Part IV : 분석/시각화

## 4.1 분석/서비스 개요





● 주요 분석 기법

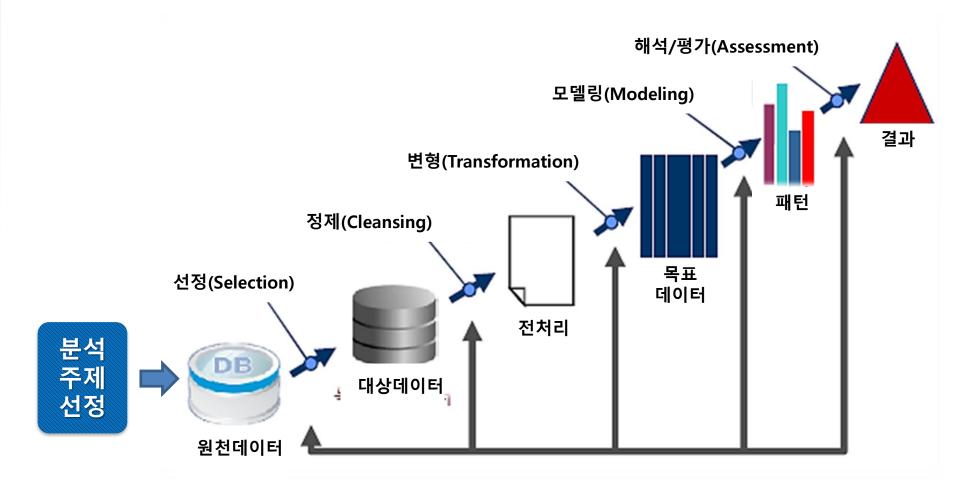




용어	설명
텍스트 마이닝 (Text Mining)	자연어 처리(Natural Language Processing) 기술을 사용해 인간의 언어로 쓰인 비 정형 텍스트에서 유용한 정보를 추출하거나 다른 데이터와의 연계성을 파악하여, 분류나 군집화 등 빅데이터에 숨겨진 의미 있는 정보를 발견하는 것
웹 마이닝 (Web Mining)	인터넷에서 수집한 정보를 데이터 마이닝 기법으로 분석하는 것
오피니언 마이닝 (Opinion Mining : 평판 분석)	■다양한 온라인 뉴스와 소셜 미디어 코멘트, 사용자가 만든 콘텐츠에서 표현된 의견을 추출·분류·이해하고 자산화하는 컴퓨팅 기술 ■텍스트 속의 감성과 감동, 여러 가지 감정 상태를 식별하려고 감성 분석 사용 ■마케팅에서는 버즈(Buzz:입소문 분석)이라고도 한다.
소셜 네트워크 분석 (Social Network Analysis)	수학의 그래프 이론(Graph Theory)을 바탕으로 소셜 네트워크 서비스에서 소셜 네트워크 연결 구조와 연결 강도를 분석하여 사용자의 명성 및 영향력을 측정하는 것
분류(Classification)	■미리 알려진 클래스로 구분되는 학습데이터(TrainingSet)를 학습시켜 새로 추가되는 데이터가 속할 만한 데이터군을 찾는 지도 학습(Supervised Learning) 방법 ■가장 대표적인 방법으로 KNN(K-Nearest Neighbor)이 있다.
군집(Clustering)	■특성이 비슷한 데이터를 합쳐 군(Group)으로 분류하는 학습 방법 ■분류와 달리 훈련 데이터군을 이용하지 않기 때문에 비지도 학습(Unsupervised Learning) 방법 ■트위터에서 주로 사진/카메라를 논의하는 사용자군과 게임에 관심 있는 사용자군 등 관심사나 취미에 따라 분류

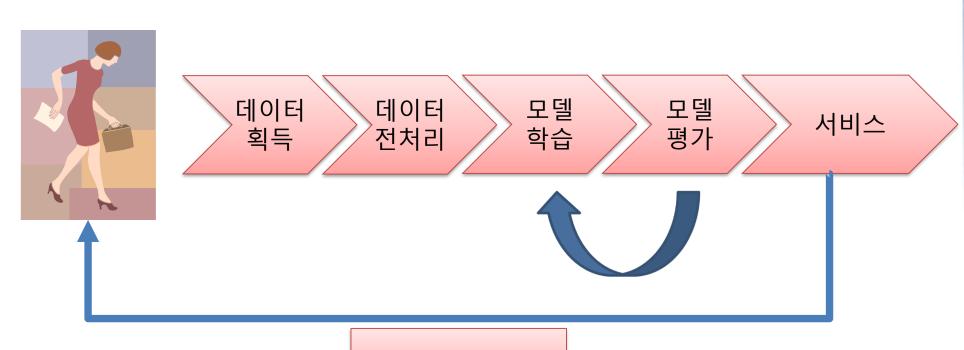


● 통계/분석 프로세스





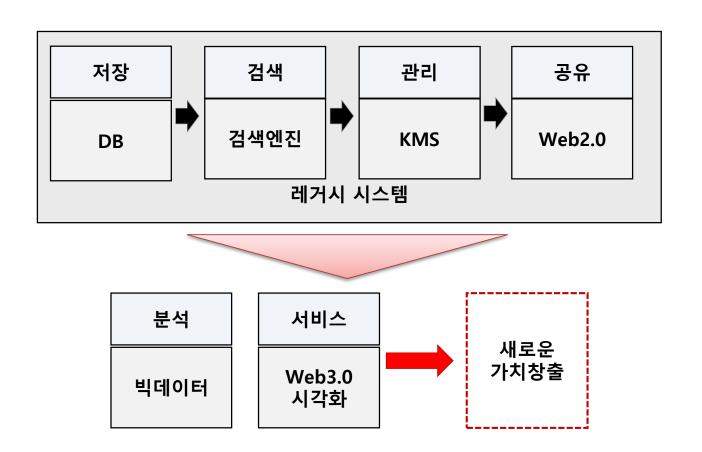
● 머신러닝 프로세스



최적화



#### ● 빅데이터의 발전



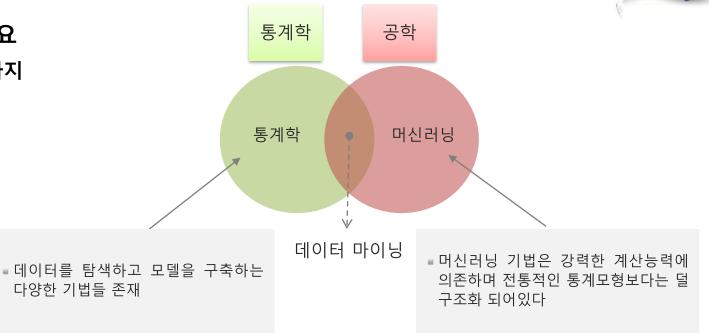
Part IV : 분석/시각화

## 4.2 통계/데이터마이닝/머신러닝

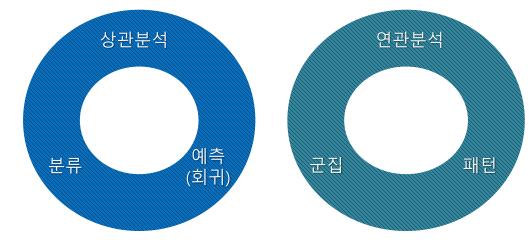




- 빅데이터 분석 개요
- 통계에서 머신러닝까지
  - 통계
  - 데이터마이닝
  - 머신러닝



- 주요분석기법
  - 상관분석
  - 분류분석
  - 예측(회귀)
  - 군집분석
  - 연관분석



## 2. 기초 통계



#### ■ 기초 통계

- 기술 통계, 빈도 및 추정 테이블, 가설 검정, 상관과 공분산, T-테스트 등

#### ■ 기술통계(Descriptive Statistics)

- 자료를 요약하는 기초적인 통계

- 기술통계량 : 데이터수 개수, 평균, 분산, 표준편차, 최소값, 1분위값, 중앙값, 3분위값, 최대값, 왜도, 첨도 등

-> 대표값 : 평균, 중앙값, 최빈치

-> 분포 : 분산, 표준편차

-> 대칭 : 왜도

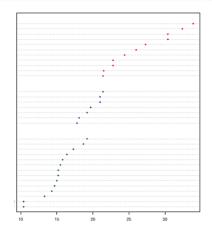
- R 예제(표): summary()

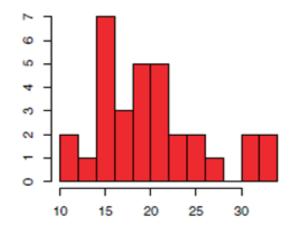
	Descriptive statistics via summary()					
		> summary(	mtcars[vars])			
m	mpg hp wt					
Min.	:10.4	Min.	:52.0	Min.	:1.51	
1st Qu.	:15.4	1st Qu.	:96.5	1st Qu.	:2.58	
Median	:19.2	Median	:123.0	Median	:3.33	
Mean	:20.1	Mean	:146.7	Mean	:3.22	
3rd Qu.	:22.8	3rd Qu.	:180.0	3rd Qu.	:3.61	
Max.	:33.9	Max.	:335.0	Max.	:5.42	

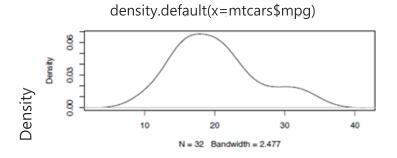
# 2. 기초 통계

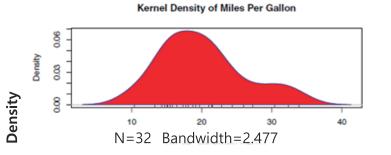
#### ■ 분포 시각화

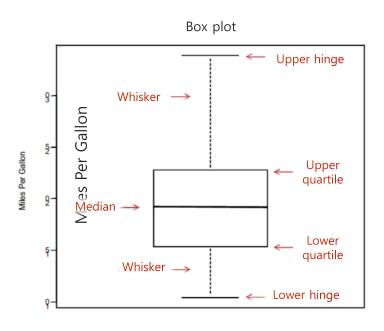
- 산포도
- 히스토그램
- 핵밀도 그래프
- 박스 그래프
- 파이 차트
- 라인 차트











## 3. 군집

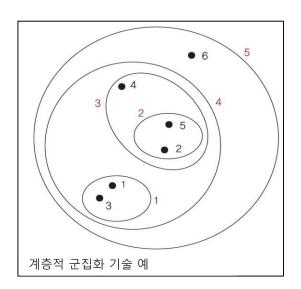


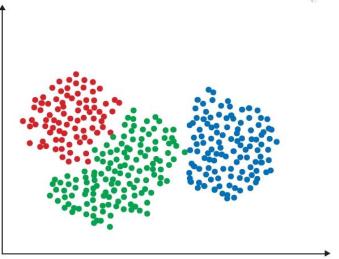
#### ■ 군집

- 개요
- 군집 방법 계층적 군집화 기술 분할적 군집화 기술
- 군집 알고리즘

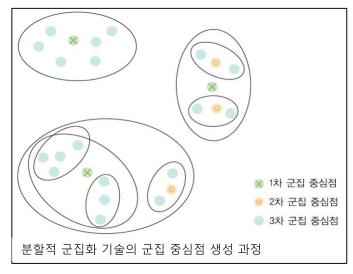
  K-Means 알고리즘

  EM(Expectation Maximization) 알고리즘





군집화 기술

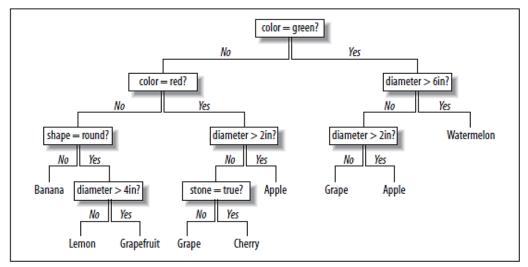


## 4. 분류



#### ■ 분류

- 분류 개요
  - 지도학습, 어떤 클래스에 속하는지 예측
- 의사결정나무의 장단점
  - 가장 대표적인 분류 기법. 설명력이 높다.
- 모델 구조
  - 의사결정 영역:사각형, 단말 영역
- 처리 과정(재귀 순환)
  - 2개(오른쪽 가지, 왼쪽 가지)의 유형으로 분류하는데 가장 적합한 속성과 그 분리값을 찾는다.
  - 분기된 가지(오른쪽, 왼쪽)로 간 다음, 아이템들의 라벨이 모두 같으면 그 값을 반환한다.
  - 각 가지에 속한 아이템의 라벨이 동일하지 않으면 처음 (1)로 간다



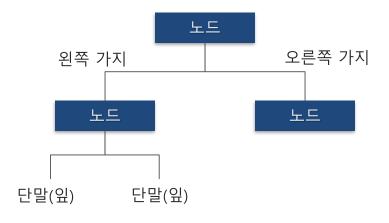


Figure 12-1. Example decision tree

### 5. 추천시스템



#### ■ 추천시스템

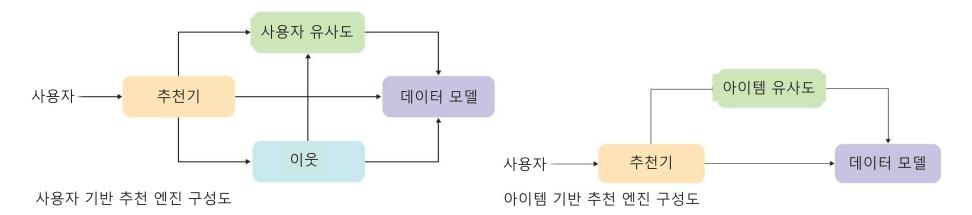
- 구매 가능성이 높은 상품을 고객에게 추천
- 대표적인 방법: 협업필터링과 연관규칙

#### ■ 협업필터링

- 사용자 기반
- 아이템 기반

#### ■ 연관규칙

- (A->B)의 지지도 = P(A∩B) = N(A∩B) / N(T)
- (A->B)의 신뢰도 = P(B|A) = P(A∩B) / P(A)
- (A->B)의 향성도 = P(B|A) / P(B) = P(A∩B) / P(A)P(B)



## 6. 머신러닝

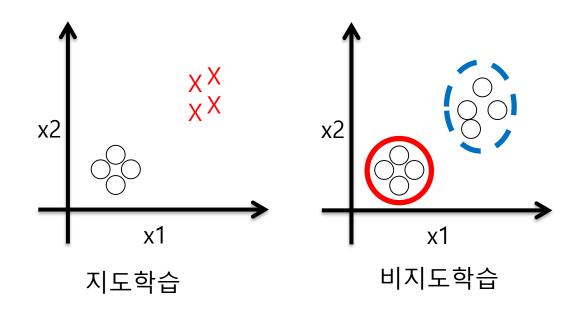


■ 머신러닝의 주요 알고리즘

신경망	사례기반 추론			
데이터 마이닝	패턴 인식			
의사결정 트리	강화학습			
유전자 알고리즘				

#### ■ 학습 방식의 분류

- 지도학습
  - Label 데이터가 있음
  - 분류, 회귀
- 비지도학습(자율학습)
  - Label 데이터가 없음
  - 군집, 차원축소
- 기타: 준지도학습, 강화학습



Part IV : 분석/시각화

## 4.3 빅데이터 분석 기술



## 1. 빅데이터 분석 기술



#### ● 대표적인 분석 기술

#### - R

- ■오픈소스 통계분석 소프트웨어. 기본적인 통계분석부터 최신 머신러닝까지 다양한 패키지를 지원. 하둡과 연동이 가능
- R, RStudio
- RHadoop, RHive , SparkR

#### - Python

- ■데이터마이닝과 머신러닝을 지원하는 다양한 패키지
- ■통계학과의 기본적인 프로그래밍 언어로 정착
- ■스크립트 기반의 대화형 분석 환경 지원
- iPython Notebook

#### - Mahout

- ■추천시스템, 분류, 군집 등 머신러닝 기능을 지원
- 하둡 기반의 머신러닝 Java 라이브러리
- 현재 Spark의 MLlib로 발전함

#### - Spark

- 대화형 분석, 머신러닝, SQL, 그래프알고리즘 등 다양한 분석 가능
- Spark Core, Spark MLlib, SparkSQL, GraphX, Spark Streaming



### 1) 협업필터링(CF) 개요

#### - 추천의 원리

- 1. 특정 사용자와 취향이 비슷한 사람들이
- 2. 좋아할만한
- 3. 아이템 중에서
- 4. 특정 사용자가 구매하지 않은 아이템 목록(ex. 5개)을 반환

#### - 알고리즘의 구분

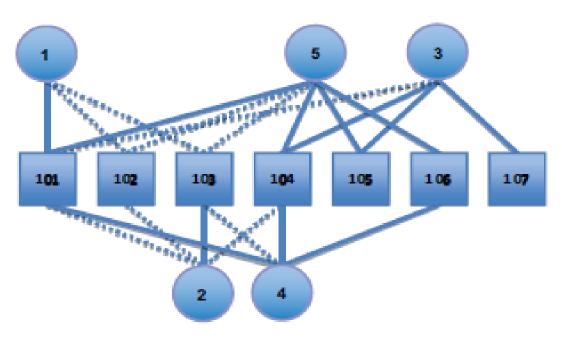
- 1. 사용자 기반 : 사용자와 아이템의 관계
- 2. 아이템 기반 : 아이템의 속성
- 3. 콘텐츠 기반



### 1) 협업필터링(CF) 개요

- 데이터 형식

1. 사용자	1,101,5.0 1,102,3.0 1,103,2.5	4,104,4.5 4,106,4.0 5,101,4.0
2. 아이템	2,101,2.0 2,102,2.5 2,103,5.0	5,102,3.0 5,103,2.0 5,104,4.0 5,105,3.5
3. 선호도	2,104,2.0 3,101,2.5 3,104,4.0 3,105,4.5 3,107,5.0	5,106,4.0
	4,101,5.0 4,103,3.0	

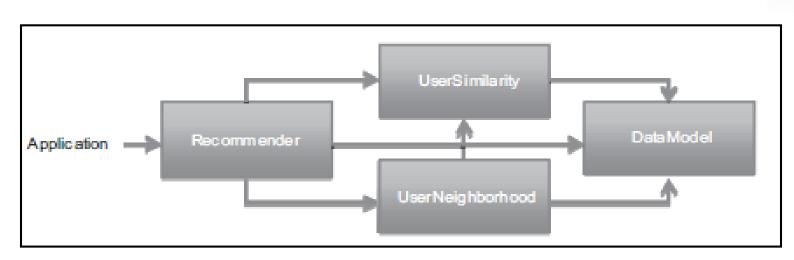


- 특정 사용자와 비슷한 취향을 가진 사람들은?
- 1. 사용자1과 비슷한 그룹?
- 2. 사용자1과 다른 그룹?



### 2) 추천시스템 : Mahout 모듈

- 1. Recommender
- -> 추천기
- 2. UserSimilarity
- -> 사용자 유사도 계산
- 3. UserNeighborhood
- -> 사용자 그룹
- 4. DataModel
- -> 데이터 모델





### 2) 추천시스템 : Mahout 소스코드

```
class RecommenderIntro {
 public static void main(String[] args) throws Exception {
    DataModel model =
      new FileDataModel (new File("intro.csv")): Load data file
    UserSimilarity similarity =
      new PearsonCorrelationSimilarity (model);
    UserNeighborhood neighborhood =
      new NearestNUserNeighborhood (2, similarity, model);
    Recommender recommender = new GenericUserBasedRecommender (
        model, neighborhood, similarity);
                                                       1 Create
                                                        recommender engine
    List<RecommendedItem> recommendations =
        recommender.recommend(1, 1);
                                                                  For user I.
                                                                   recommend
    for (RecommendedItem recommendation : recommendations) {
                                                                   Litem
      System.out.println(recommendation);
```



### 3) 사용자 유사도의 계산

- 피어슨 상관( Pearson Correlation )

값의 범위 : -1 ~ 1

-1 : 상반

0 : 중립

1 : 유사

Table 4.1 The Pearson correlation between user 1 and other users based on the three items that user 1 has in common with the others

	Item 101	item 102	Item 103	Correlation with user 1
User 1	5.0	3.0	2.5	1.000
User 2	2.0	2.5	5.0	-0.764
User 3	2.5	-	-	-
User 4	5.0	-	3.0	1.000
User 5	4.0	3.0	2.0	0.945

Note A user's Pearson correlation with itself is always 1.0.

#### # 문제점

- 1. 아이템의 개수를 고려하지 않음
- 2. 두 사용자의 중복 아이템이 1개인 경우 -> 값을 구할 수 없음
- 3. 사용자의 선호도가 모두 같을 때



### 3) 사용자 유사도의 계산

- 스피어만 상관( Spearman Correlation )

피어슨 상관과 유사

-> 절대값이 아닌 상대적인 선호값의 차이를 기준으로 계산

Table 4.3 The preference values transformed into ranks, and the resulting Spearman correlation between user 1 and each of the other users

	Item 101	item 102	Item 103	Correlation to user 1
User 1	3.0	2.0	1.0	1.0
User 2	1.0	2.0	3.0	-1.0
User 3	1.0	-	-	-
User 4	2.0	-	1.0	1.0
User 5	3.0	2.0	1.0	1.0



### 3) 사용자 유사도의 계산

- 유클리드 거리( Euclidean Distance )

d = 두 사용자의 유클리드 거리

유사도 = 1/(1+d)

Table 4.2 The Euclidean distance between user 1 and other users, and the resulting similarity scores

	Item 101	Item 102	Item 103	Distance	Similarity to user 1
User 1	5.0	3.0	2.5	0.000	1.000
User 2	2.0	2.5	5.0	3.937	0.203
User 3	2.5	-	-	2.500	0.286
User 4	5.0	-	3.0	0.500	0.667
User 5	4.0	3.0	2.0	1.118	0.472



### 3) 사용자 유사도의 계산

- 타니모토 계수( Tanimoto Coefficient )

선호값은 무시하고 동일한 아이템의 비율로 계산

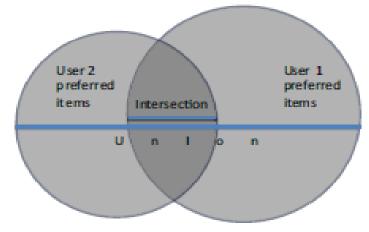


Table 4.4 The similarity values between user 1 and other users, computed using the Tanimoto coefficient. Note that preference values themselves are omitted, because they aren't used in the computation.

	Item 101	Item 102	Item 103	Item 104	Item 105	Item 106	Item 107	Similarity to user 1
User 1	Х	X	X					1.0
User 2	X	X	X	X				0.75
User 3	X			X	X		Х	0.17
User 4	X		X	X		X		0.4
User 5	X	X	X	X	X	Х		0.5



### 4) Mahout + 맵리듀스로 아이템 기반 추천기 구현

- 동시발생 메트릭스( Co-Occurrence Matrix )

아이템 기반 추천기

사용자의 유사도는 고려하지 않음

아이템의 유사도를 계산하는 대신 두 아이템 동시 발생 빈도수를 계산

Table 6.1 The co-occurrence matrix for items in a simple example data set. The first row and column are labels and not part of the matrix.

	101	102	103	104	105	106	107
101	5	3	4	4	2	2	1
102	3	3	3	2	1	1	0
103	4	3	4	3	1	2	0
104	4	2	3	4	2	2	1
105	2	1	1	2	2	1	1
106	2	1	2	2	1	2	0
107	1	0	0	1	1	0	1

1,101,5.0 1,102,3.0 1,103,2.5 2,101,2.0
2,102,2.5 2,103,5.0 2,104,2.0
3,101,2.5 3,104,4.0 3,105,4.5 3,107,5.0
4,101,5.0 4,103,3.0 4,104,4.5
4,106,4.0 5,101,4.0 5,102,3.0
5,103,2.0 5,104,4.0 5,105,3.5 5,106,4.0
5,106,4.0



### 4) Mahout + 맵리듀스로 아이템 기반 추천기 구현

#### - 추천기 구현

ex. 사용자3에게 아이템을 추천

동시 발생 메트릭스와 사용자3의 선호도 벡터를 행렬 곱하여 계산

	101	102	103	104	105	106	107	
101	5	3	4	4	2	2	1	
102	3	3	3	2	1	1	0	
103	4	3	4	3	1	2	0	
104	4	2	3	4	2	2	1	
105	2	1	1	2	2	1	1	
106	2	1	2	2	1	2	0	
107	1	0	0	1	1	0	1	

U3	
2.0	
0.0	
0.0	=
4.0	
4.5	
0.0	
5.0	

Х

R	
40.0	
18.5	
24.5	
40.0	
26.0	
16.5	
15.5	



### 5) 기타

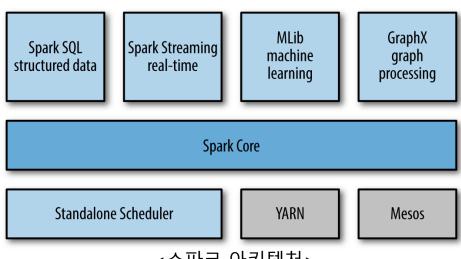
- 사용자 기반 추천기의 단점
  - 사용자가 로그인을 하지 않으면?
  - 사용자가 많으면 계산 비용이 기하급수적으로 증가함
- 아이템 기반 추천기의 단점
  - 사용자 기반에 비해 상대적으로 계산 비용이 과다하지는 않지만
  - 아이템의 개수가 많으면 계산 비용이 높아서
  - 사용자 기반과 마찬가지로 서비스가 불가능함

### 3. Spark



#### Spark

- 개요
  - Apache Spark<sup>™</sup> is a fast and general engine for large-scale data processing.
  - UC 버클리 AMP Lab에서 시작된 대용량 분산병렬처리 기술
  - 하둡의 맵리듀스의 제약을 다양한 연산 제공으로 해결
  - ■메모리 활용으로 성능 향상
- 스파크 설계 목표
  - 과거 데이터에 대한 빠른 대화형 질의 가능
  - ■실시간으로 들어오는 스트리밍 데이터에 대한 질의 가능
  - ■복잡한 분석이 가능



<스파크 아키텍처>

## 3. Spark



#### Spark MLlib

- Spark MLlib 개요
  - 개요 : MLlib는 머신러닝 기능들을 모아 놓은 스파크 라이브러리
  - 데이터 소스
    - · 지원 스토리지 : 분산파일시스템(HDFS, S3 등), 데이터베이스(Mysql, Oracle 등), NoSQL(HBase 등)
    - · 파일 포맷 : Text, JSON, CSV, Hadoop Format
- Spark MLlib 지원 기법
  - ・ 분류 및 회귀
  - · 추천
  - ㆍ 군집
  - ㆍ 차원 축소
  - ・ 빈발 패턴 마이닝

## 3. Spark



#### ● Spark MLlib의 예제

- MLlib를 활용한 분류기의 구현 예제 : 의사결정나무(Decision Tree Classifier)

scala> import org.apache.spark.mllib.tree.DecisionTree 클래스 **Import** scala > import org.apache.spark.mllib.tree.model.DecisionTreeModel scala > val data = MLUtils.loadLibSVMFile(sc, "file:///data/spark-1.6.0/data/mllib/sample\_libsvm\_data.txt") scala > val splits = data.randomSplit(Array(0.7, 0.3)) Data Loading scala> val (trainingData, testData) = (splits(0), splits(1)) scala> val numClasses = 2 scala> val categoricalFeaturesInfo = Map[Int, Int]() 파라메터 scala> val impurity = "gini" 설정 scala> val maxDepth = 5 scala> val maxBins = 32 학습 scala > val model = DecisionTree.trainClassifier(trainingData, numClasses, categoricalFeaturesInfo, impurity, maxDepth, maxBins) scala > val labelAndPreds = testData.map { point => val prediction = model.predict(point.features) 모델 (point.label, prediction) } scala> val testErr = labelAndPreds.filter(r => r.\_1 != r.\_2).count.toDouble 평가 scala > model.save(sc, "myModel DTree")

scala > val DTModel = DecisionTreeModel.load(sc, "myModel DTree")

모델저장 및 로딩 Part IV : 분석/시각화

## 4.4 빅데이터 시각화





A Picture is Worth a Thousand Words

Visualization is a critical component of data analysis

Graphics are the most efficient way to digest large volumes of data & identify trends

Graphical design is a mixture of mathematical and perceptual science

[Yann Abraham, "Elegant Graphics for Data Analysis with ggplot2", BaselR, 2010.04]



🔷 시각화의 정의

정의 : Visualization은 숫자를 공간에 배치해서 보여줌으로써 그 패턴을 인지하게 만드는 것이다. 통계/분석의 가장 중요한 부분.

- 1. 데이터의 특성을 쉽게 파악( vs 통계량 )
- 2. 결과 레포트에 활용
  - -> 독립된 학문/기술 분야로 발전

근거: 사람은 탁월한 패턴 인식 능력이 있음.

데이터 패턴 파악의 도구, 분석을 돕는 역할

-> 데이터의 스토리텔링 : 소통을 위한 전달체, 재미/유머/오락성

#### ♦ Anscombe's quartet( 4세트의 데이터 )

I		II		III		IV	
X	у	X	у	X	у	X	у
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

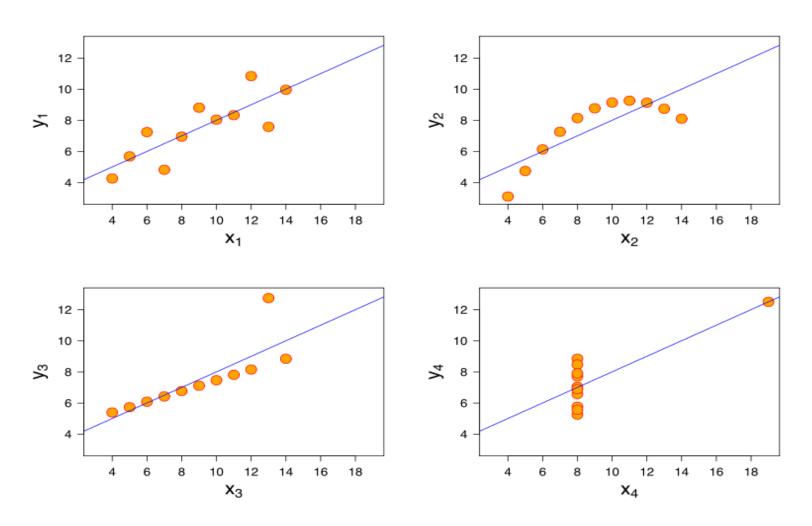
[데이터셋]

Property	Value				
<b>Mean</b> of <i>x</i> in each case	9 (exact)				
<b>Variance</b> of <i>x</i> in each case	11 (exact)				
<b>Mean</b> of <i>y</i> in each case	7.50 (to 2 decimal places)				
<b>Variance</b> of <i>y</i> in each case	4.122 or 4.127 (to 3 decimal places)				
Correlation between x and y in each case	0.816 (to 3 decimal places)				
<b>Linear regression</b> line in each case	y = 3.0 + 0.5x (to 2 and 3 decimal places, respectively)				

#### [ 기초통계량 ]



🤷 Anscombe's quartet -> Trellis Graph(다수의 그래프 비교)



- **💠** 시각화 공부하기
- 이론 교재

#### VISUALIZE THIS / 비주얼라이즈 디스

- 네이선 야우 | 송용근 옮김 | 에이콘 | 2012.04

#### 데이터 포인트

- 네이선 야우 | 서경진 옮김, 비제이퍼블릭, 2013.11

#### ■ 실습 교재

#### **R In Action**

- Kabacoff, Robert | Manning | 2011.09

## 인포그래픽 & 스토리텔링

- 🔷 인포그래픽 & 스토리텔링
- 뉴욕타임즈

Computer Assisted Reporting 편집국

데이터 인터랙티브, 그래픽, 지도만 작성하는 개별 부서

데이터 수치에 관련한 기사만 보도

http://datafl.ws/nytimes

■ 한스 로슬링

스토리텔링 분야의 최고 전문가, 국제 보건학회 교수, 갭마인더 리더

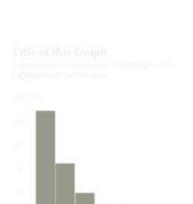
http://datafl.ws/hans

시사기획 창 : 빅데이터 세상을 바꾸다.

http://www.gapminder.org/

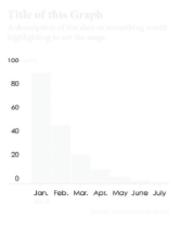
- 1.시각적 단서들
- 2.좌표계

- 3.척도
- 4.맥락



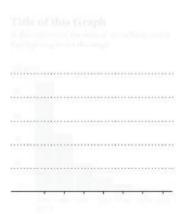
#### **Visual Cues**

Visualization involves encoding data with shapes, colors, and sizes. Which cues you choose depends on your data and your goals.



#### Scale

Increments that make sense can increase readability, as well as shift focus.

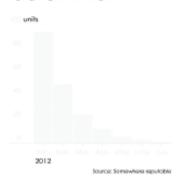


## Coordinate System

You map data differently with a scatterplot than you do with a pie chart. It's x- and y-coordinates in one and angles with the other; it's cartesian versus polar.

#### Title of this Graph

A description of the data or something worth highlighting to set the stage.



#### Context

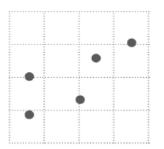
If your audience is unfamiliar with the data, it's your job to clarify what values represent and explain how people should read your visualization.



## ■ 1.시각적 단서들

#### **Position**

Where in space the data is



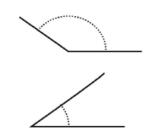
#### Length

How long the shapes are



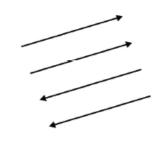
#### Angle

Rotation between vectors



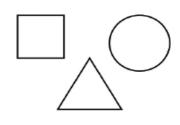
#### Direction

Slope of a vector in space



#### Shapes

Symbols as categories



#### Area

How much 2-D space



Volume

How much 3-D space



FIGURE 3-3 Visual cues

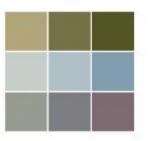
## Color saturation

Intensity of a color hue



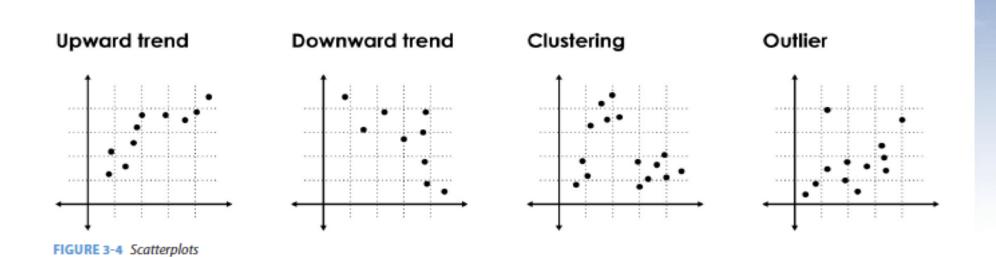
## Color hue

Usually referred to as color





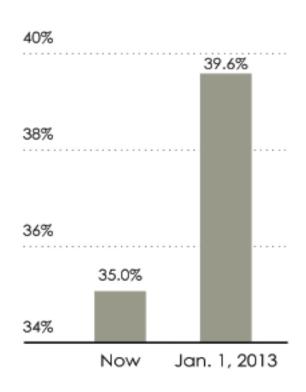
■ 1.시각적 단서들 - (1)위치





## ■ 1.시각적 단서들 - (2)길이

## Axis starting at 34 percent



## Axis starting at 0 percent

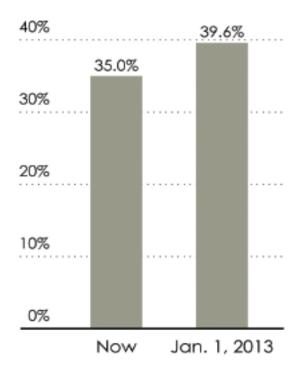


FIGURE 3-5 Incorrect bar graph on left and correct one on the right



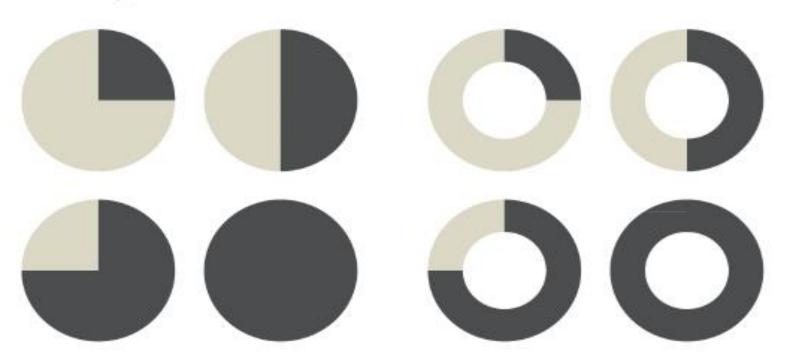
■ 1.시각적 단서들 - (3)각도

**Pies** 

The visual cue is the relative degrees in the circle.

## **Donuts**

Arc length is the visual cue because the center is cut out.

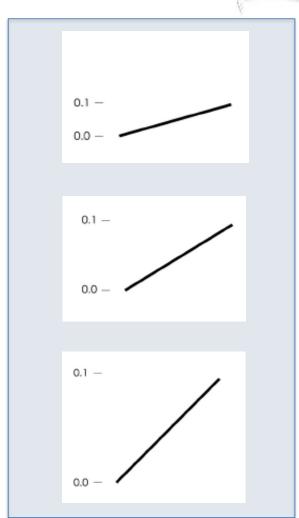


## ■ 1.시각적 단서들 - (4)방향

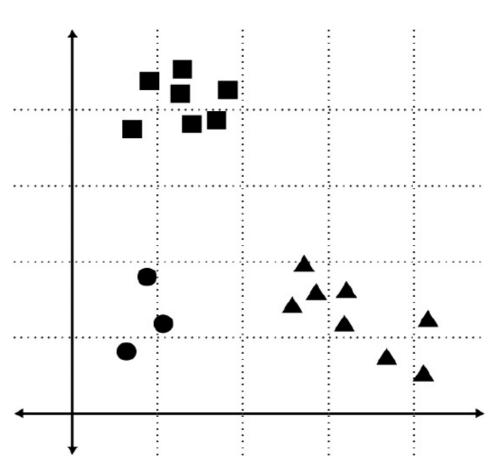
# Direction in a time series 20 metric tons of CO2 per capita in Australia 15 10 5 1960 1970 1980 1990 2000

Source: The World Bank

FIGURE 3-7 Slope and time series



■ 1.시각적 단서들 - (5)형태(도형)



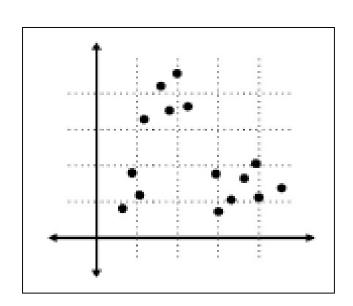


FIGURE 3-9 Different shapes in scatterplot

■ 1.시각적 단서들 - (6,7)면적과 부피

## Sizing by area

This is one unit.

Four units sized by area

4 times the area as unit square by side length

Four units incorrectly sized

16 times the area as unit square



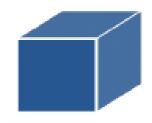
■ 1.시각적 단서들 - (6,7)면적과 부피

## Sizing by volume

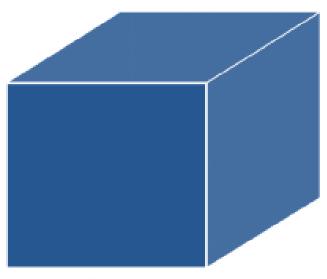
This is one unit.



Four units sized by volume



4 times the volume as unit cube Four units incorrectly sized by edge length



32 times the volume as unit square

FIGURE 3-10 Squares and cubes sized by different dimensions

■ 1.시각적 단서들 - (8,9)색상 : 색조/채조

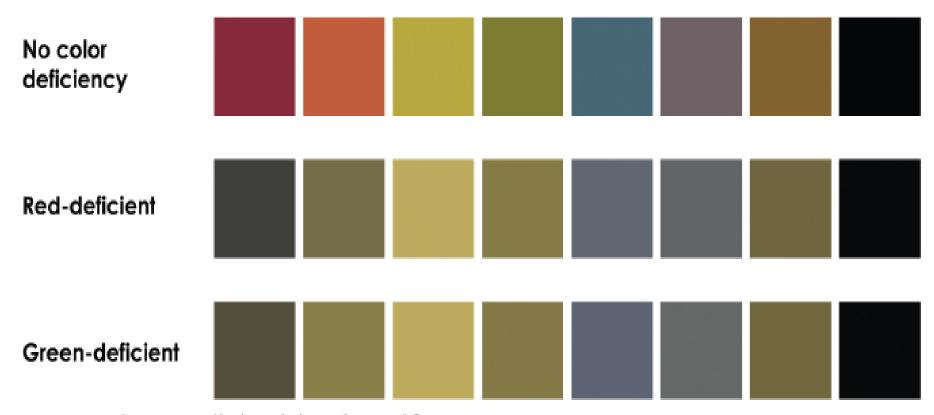
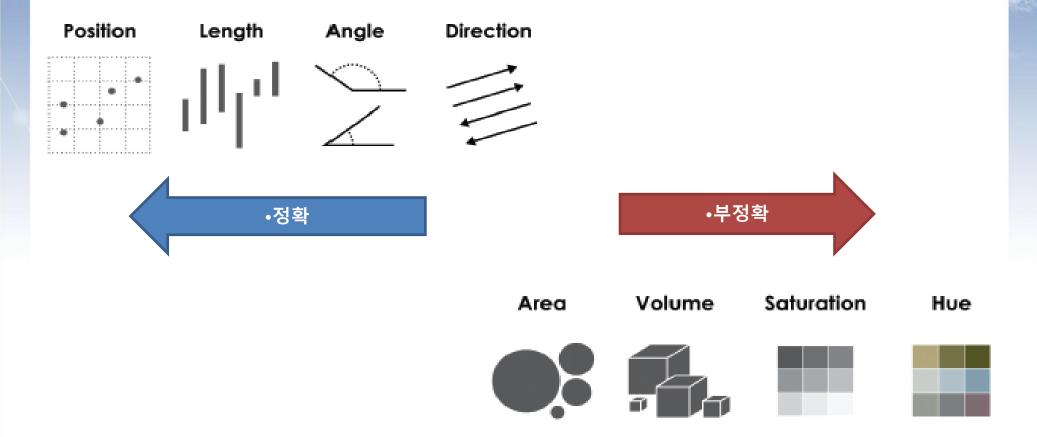


FIGURE 3-11 Colors as perceived by those who have color vision deficiencies

■ 1.시각적 단서들의 인지

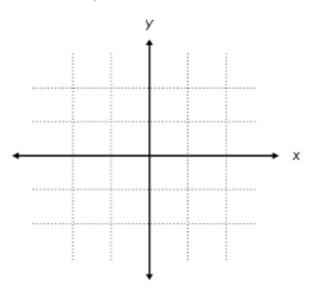




## ■ 2.좌표계

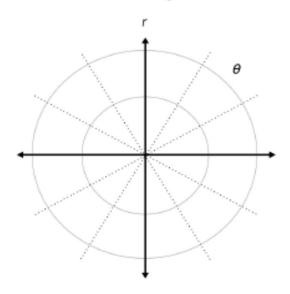
## Cartesian

If you've ever made a graph, the xand y-coordinate system will look familiar to you.



## **Polar**

Pie charts use this system. Coordinates are placed based on radius r and angle  $\theta$ .





■ 2.좌표계 (지도)

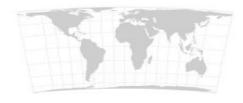
#### Equirectangular

Typically used for thematic mapping, but doesn't preserve area or angle



#### Albers

Scale and shape not preserved; angle distortion is minimal



#### Mercator

Preserves angles and shapes in small areas, making it good for directions



#### Lambert conformal conic

Better for showing smaller areas and often used for aeronautical maps.



#### Sinusoidal

Preserves area; useful for areas near the prime meridian



#### Polyconic

Was often used to show US in the mid-1900s; little distortions in small areas near merdian



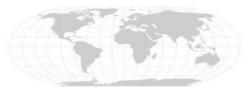
#### **Winkel Tripel**

Minimized area, angle, and distance distortion; good choice for world map



#### Robinson

A compromise between preserving areas and angles; good to show world map



#### Orthographic

Representing a 3-D object in 2-D, need to rotate to area of interest

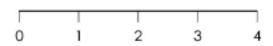




## ■ 3.척도

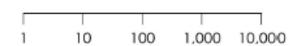
## Linear

Values are evenly spaced



## Logarithmic

Focus on percent change



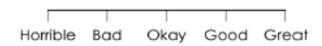
## Categorical

Discrete placement in bins



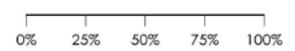
## Ordinal

Categories where order matters



### Percent

Representing parts of a whole



## **Time**

Units of months, days, or hours



FIGURE 3-15 Scales

■ 4.맥락(Context)



Sample —



FIGURE 3-20 Measuring Type (2010) by Matt Robinson and Tom Wrigglesworth, http://datafl.ws/27m

# 잘 표현하기



■ 좌표계 + 시각적단서들

	Position	Length	Angle
Coordinate systems			
Cartesian			
Polar		<b>₩</b>	
Geographic			

# 잘 표현하기

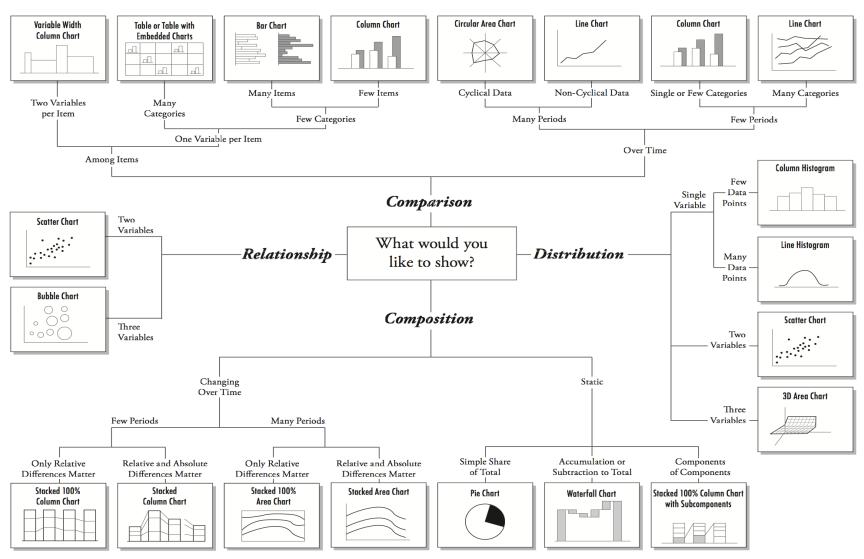


Direction	Shapes	Area or Volume	Color

## 잘 표현하기



## Chart Suggestions—A Thought-Starter



# 감사합니다.