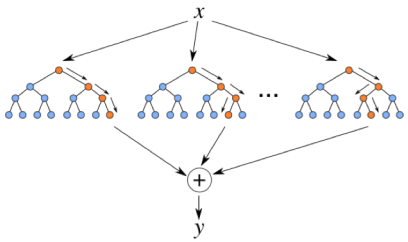
Accuracy ↑ **Complexity** ↓

Q1. Which model we should choose?



Multi-objective evolutionary process

Q2. How to combine the chosen models?



Random Forest

Proposed methodology

Step2 optimal sub-ensemble selection

- 최적의 sub-ensemble을 선택하기 위해 bi-objective optimization problem을 정의
- Elitist non-dominated sorting genetic algorithm (**NSGA-II**)를 활용하여 best sub-ensemble을 선택

$$Maximize\ Acc_{gp} = 1 - \frac{1}{N_{te}} \sum_{j=1}^{Nte} \left[E_{gp}(x_j) - y_j \right]^2$$

$$Minimize\ Complexity_{gp} = \sum_{k=1}^{K} Ind_{gp}(k)$$

gp : the pth individual of the population in gth generation
k : the number of base learner
$E_{gp}(x_j)$: the predicted label of the jth testing sample

Proposed methodology

Step2 optimal sub-ensemble selection

Sub-ensemble 1	Sub-ensemble 2	Sub-ensemble 3	…	Sub-ensemble P-2	Sub-ensemble P-1	Sub-ensemble P

Chromosome

Base leaner 1	Base leaner 2	Base leaner 3	Base leaner 4	Base leaner 5	…	Base leaner K
1	0	0	1	1	…	1

$L =$

Base leaner 1		Base leaner 2		…	Base leaner k-1		Base leaner k	
$p_1(y_0\|x_1)$	$p_1(y_1\|x_1)$	$p_2(y_0\|x_1)$	$p_2(y_1\|x_1)$	…	$p_{k-1}(y_0\|x_1)$	$p_{k-1}(y_1\|x_1)$	$p_k(y_0\|x_1)$	$p_k(y_1\|x_1)$
$p_1(y_0\|x_2)$	$p_1(y_1\|x_2)$	$p_2(y_0\|x_2)$	$p_2(y_1\|x_2)$		$p_{k-1}(y_0\|x_2)$	$p_{k-1}(y_1\|x_2)$	$p_k(y_0\|x_2)$	$p_k(y_1\|x_2)$
⋮		⋮		⋮	⋮		⋮	
$p_1(y_0\|x_{n-1})$	$p_1(y_1\|x_{n-1})$	$p_2(y_0\|x_{n-1})$	$p_2(y_1\|x_{n-1})$		$p_{k-1}(y_0\|x_{n-1})$	$p_{k-1}(y_1\|x_{n-1})$	$p_k(y_0\|x_{n-1})$	$p_k(y_1\|x_{n-1})$
$p_1(y_0\|x_n)$	$p_1(y_1\|x_n)$	$p_2(y_0\|x_n)$	$p_2(y_1\|x_n)$	…	$p_{k-1}(y_0\|x_n)$	$p_{k-1}(y_1\|x_n)$	$p_k(y_0\|x_n)$	$p_k(y_1\|x_n)$

Proposed methodology

Step2 optimal sub-ensemble selection

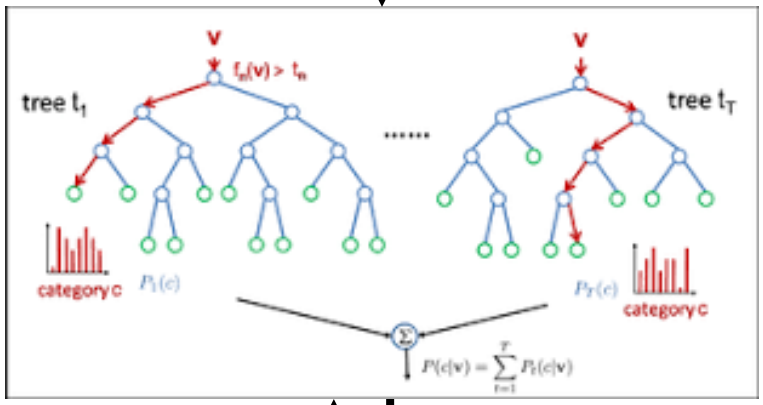
Base leaner 1		⋯	Base leaner k	
$p_1(y_0\|x_1)$	$p_1(y_1\|x_1)$	…	$p_k(y_0\|x_1)$	$p_k(y_1\|x_1)$
$p_1(y_0\|x_2)$	$p_1(y_1\|x_2)$		$p_k(y_0\|x_2)$	$p_k(y_1\|x_2)$
⋮		⋮	⋮	
$p_1(y_0\|x_{n-1})$	$p_1(y_1\|x_{n-1})$		$p_k(y_0\|x_{n-1})$	$p_k(y_1\|x_{n-1})$
$p_1(y_0\|x_n)$	$p_1(y_1\|x_n)$	…	$p_k(y_0\|x_n)$	$p_k(y_1\|x_n)$

[Meta-data of Training set]

Base leaner 1		⋯	Base leaner k	
$p_1(y_0\|x_1)$	$p_1(y_1\|x_1)$	…	$p_k(y_0\|x_1)$	$p_k(y_1\|x_1)$
$p_1(y_0\|x_2)$	$p_1(y_1\|x_2)$		$p_k(y_0\|x_2)$	$p_k(y_1\|x_2)$
⋮		⋮	⋮	
$p_1(y_0\|x_{n-1})$	$p_1(y_1\|x_{n-1})$		$p_k(y_0\|x_{n-1})$	$p_k(y_1\|x_{n-1})$
$p_1(y_0\|x_n)$	$p_1(y_1\|x_n)$	…	$p_k(y_0\|x_n)$	$p_k(y_1\|x_n)$

[Meta-data of Validation set]

Training RF

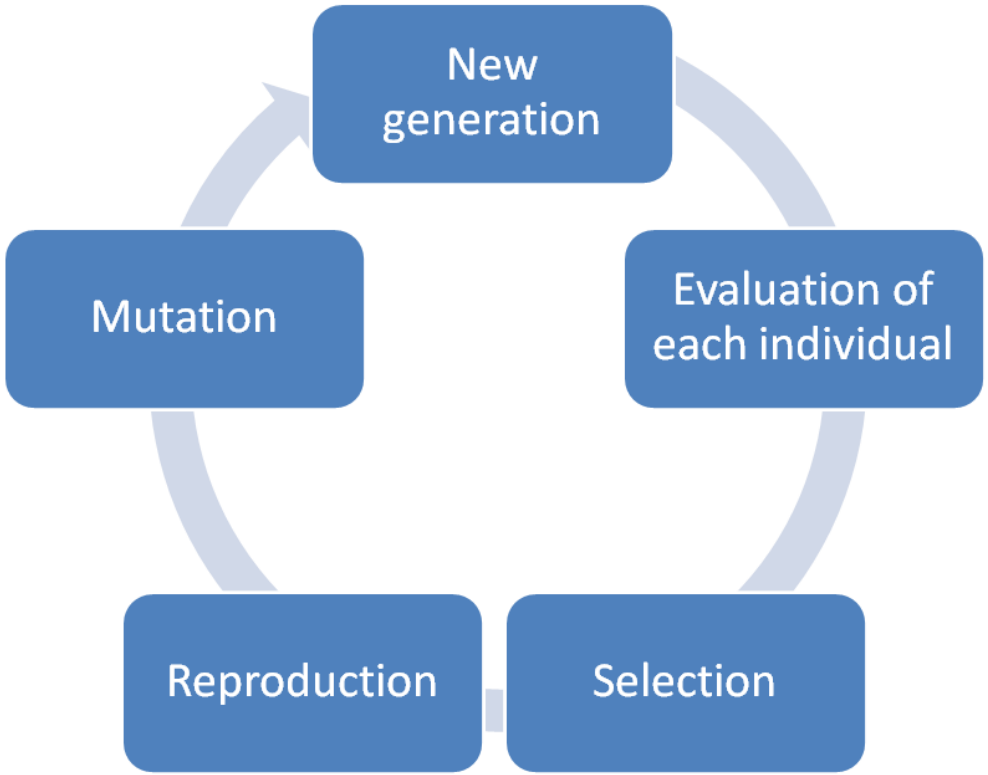


Prediction

"0" or "1"

Proposed methodology

Step2 optimal sub-ensemble selection



Proposed methodology

Step2 optimal sub-ensemble selection

Sub-ensemble 1	Sub-ensemble 2	Sub-ensemble 3	···	Sub-ensemble P-2	Sub-ensemble P-1	Sub-ensemble P

$$Maximize\ Acc_{gp} = 1 - \frac{1}{N_{te}} \sum_{j=1}^{Nte} [E_{gp}(x_j) - y_j]^2$$

$$Minimize\ Complexity_{gp} = \sum_{k=1}^{K} Ind_{gp}(k)$$

G번 반복 수행

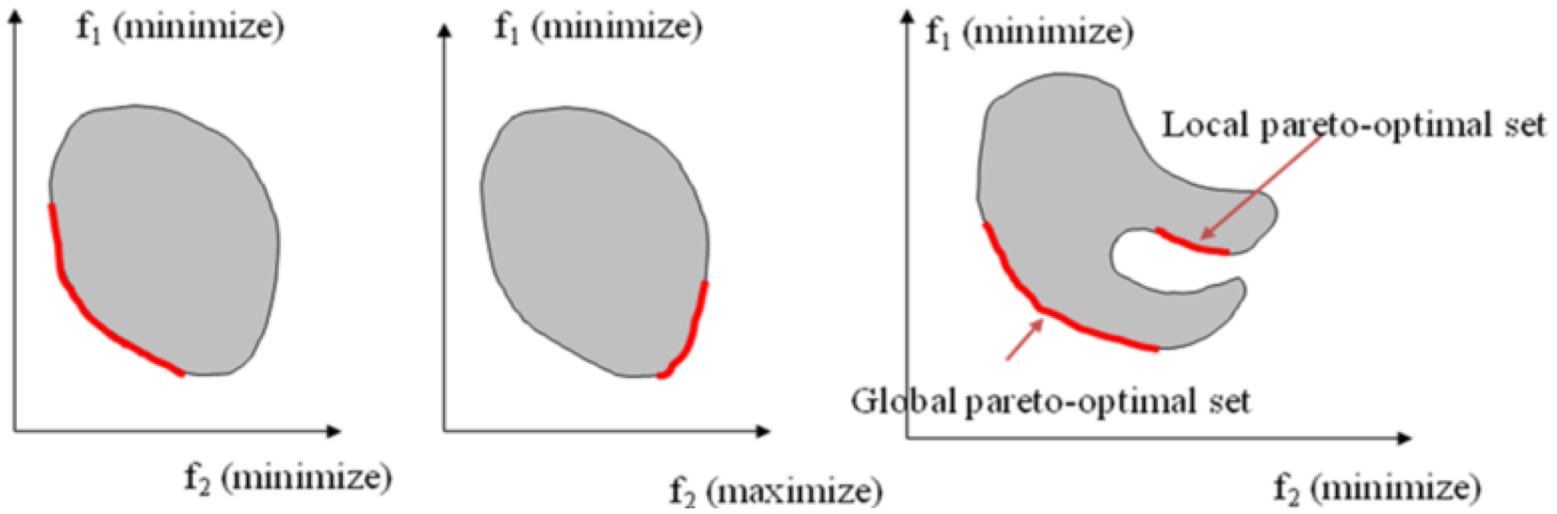
Pereto-optimal set

Sub-ensemble G1	Sub-ensemble G2	Sub-ensemble G3	···	Sub-ensemble GP-2	Sub-ensemble GP-1	Sub-ensemble GP

Multi objective evolutionary process

Multi object

- 목적함수가 여러 개 존재할 경우, 목적 함수 사이에 trade-off 관계가 존재
- 다수의 목적함수를 모두 만족하는 값은 하나가 아니라 여러 개가 존재할 수 있음
- 최적의 해의 집합을 찾는 것을 목적으로 함 (pareto-optimal set)

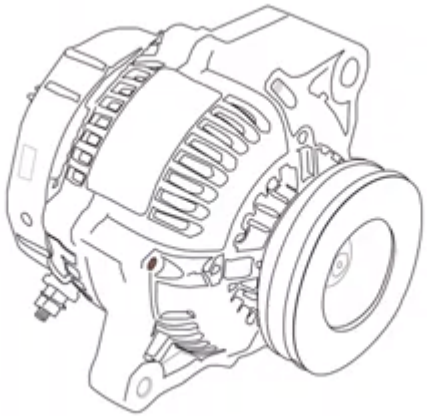


Multi objective evolutionary process

Multi object example

Goals:
Minimal acceleration time
Maximal range

Size

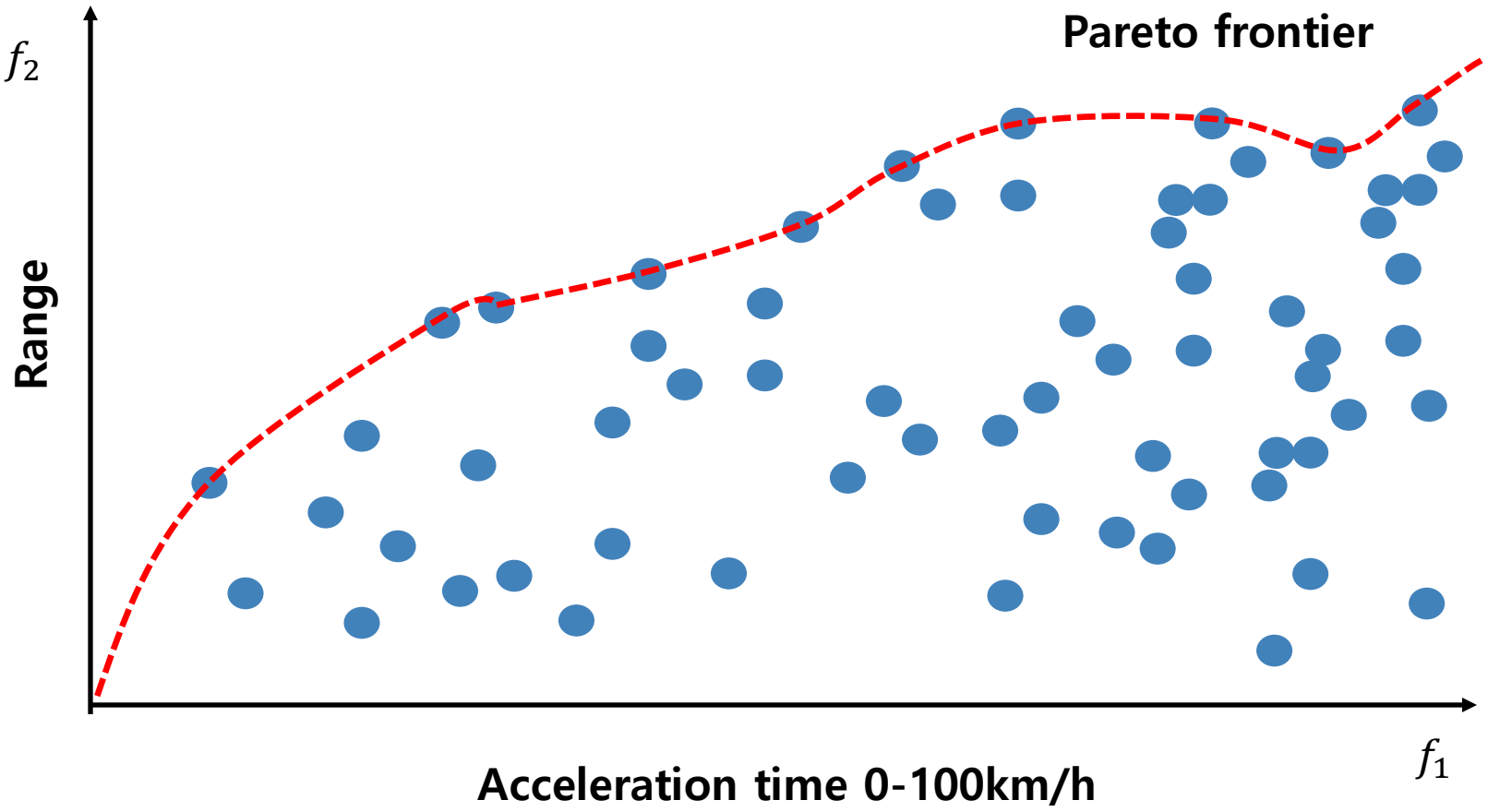


Power

Capacity

Multi objective evolutionary process

Multi object example



Kalyanmoy, D., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on evolutionary computation*, *6*(2), 182-197.

Proposed methodology

Step2 optimal sub-ensemble selection

- 최적의 sub-ensemble을 찾기 위해 구해진 pareto-optimal set 중 예측 성능이 높은 sub-ensemble을 선택

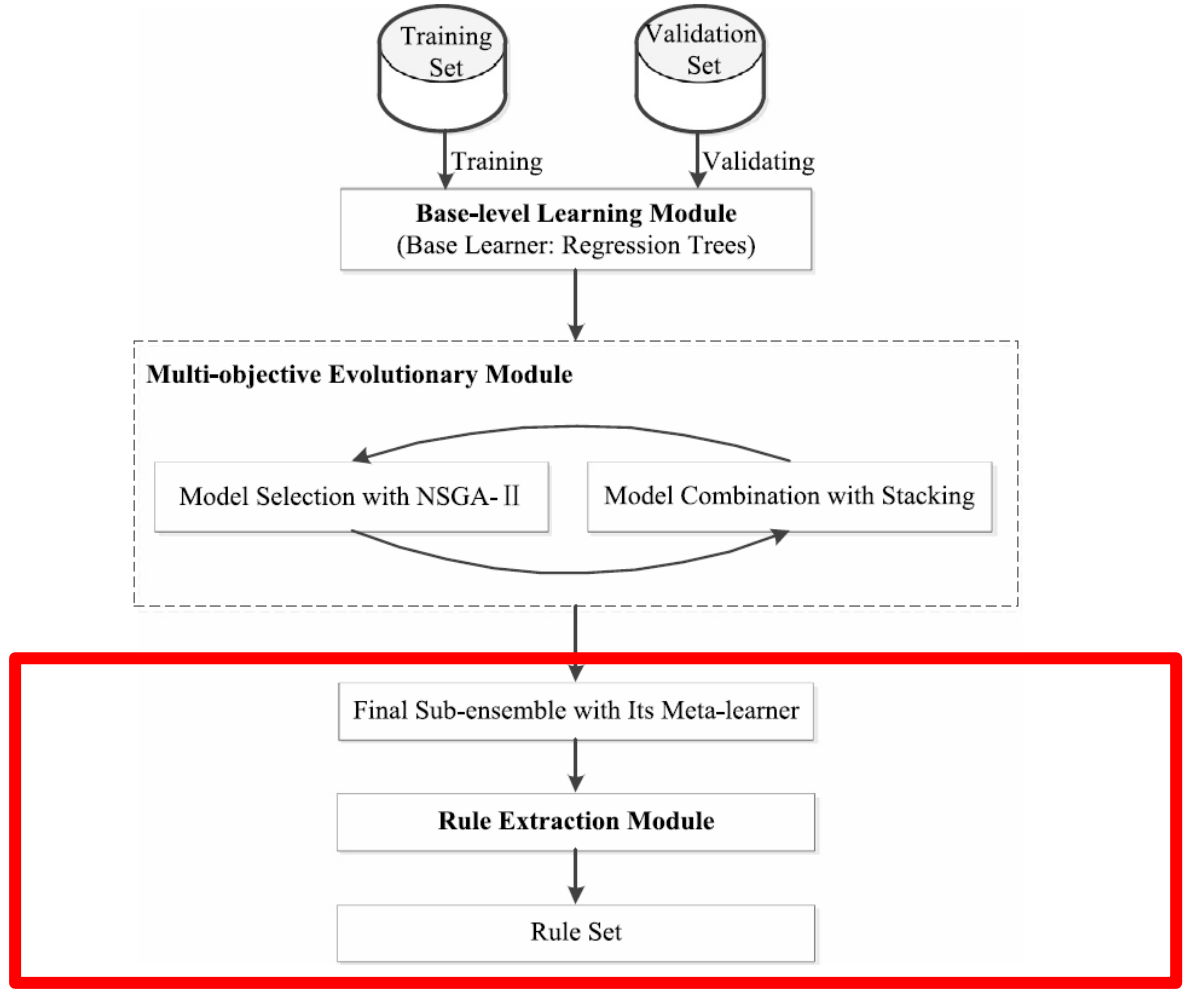
Pareto-optimal set

Sub-ensemble G1	Sub-ensemble G2	Sub-ensemble G3	…	Sub-ensemble GP-2	Sub-ensemble GP-1	Sub-ensemble GP

Sub-ensemble accuracy

0.84	0.83	0.82	…	0.89	0.81	0.82

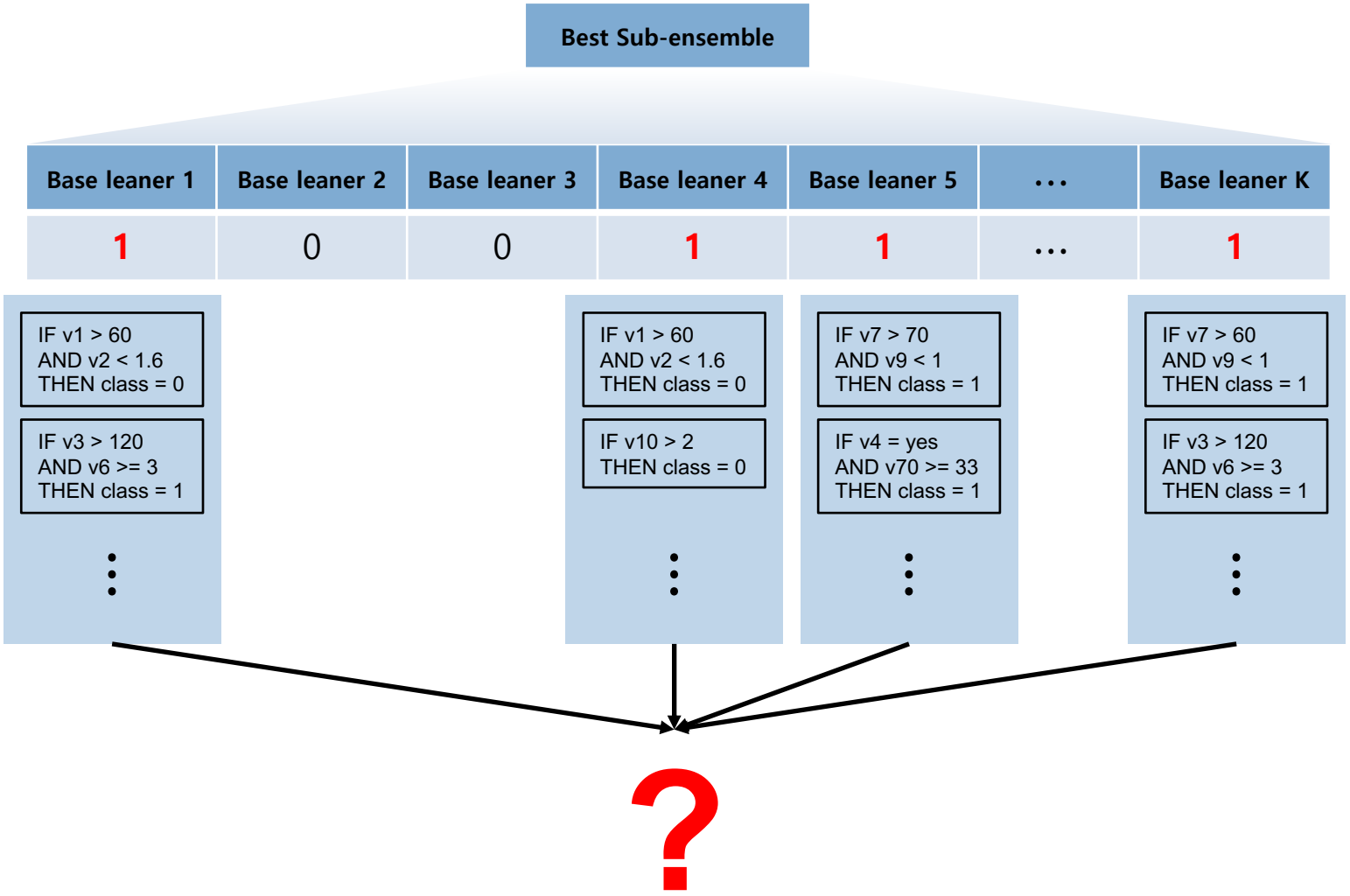
Proposed methodology



Step3

Proposed methodology

Step3 Rule extraction



Proposed methodology

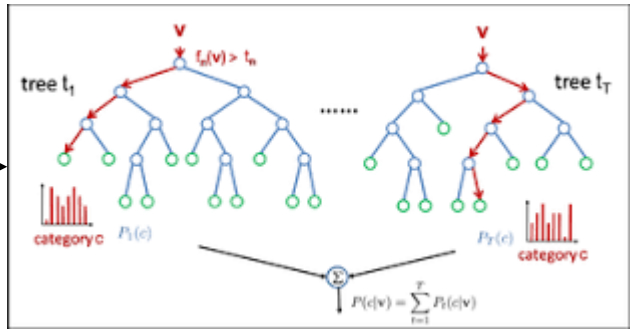
Step3 Rule extraction process

- Best sub-ensemble에 대해 meta learner로 Random forest를 사용
- RF를 통해 각 base leaner에 대해 Importance score를 얻을 수 있음

[Meta-data of Training set]

Base leaner 1		⋯	Base leaner k	
$p_1(y_0\|x_1)$	$p_1(y_1\|x_1)$	…	$p_k(y_0\|x_1)$	$p_k(y_1\|x_1)$
$p_1(y_0\|x_2)$	$p_1(y_1\|x_2)$		$p_k(y_0\|x_2)$	$p_k(y_1\|x_2)$
⋮		⋮	⋮	
$p_1(y_0\|x_{n-1})$	$p_1(y_1\|x_{n-1})$		$p_k(y_0\|x_{n-1})$	$p_k(y_1\|x_{n-1})$
$p_1(y_0\|x_n)$	$p_1(y_1\|x_n)$	…	$p_k(y_0\|x_n)$	$p_k(y_1\|x_n)$

Training RF



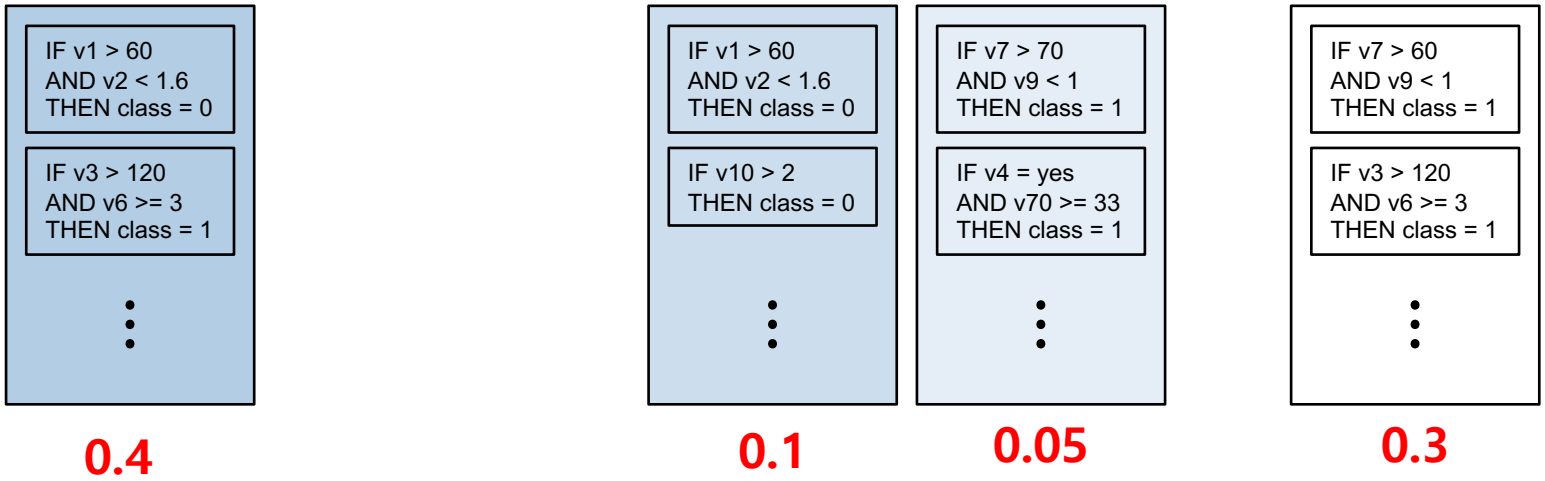
[Meta-data of Training set]

Proposed methodology

Step3 Rule extraction process

- Importance score를 통해 각 base leaner의 규칙 셋에 가중치를 부여

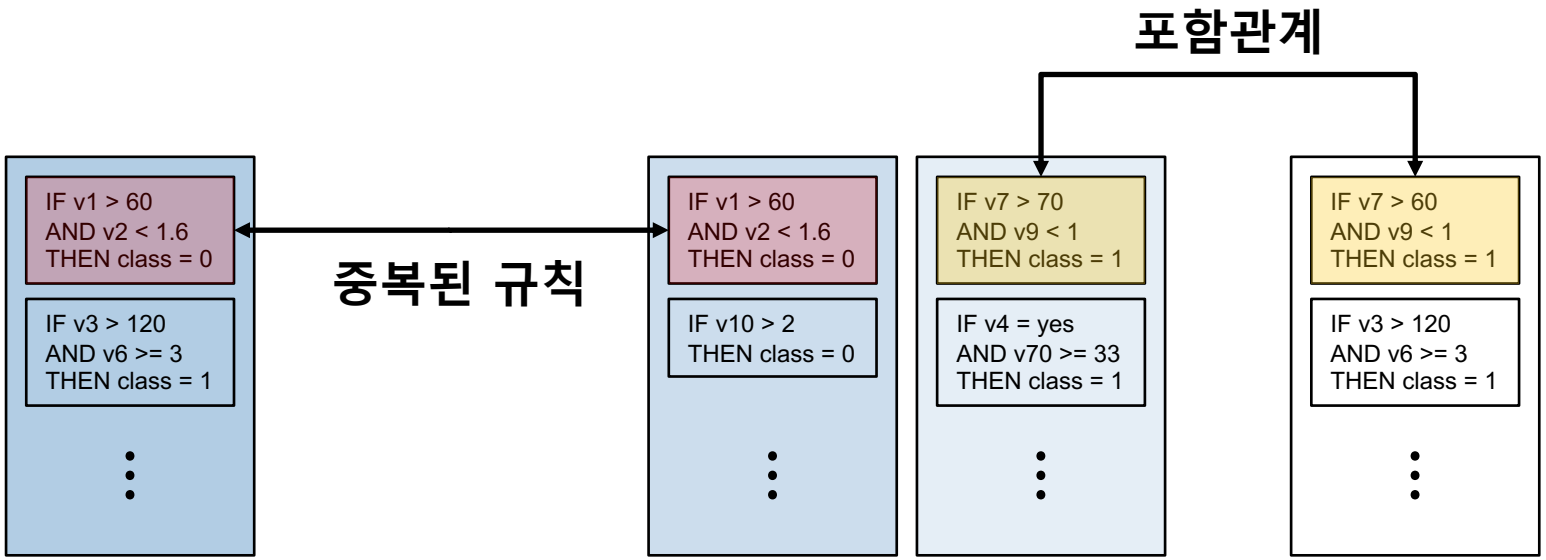
Base leaner 1	Base leaner 2	Base leaner 3	Base leaner 4	Base leaner 5	…	Base leaner K
1	0	0	1	1	…	1
0.4	-	-	0.1	0.05		0.3



Proposed methodology

Step3 Rule extraction process

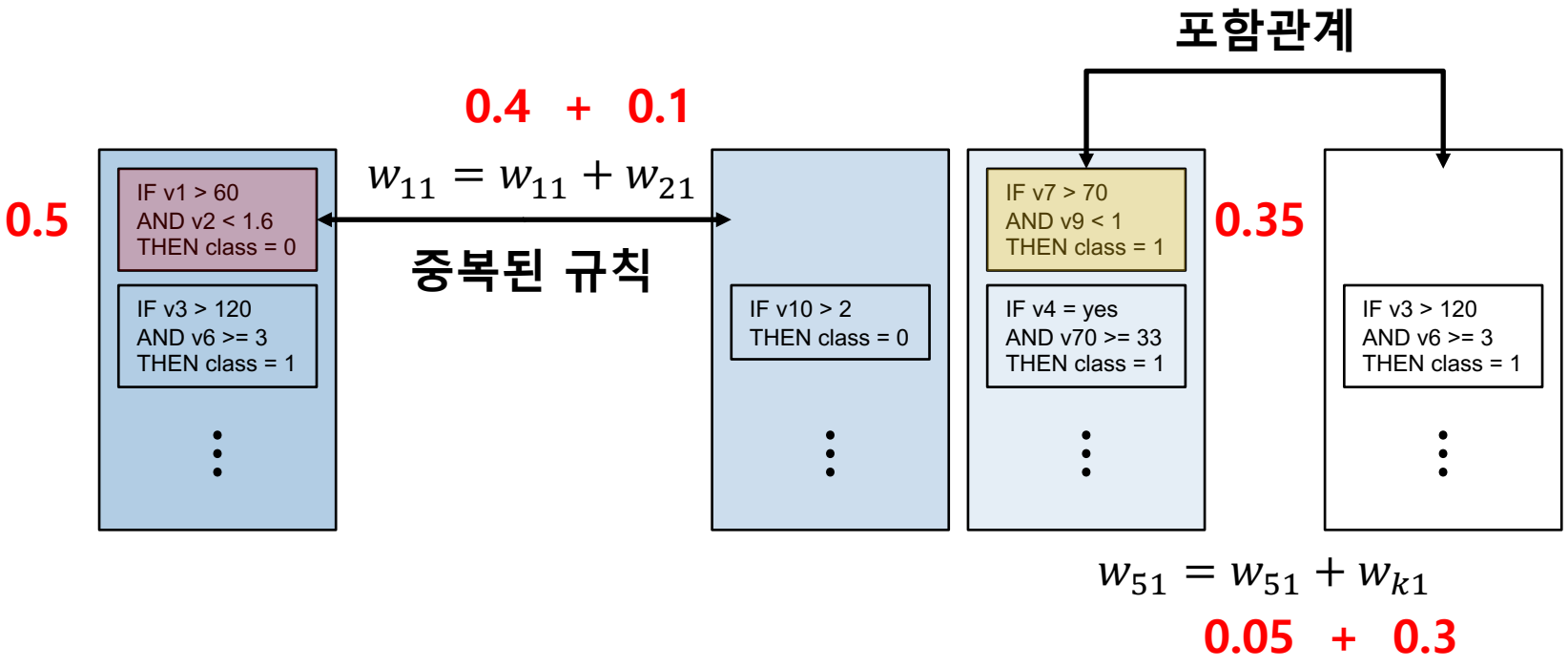
- 규칙 중 중복되거나, 혹은 규칙이 포함관계 있는 규칙을 제거
- 이 과정에서 규칙의 부여된 가중치를 더하여 규칙의 중요도를 업데이트



Proposed methodology

Step3 Rule extraction process

- 규칙 중 중복되거나, 혹은 규칙이 포함관계 있는 규칙을 제거
- 이 과정에서 규칙의 부여된 가중치를 더하여 규칙의 중요도를 업데이트



Proposed methodology

Step3 Rule extraction process

- 계산된 가중치에 따라 전체 규칙에 Rank를 부여함
- 각 규칙에 대해 Coverage와 complexity를 정의하고 산출

Index	Rules	weights	Ranks
R_{11}	IF v1 > 60 AND v2 < 1.6 THAN class= 1	0.5	1
R_{12}	IF v3 > 120 AND v6 >= 3 THEN class = 1	0.4	2
R_{21}	…	…	3
R_{23}	…	…	4
⋮	…	…	⋮
R_{k1}	…	…	n-1
R_{k5}	IF v3 > 120 AND v6 >= 3 THEN class = 1	0.24	n

$$coa_{ki} = \frac{sampleNum\_a_{ki}}{sampleNum_a}$$

Training set 중 해당하는 관측치 비율

$$cot_{ki} = \frac{sampleNum\_t_{ki}}{sampleNum_t}$$

validation set 중 해당하는 관측치 비율

$$complexity_{ki} = \frac{featureNum_{ki}}{featureNum_a}$$

전체 변수 중 규칙에 포함되는 변수 개수

Proposed methodology

Step3 Rule extraction process

- 산출된 coverage와 complexity를 정규화
- 정규화 된 값을 통해 각 규칙의 score를 산출함

Index	Rules	weights	$*\ coa_{ki}$	$*\ cot_{ki}$	$*\ complexity_{ki}$	$Score_{ki}$
R_{11}	IF v1 > 60 AND v2 < 1.6 THAN class= 1	0.5	**0.8**	**0.61**	**0.2**	**0.805**
R_{12}	IF v3 > 120 AND v6 >= 3 THEN class = 1	0.4	**0.6**	**0.2**	**0.1**	**0.68**
R_{23}	…	…	…	…	…	…
⋮	…	…	…	…	…	…
R_{k5}	IF v3 > 120 AND v6 >= 3 THEN class = 1	0.24	**0.9**	**0.9**	**0.1**	**0.648**

$$score_{ki} = w_{ki}(coa_{ki} + cot_{ki} + (1 - complexity_{ki}))$$

*** Minmax scaling**

Proposed methodology

Step3 Rule extraction process

- 산출된 score를 기반으로 재정렬 이후 각 규칙에 대해 평균정확도 산출
- 평균 정확도가 기준치 이상*인 규칙을 최종적으로 추출함

Index	Rules	weights	$*\ coa_{ki}$	$*\ cot_{ki}$	$*\ complexity_{ki}$	$Score_{ki}$	$avgAcc_{ki}$
R_{11}	IF v1 > 60 AND v2 < 1.6 THAN class= 1	0.5	**0.8**	**0.61**	**0.2**	**0.805**	**0.76**
R_{12}	IF v3 > 120 AND v6 >= 3 THEN class = 1	0.4	**0.6**	**0.2**	**0.1**	**0.68**	**0.7**
R_{k5}	IF v3 > 120 AND v6 >= 3 THEN class = 1	0.24	**0.9**	**0.9**	**0.1**	**0.648**	**0.8**
⋮	…	…	…	…	…	…	…
R_{23}	…	…	…	…	…	…	…

$$avgAccuracy_{ki} = \frac{1}{2}\left(\frac{correct\_a_{ki}}{sampleNum\_a_{ki}} + \frac{correct\_t_{ki}}{sampleNum\_t_{ki}}\right)$$

Training set / validation set 에 속하는 관측치에 대해
규칙$_{ki}$를 만족하는 관측치 중 정확하게 분류된 관측치 개수

*** User parameter**

목차

Results

Data

Prostate cancer dataset
(http://clinic ncmi.cn)

2007 ~ 2013년 환자 기록
관측치 수 : 1657개

Table 1
List of variables used in the experiments.

Variable category	Variable ID	Variable	Variable description
Target variable	1	Label	"0"—not prostate cancer, "1"—prostate cancer
Demographic information	2	Age	Patient age
Physical Examination	3	Volume	Prostate volume
	4	pdia	The anteroposterior diameter of the prostate
Blood routine examination	5	B1	White blood cell count
	6	B2	Monocytes
	7	B3	Haematocrit (HCT) determination
	8	B4	Red blood cell count
	9	B5	Red blood cell volume distribution width (RDW)
	10	B6	Lymphocytes
	11	B7	Mean corpuscular volume (MCV)
	12	B8	Mean corpuscular hemoglobin (MCH)
	13	B9	Mean corpuscular hemoglobin concentration (MCHC)
	14	B10	Basophile granulocytes
	15	B11	Eosinophil count
	16	B12	Eosinophilic granulocytes
	17	B13	Hemoglobin determination
	18	B14	Platelet count
	19	B15	neutrophile granulocytes
Prostate Specific Antigen (PSA) examination	20	tpsa	Total prostate specific antigen concentration
	21	fpsa	Free prostate specific antigen concentration
	22	fpsa/tpsa	The ratio of fpsa to tpsa
Urine routine examination	23	U1	Location of the urine 70% red cell's forward scattered light
	24	U2	Urine white blood cell
	25	U3	Urine white cell examination
	26	U4	Urine specific gravity determination
	27	U5	Qualitative test of bilirubin in urine
	28	U6	Qualitative test of urobilinogen
	29	U7	Qualitative test of urine protein
	30	U8	Urine red blood cell
	31	U9	Urine red cell examination
	32	U10	Urine erythrocyte forward scattered light distribution width
	33	U11	Urine yeast cells
	34	U12	Urinary epithelial cells
	35	U13	Microscopic examination of urine epithelial cells
	36	U14	Qualitative test of urine glucose
	37	U15	Urine acetone body test
	38	U16	Urine conductivity
	39	U17	Urine casts
	40	U18	Urine crystallization
	41	U19	Urine pH test
	42	U20	Urine nitrite test
	43	U21	Urine color

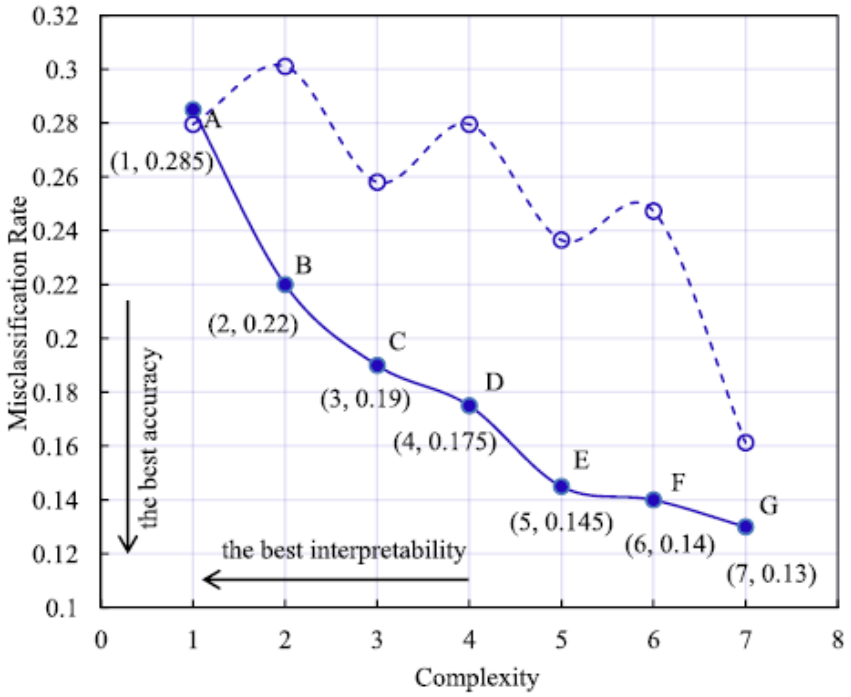
Results

Subspace size / pareto-optimal solution set

Table 2
Results under different subspace size.

Subspace size	Ensemble size	Test error
3	7.0000	0.2624
6	6.6000	0.2086
9	6.6000	0.1978
12	7.1000	0.2043
15	7.1538	0.1927
18	**9.5200**	**0.1772**
21	10.6667	0.1783
24	8.0000	0.1849
27	10.9000	0.1839
30	8.6000	0.1839
33	7.0667	0.1778
36	8.3636	0.1779
39	7.5000	0.1788
42	7.5000	0.1785

[the results different subspace sizes]



[Misclassification rate vs complexity
of the pareto-optimal solutions from one run]

Results

Validation and test results

Table 3
Validation and test results of Pareto-optimal solutions from one run.

Solution	Ensemble complexity	Validation result			Test result		
		Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
A	1	0.7150	0.5571	0.8000	0.7204	0.3929	0.8615
B	2	0.7800	0.5143	0.9077	0.6989	0.3214	0.8615
C	3	0.8100	0.6571	0.8615	0.7419	0.6071	0.8000
D	4	0.8250	0.6857	0.8692	0.7204	0.6071	0.7692
E	5	0.8550	0.7571	0.8846	0.7634	0.5714	0.8462
F	6	0.8600	0.7143	0.8923	0.7527	0.6429	0.8000
G	7	**0.8700**	**0.7857**	**0.9077**	**0.8387**	**0.7857**	**0.8615**

Results

Test results

Table 4
Test results of the proposed method from 25 runs.

Run number	Ensemble complexity	Testing results		
		Accuracy	Sensitivity	Specificity
1	7	0.8387	0.7857	0.8615
2	11	0.8172	0.7143	0.8615
3	8	0.8602	0.8214	0.8769
4	10	0.8387	0.7143	0.8923
5	9	0.7957	0.7143	0.8308
6	7	0.8172	0.7500	0.8462
7	7	0.8065	0.7500	0.8308
8	7	0.8065	0.6429	0.8769
9	10	0.8172	0.7500	0.8462
10	7	0.8602	0.8571	0.8615
11	9	0.8495	0.7500	0.8923
12	8	0.8065	0.6786	0.8615
13	18	0.8280	0.7500	0.8615
14	11	0.8172	0.7500	0.8462
15	12	0.8387	0.7143	0.8923
16	9	0.8280	0.7500	0.8615
17	12	0.7849	0.8214	0.7692
18	8	0.8065	0.6071	0.8923
19	19	0.8172	0.6786	0.8769
20	9	0.8280	0.7857	0.8462
21	10	0.8280	0.7143	0.8769
22	6	0.8065	0.7500	0.8308
23	5	0.8065	0.6786	0.8615
24	9	0.8280	0.7500	0.8615
25	10	0.8387	0.7857	0.8615
Min	5	0.7849	0.6071	0.7692
Median	9	0.8172	0.75	0.8615
Max	19	**0.8602**	**0.8571**	**0.8923**
Ave	9.52	**0.8228**	**0.7386**	**0.8591**

Results

Comparison with other machine learning methods

Table 5
Classification results of different methods.

Method	TP%	FP%	TN%	FN%	Accuracy	Sensitivity	Specificity
DT	24.07%	10.62%	54.15%	11.17%	0.7878	0.6938	0.8290
Linear-SVM	17.02%	17.66%	42.82%	22.50%	0.6044	0.4908	0.6556
RBF-SVM	0.00%	34.69%	**65.31%**	**0.00%**	0.6989	0.0000	1.0000
RBF	12.39%	22.30%	45.42%	19.90%	0.6954	0.3571	0.6954
Proposed method	**25.62%**	**9.07%**	56.11%	9.20%	**0.8228**	**0.7386**	**0.8591**

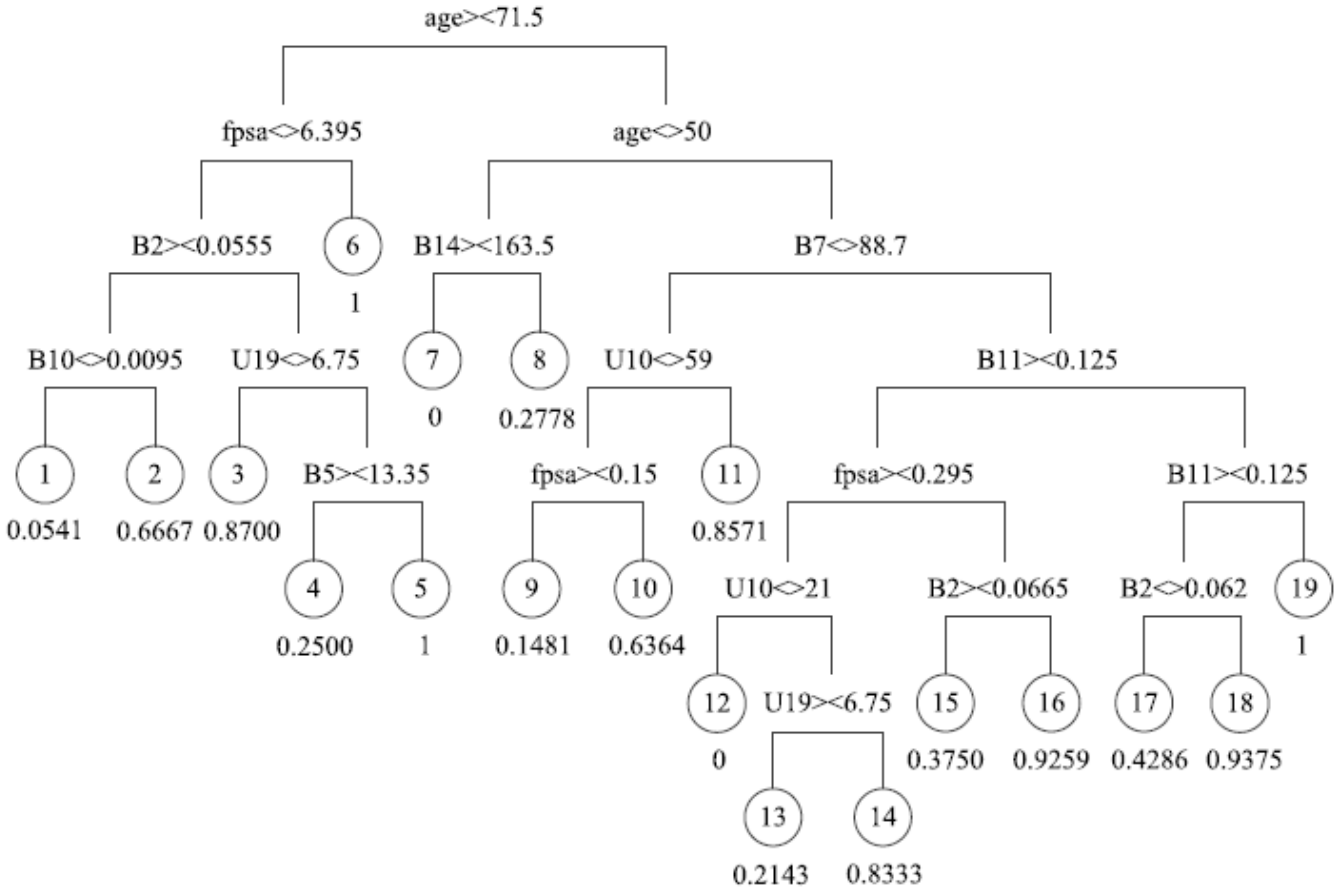
Table 6
Classification results of several ensemble methods under same complexity.

Method	Mean ± STD.			*p*-value (Method VS. Proposed method)		
	Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
Proposed method	**0.8228** ± 0.0189	**0.7386** ± 0.0562	**0.8591** ± 0.0265	–	–	–
AdaBoost	0.7947 ± 0.0138	0.7208 ± 0.0357	0.8266 ± 0.02448	**0.0300**	0.0550	**0.0390**
Gradient boosting	0.7957 ± 0.0113	0.6786 ± 0.0157	0.8462 ± 0.0111	**0.0000**	**0.0000**	**0.0070**
Random forest	0.8065 ± 0.0235	0.7143 ± 0.0564	0.8462 ± 0.0326	**0.0000**	0.0620	**0.0000**

Results

Extracted rules

- 최종 best sub-ensemble의 개수는 11개의 base learner를 갖음
- 최종 규칙의 수는 190개 → 제안하는 규칙 추출 방법론을 적용



Results

Extracted rules

- 추출된 19개의 규칙 중 평균 정확도가 90% 이상인 규칙 13개를 추출

Table 8
Interpretable rules after ranking with accuracy higher than 0.9000.

Rule ID	Tree ID	Weight	Rule
8	9	0.1432	**IF** age< 47.5 AND pdia< 2.35 AND B6>= 0.3225 **THEN** label = 0
36	1	0.0918	**IF** age< 50 AND B14>= 163.5 **THEN** label = 0
10	9	0.1432	**IF** age< 47.5 AND pdia< 2.35 AND B3< 0.3375 AND B6 < 0.3225 **THEN** label = 0
17	6	0.0787	**IF** age< 77.5 AND age>= 50 AND volume>= 50.62 AND B7>= 77 AND B12< 0.037 **THEN** label = 0
5	10	0.1147	**IF** pdia>= 1.95 AND B3< 0.299 AND B15>= 0.68 **THEN** label = 0
11	9	0.1432	**IF** age>= 47.5 AND age< 71 AND pdia< 2.35 AND fpsa/tpsa>= 0.2419 AND B13>= 148.5 **THEN** label = 0
6	10	0.1147	**IF** pdia>= 1.95 AND B3< 0.299 AND B5< 14.35 AND B15< 0.68 **THEN** label = 1
33	2	0.0551	**IF** pdia< 2.15 AND fpsa>= 1.215 AND B4>= 3.44 **THEN** label = 0
1	11	0.0768	**IF** age< 76.5 AND age>= 47.5 AND tpsa>= 1.78 AND U10< 27 AND B13>= 138.5 **THEN** label = 0
18	6	0.0787	**IF** age< 77.5 AND age>= 50 AND volume>= 32.28 AND B7< 77 **THEN** label = 1
34	2	0.0551	**IF** pdia< 2.15 AND fpsa< 1.215 AND fpsa>= 0.095 AND B4< 4.705 AND B6>= 0.1395 AND B13< 103.5 **THEN** label = 0
29	3	0.0512	**IF** age>= 82 AND volume>= 12.01 AND U10< 27 AND U14< 650 AND B7< 90.3 **THEN** label = 0
35	2	0.0551	**IF** pdia< 2.15 AND fpsa< 1.215 AND fpsa>= 0.085 AND B4< 4.705 AND B6< 0.413 AND B6>= 0.2455 AND B7>= 88.95 AND B10>= 0.0065 AND B13>= 103.5 **THEN** label = 1

Table 9
The performance of the extracted rules.

Rule ID	Performance on all data		Performance on test data		Comprehensive indicators		
	Coverage	Accuracy	Coverage	Accuracy	Complexity	Average accuracy	Score
8	0.1845	1.0000	0.1068	1.0000	0.9412	**1.0000**	0.2643
36	0.3301	0.9118	0.1748	1.0000	0.9706	0.9559	0.2490
10	0.0777	1.0000	0.0485	1.0000	0.9118	**1.0000**	0.1611
17	0.4563	0.9574	0.0680	1.0000	0.8824	0.9787	0.1543
5	0.1748	0.9444	0.0194	1.0000	0.9412	0.9722	0.1519
11	0.1553	0.8750	0.0291	1.0000	0.8824	0.9375	0.1502
6	0.0485	1.0000	0.0194	1.0000	0.9118	**1.0000**	0.1023
33	0.2233	0.8696	0.0583	1.0000	0.9412	0.9348	0.0913
1	0.1845	1.0000	0.0388	1.0000	0.8824	**1.0000**	0.0899
18	0.0194	1.0000	0.0097	1.0000	0.9118	**1.0000**	0.0606
34	0.1553	0.8125	0.0388	1.0000	0.8529	0.9063	0.0530
29	0.1068	1.0000	0.0097	1.0000	0.8824	**1.0000**	0.0423
35	0.1165	0.9167	0.0291	1.0000	0.7647	0.9583	0.0214

목차

Conclusion

- Ensemble learning 중 stacking 기법을 활용하여 예측 성능을 향상
- 최적의 Sub-ensemble을 선택하기 위해 Multi-objective evolutionary process를 적용
- Decision tree 기반으로 생성된 규칙을 종합하고 해석력을 증대 시키기 위해 규칙 추출방법론을 제안함

감사합니다