인공지능을 위한 머신러닝 알고리즘

13. Weka를 이용한 머신러닝 실습

Tacademy

CONTENTS

학습 목표

- Weka를 사용하여 데이터를 전처리하고 분석할 수 있다.
- Weka로 분류를 하기 위해 알고리즘의 파라미터를 설정할 수 있다.
- Weka로부터 얻은 분류 결과를 해석할 수 있다.

1. Weka 소개

Weka란?

- **Weka(Waikato Environment for Knowledge Analysis)**
  - 뉴질랜드의 **Waikato** 대학교 컴퓨터공학부에서 제작
  - **Weka**는 뉴질랜드에서만 발견되는 새이기도 함

- 대표적인 기계학습 알고리즘 모음, 데이터 마이닝 도구

Weka의 특징

❖ Weka의 주요 기능

- 데이터 전처리, 특징값 선별(Feature Selection)
- 군집화, 데이터 가시화
- 분류, 회귀 분석, 시계열 예측

❖ 소프트웨어 특성

- 무료 및 소스 공개 소프트웨어 (free & open source GNU General Public License)
- Java로 구현, 다양한 플랫폼에서 실행 가능

Weka를 구성하는 인터페이스

Explorer : 다양한 분석 작업을 한 단계씩 분석 수행 및 결과 확인 가능,
일반적으로 가장 먼저 실행

Experimenter: 분류 및 회귀 분석을 일괄 처리. 결과 비교 분석
- 다양한 알고리즘 및 파라미터 설정
- 여러 데이터 알고리즘 조합 동시 분석
- 분석 모델 간 통계적 비교
- 대규모 통계적 실험 수행

KnowledgeFlow: 데이터 처리 과정의 주요 모듈을 그래프로 가시화하여 구성

Simple CLI: 다른 인터페이스를 컨트롤하는 스크립트 입력창
Weka의 모든 기능을 명령어로 수행 가능

2. Weka로 붓꽃(iris) 분류하기

기계학습 – 패턴 분류 절차

❖ 피처 정의 (features or attributes)

- **sepal length, sepal width, petal length, petal width**
- 클래스 (class) label: 붓꽃의 세 아종을 예측 목표 변수로 설정

  setosa, versicolor, or virginica

❖ 샘플 수집 및 데이터셋 구성

- 붓꽃의 각 아종 별로 50개체의 피처를 측정
- 데이터셋: 2차원 표 형태: **150 samples (or instances) * 5 attributes**

기계학습 – 패턴 분류 절차

❖ 패턴 분류 수행

- 기계학습 알고리즘을 활용
- 예> 결정 트리, 랜덤 포레스트, **SVM**, 다층 퍼셉트론

❖ 패턴 분류 성능 평가

- 패턴 분류 모델의 상대적 비교 과정 (알고리즘 + 파라미터 설정)
- 다양한 평가 기준을 적용: 분류 정확도, **precision** + **recall** 등

피처 정의 – 붓꽃(iris)

Iris setosa

Iris versicolor

Iris virginica

분류에 사용할 꽃의 특징

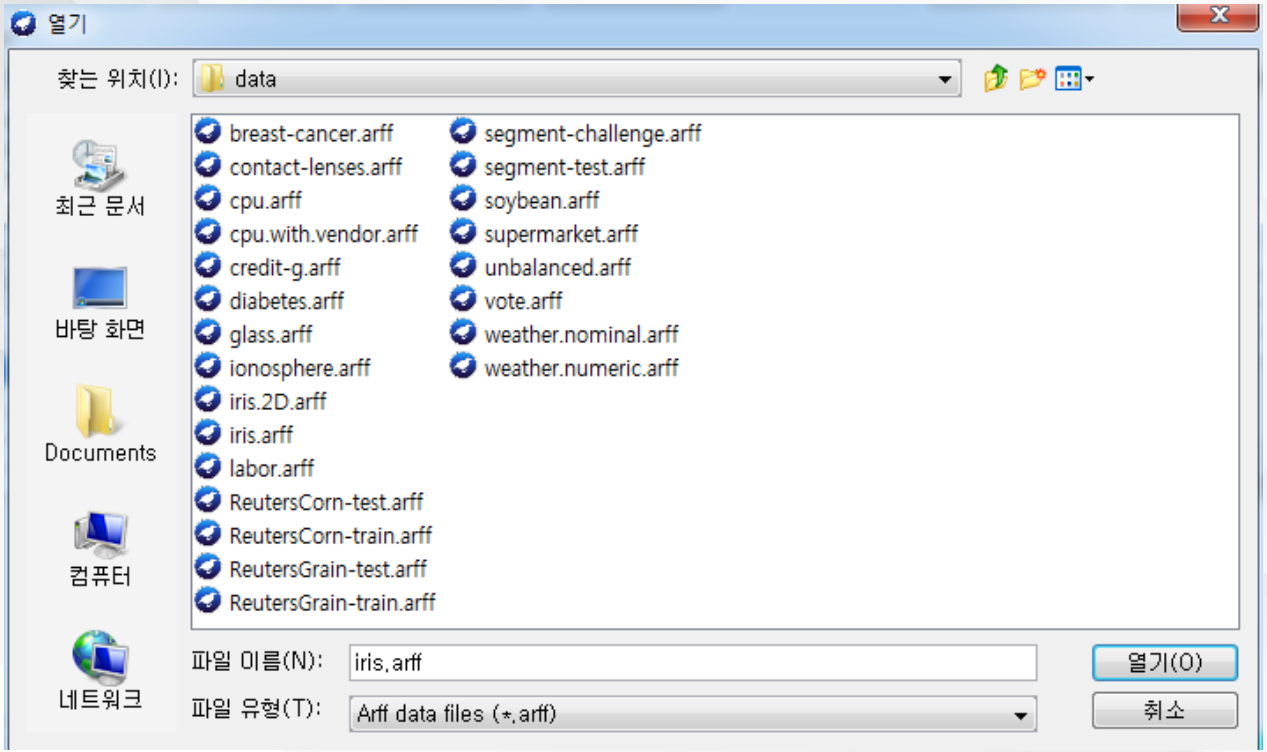
꽃받침 **(Sepal)**	길이 **(Length)**
꽃잎 **(Petal)**	너비 **(Width)**

특징 기준 판별

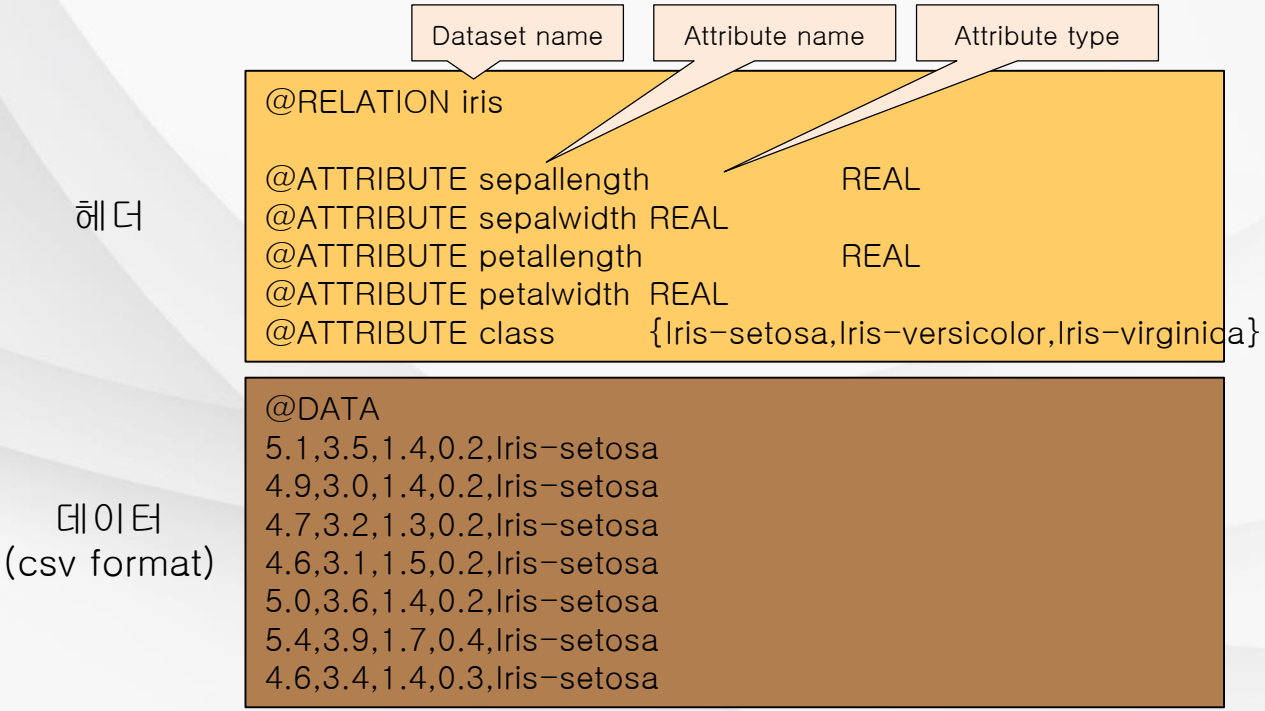
2.5, 9.4, 1.0, 0.5, Iris-setosa
1.2, 9.2, 1.2, 0.2, Iris-setosa
1.9, 9.0, 1.4, 0.4, Iris-setosa
...
7.2, 1.2, 3.4, 9.1, Iris-versicolor

데이터셋 – 파일 열기

Weka 폴더 → **'data'** 폴더에서 **'iris.arff'** 파일 선택



데이터셋 - Weka의 데이터 형식 (.arff)



Excel을 이용하여 csv 파일 생성 후, 헤더만 추가하면 쉽게 arff 포맷의 파일 생성 가능

데이터셋 구성 - 데이터 전처리

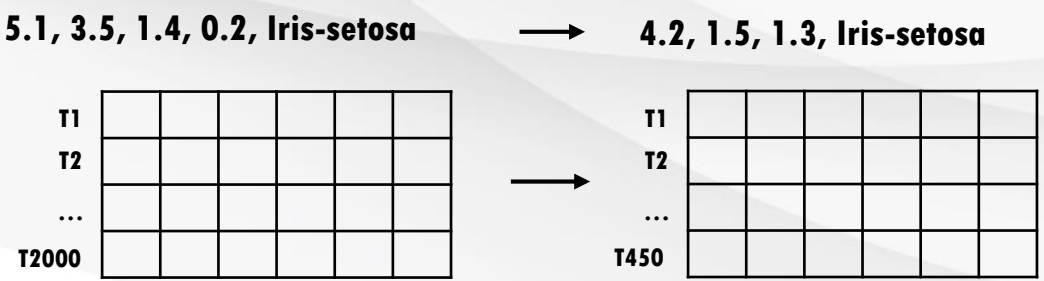
- 예측 모델 학습 및 평가를 위해 준비하는 데이터 집합을 모델에 입력하기 전에 다양한 처리를 하여 데이터의 품질을 향상시키는 과정

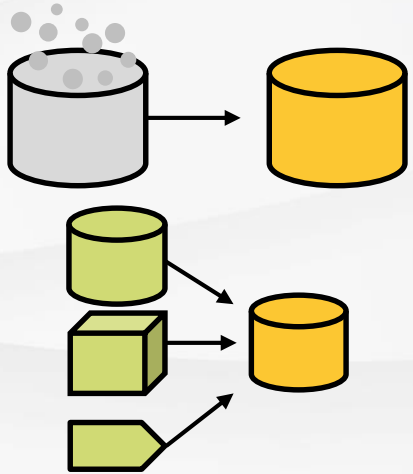
Data cleaning

Data integration

Data transformation

Data reduction





데이터셋 구성 – 전처리(1)

❖ 적용할 **filter**를 선택

❖ 특징 **(attribute)**

◉ 데이터 차원 축소

weka.filters.supervised.attribute.AttributeSelection

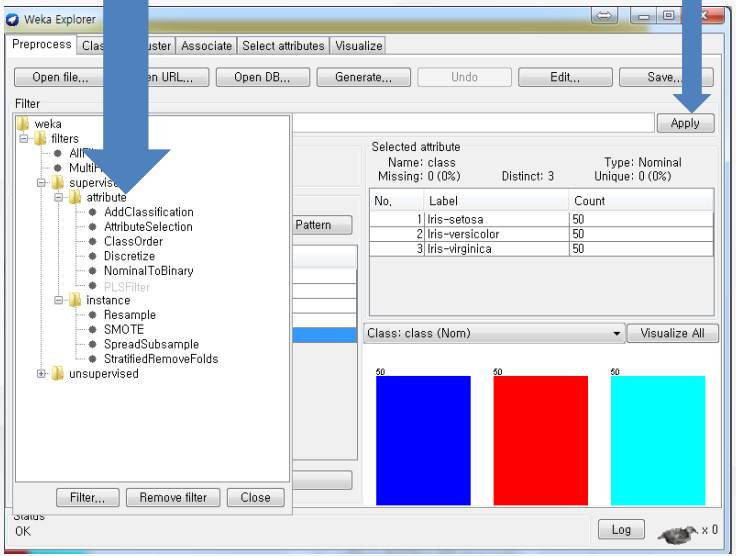
◉ 데이터 변형 및 데이터 이산화

weka.filters.supervised.attribute.Discretize

weka.filters.unsupervised.attribute.Normalize, Standardize

적용 가능한
데이터 전처리 기법

선택 후 **apply** 버튼 클릭



데이터셋 구성 – 전처리(1)

❖ 데이터 인스턴스

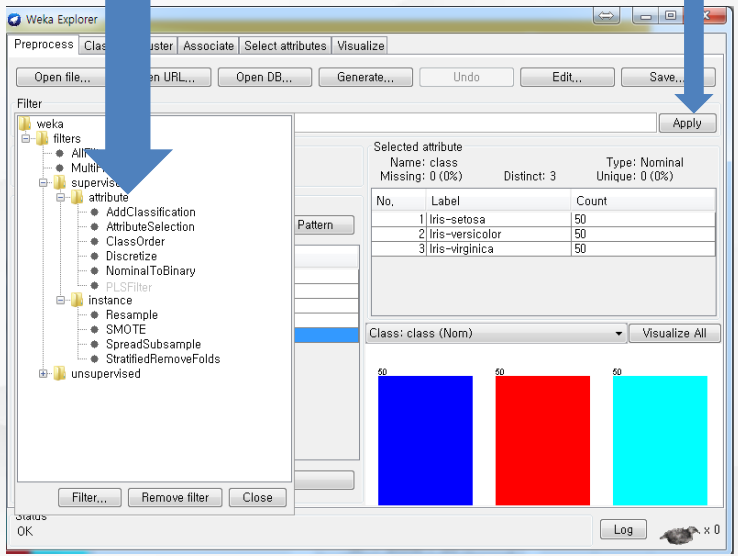
- 데이터 개수 증대

weka.filters.supervised.instance.Resample

weka.filters.supervised.instance.SMOTE

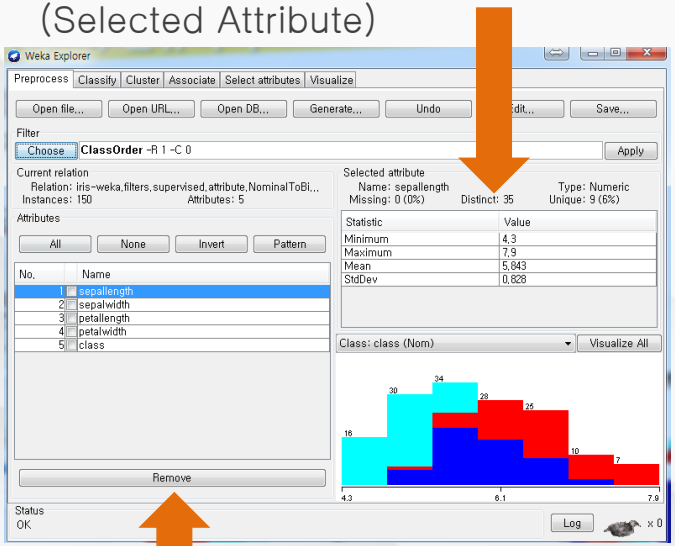
적용 가능한
데이터 전처리 기법

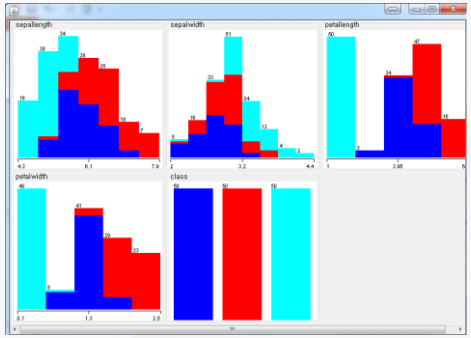
선택 후 apply 버튼 클릭



데이터셋 구성 – 전처리(2)

특징값 별 기초적 통계 분석
(Selected Attribute)





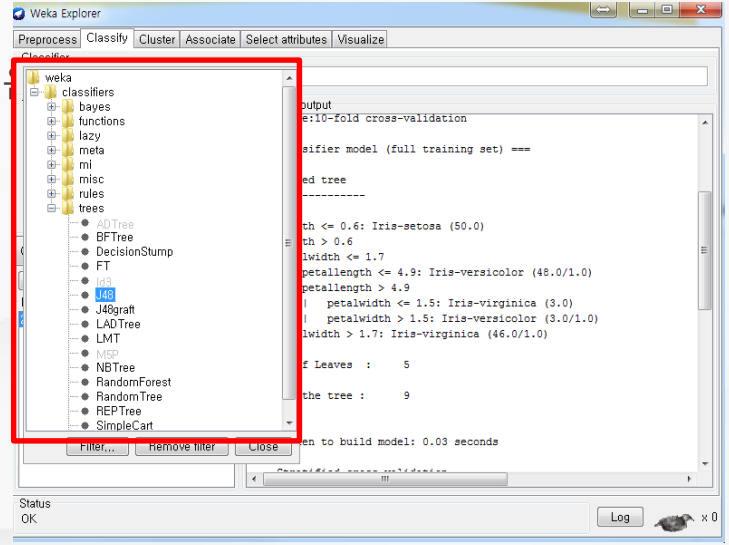
모든 특징값을 대상으로
클래스 레이블 분포 가시화

특징값 삭제 (Remove Attributes)

패턴 분류 수행 - 알고리즘 선택

❖ J48 (c4.5의 Java 구현 버전)

- 학습 결과 모델에서 분류 규칙을 '트리'형태로 얻을 수 있음
- **Weka**에서 찾아가기: **classifiers-trees-J48**



❖ 랜덤 포레스트

- 결정 트리의 앙상블 모델
- 특징들을 무작위로 선택하여 결정 트리들이 생성됨
- 분류 과정에서 각 트리는 투표를 하며 가장 많은 표를 얻은 클래스가 선택됨
- **Weka**에서 찾아가기: **classifiers-trees-RandomForest**

❖ 다층 퍼셉트론

- 실용적으로 매우 폭 넓게 쓰이는 대표적 분류 알고리즘
- **Weka**에서 찾아가기: **classifiers-functions-MultilayerPerceptron**

분류 알고리즘의 파라미터 설정

❖ 파라미터 설정 = 자동차 튜닝

- 많은 경험 또는 시행착오 필요
- 파라미터 설정에 따라 동일한 알고리즘에서도 최악에서 최고의 성능을 모두 보일 수도 있음

❖ 결정트리의 주요 파라미터 (J48, SimpleCart in Weka)

- 트리의 크기에 직접적 영향을 주는 파라미터: **confidenceFactor, pruning, minNumObj** 등

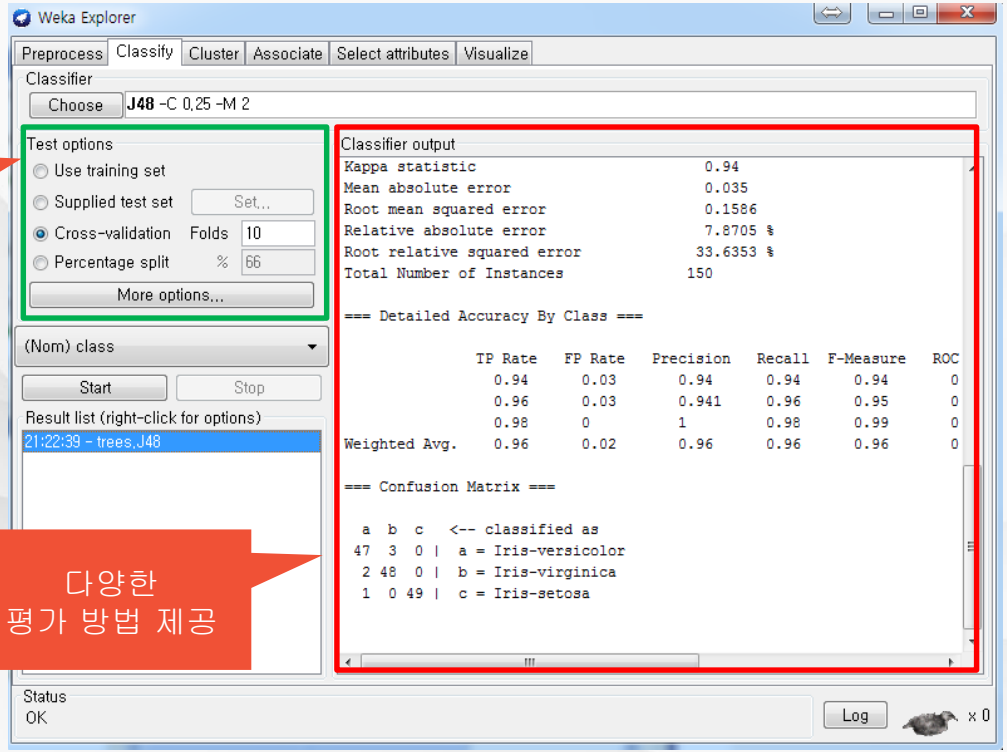
❖ Random Forest의 주요 파라미터 (RandomForest in Weka)

- **numTrees:** 학습 및 예측에 참여할 **tree**의 수를 지정. 대체로 많을 수록 좋으나, **overfitting**에 주의해야 함

❖ 참고: 신경망의 주요 파라미터 (MultilayerPerceptron in Weka)

- 구조 관련: **hiddenLayers,**
- 학습 과정 관련: **learningRate, momentum, trainingTime (epoch), seed**

패턴 분류 성능 평가


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
J48 -C 0.25 -M 2
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) class
Start
Stop
Result list (right-click for options)
21:22:39 - trees.J48
Classifier output
Kappa statistic 0.94
Mean absolute error 0.035
Root mean squared error 0.1586
Relative absolute error 7.8705 %
Root relative squared error 33.6353 %
Total Number of Instances 150
=== Detailed Accuracy By Class ===
TP Rate FP Rate Precision Recall F-Measure ROC
0.94 0.03 0.94 0.94 0.94 0
0.96 0.03 0.941 0.96 0.95 0
0.98 0 1 0.98 0.99 0
Weighted Avg. 0.96 0.02 0.96 0.96 0.96 0
=== Confusion Matrix ===
a b c <-- classified as
47 3 0 | a = Iris-versicolor
2 48 0 | b = Iris-virginica
1 0 49 | c = Iris-setosa
Status
OK
Log
x 0
평가를 위한 데이터 집합 세팅
다양한 평가 방법 제공

패턴 분류 성능 평가

실행 정보

```
=== Run information ===

Scheme:weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     iris-weka.filters.supervised.attribute.NominalToBinary-w
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
Test mode:10-fold cross-validation
```

분류 모델
(훈련 데이터 학습 모델)

```
=== Classifier model (full training set) ===

J48 pruned tree
------------------

petalwidth <= 0.6: Iris-setosa (50.0)
petalwidth > 0.6
|   petalwidth <= 1.7
|   |   petallength <= 4.9: Iris-versicolor (48.0/1.0)
|   |   petallength > 4.9
|   |   |   petalwidth <= 1.5: Iris-virginica (3.0)
|   |   |   petalwidth > 1.5: Iris-versicolor (3.0/1.0)
|   petalwidth > 1.7: Iris-virginica (46.0/1.0)

Number of Leaves  :     5

Size of the tree :      9
```

평가 결과
- 종합적 요약
- 클래스별 성능
- Confusion Matrix

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         144               96      %
Incorrectly Classified Instances         6                4      %
Kappa statistic                          0.94
Mean absolute error                      0.035
Root mean squared error                  0.1586
Relative absolute error                  7.8705 %
Root relative squared error             33.6353 %
Total Number of Instances              150

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.94      0.03       0.94      0.94       0.94      0.952    Iris-versicolor
                 0.96      0.03       0.941     0.96       0.95      0.961    Iris-virginica
                 0.98      0          1         0.98       0.99      0.99     Iris-setosa
Weighted Avg.    0.96      0.02       0.96      0.96       0.96      0.968

=== Confusion Matrix ===

  a  b  c   <-- classified as
 47  3  0 |  a = Iris-versicolor
  2 48  0 |  b = Iris-virginica
  1  0 49 |  c = Iris-setosa
```

결과는 모델에 따라 변할 수 있음

학습정리

지금까지 [Weka를 이용한 머신러닝 실습]에 대해서 살펴보았습니다.

Weka의 기능

데이터 전처리: 특징값 선별(feature selection) / 제거, 데이터 리샘플링
데이터 분류: 결정 트리, 랜덤 포레스트, 다층 퍼셉트론 등 다양한 알고리즘 선택
회귀 분석 및 시계열 예측

arff 파일

헤더: relation, attributes, data
데이터: csv format
Excel을 이용하여 csv 파일 생성 후, 헤더를 추가하여 arff 파일 생성

머신러닝 실습과 해석

테스트 옵션: cross validation
결과 해석: confusion matrix, classification accuracy, precision / recall