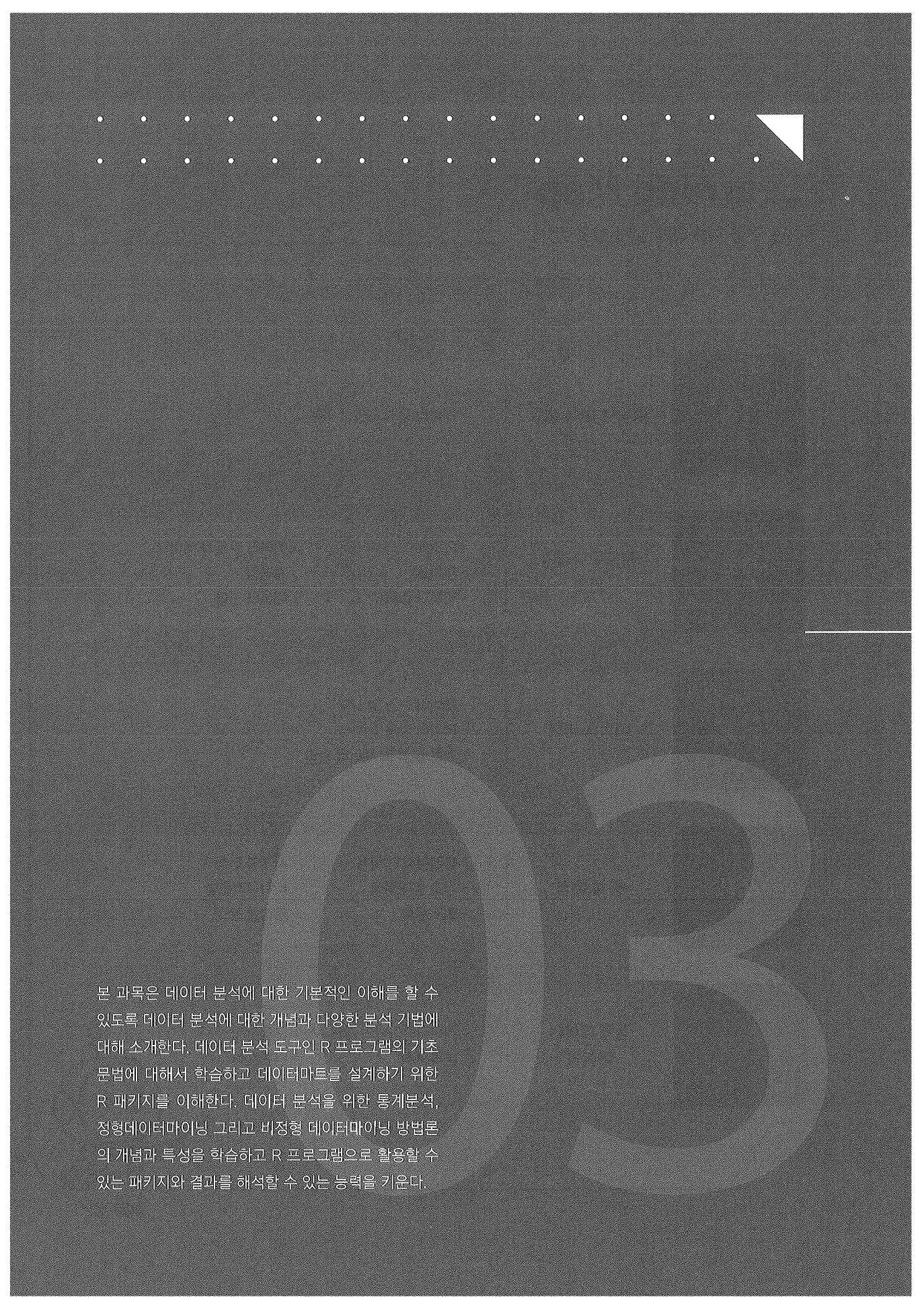
PART

03

데이터 분석



본 과목은 데이터 분석에 대한 기본적인 이해를 할 수 있도록 데이터 분석에 대한 개념과 다양한 분석 기법에 대해 소개한다. 데이터 분석 도구인 R 프로그램의 기초 문법에 대해서 학습하고 데이터마트를 설계하기 위한 R 패키지를 이해한다. 데이터 분석을 위한 통계분석, 정형데이터마이닝 그리고 비정형 데이터마이닝 방법론의 개념과 특성을 학습하고 R 프로그램으로 활용할 수 있는 패키지와 결과를 해석할 수 있는 능력을 키운다.

Learning Map

어떤 것을 학습하게 될지 살펴보자!

1장 데이터 분석 개요

- 데이터 분석 기법의 이해

2장 R 프로그래밍 기초

- R 소개
- R 기초
- 입력과 출력
- 데이터 구조와 데이터 프레임
- 데이터 변형

3장 데이터 마트

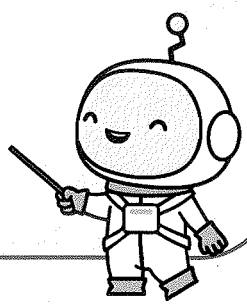
- 데이터 변경 및 요약
- 데이터 가공
- 기초 분석 및 데이터 관리

4장 통계 분석

- 통계분석의 이해
- 기초 통계분석
- 회귀분석
- 시계열 분석
- 다차원척도법
- 주성분 분석

5장 정형 데이터 마이닝

- 데이터마이닝의 개요
- 분류분석
- 군집분석
- 연관분석



PART 03
데이터 분석

1장 데이터 분석 개요

7 DAY

데이터 분석 set을 만들기 위해 어떤 데이터를 활용할 것인지, 어떻게 하면 더 효과적으로 정제되고 안정적인 데이터 set을 가져와서 데이터마트를 구성할 수 있을지 그림을 통해 이해하도록 합니다. 또한 시각화, 공간 분석, 탐색적 자료 분석, 데이터 마이닝, 텍스트 마이닝의 개념과 특성을 알아야 합니다.
본 절에서 2문제 이상의 객관식 및 주관식 문제가 출제될 수 있으니 반드시 외우도록 합시다~

○ 학습 목표

- 데이터 처리 프로세스를 이해한다.
- 데이터 분석 기법 중 시각화를 이해한다.
- 데이터 분석 기법 중 공간분석을 이해한다.
- 데이터 분석 기법 중 탐색적 자료 분석을 이해한다.

○ 눈높이 체크

✓ 데이터 분석을 위해 데이터 마트를 어떻게 만들까요?

대기업에서는 데이터 분석을 위해 데이터웨어하우스(DW)나 데이터마트(DM)에서 데이터를 추출해 옵니다. 또한 운영시스템에서 데이터를 추출하여 분석용 데이터를 구성하게 됩니다. 데이터를 추출 가능한 기업내 여러 시스템의 명칭과 프로세스를 이해하면 보다 효과적으로 분석데이터마트를 구성할 수 있게 됩니다.

✓ 데이터 분석 방법 중 시각화를 들어보셨나요?

데이터 시각화는 데이터를 도표나 그림으로 한눈에 분석내용을 인지할 수 있는 데이터 분석기법으로 가장 낮은 수준의 분석이지만 복잡한 분석보다 더 효율적으로 인사이트를 얻을 수 있습니다. 그래서 빅데이터 분석에서는 필수적인 분석 방법으로 활용되고 있습니다.

✓ 데이터 분석 방법 중 공간분석을 들어보셨나요?

공간분석은 공간적 차원과 관련된 속성을 지도 위에 시각화하여 인사이트를 얻는 방법으로 여러 분야에서 활용되고 있습니다.

✓ 데이터 분석 방법 중 탐색적 자료분석을 들어보셨나요?

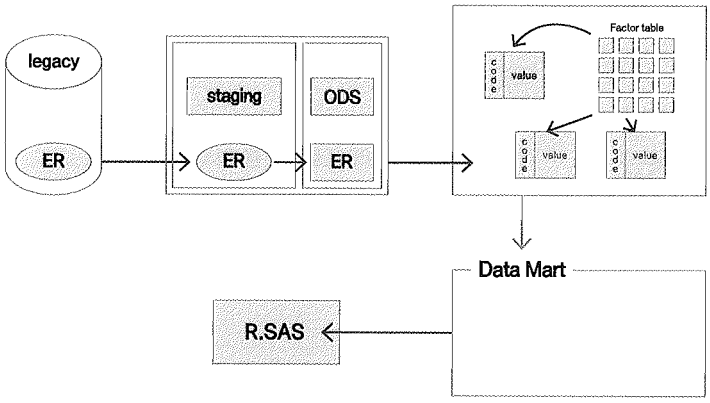
탐색적 자료분석은 다양한 차원과 값을 조합해 특이한 점이나 의미있는 사실을 도출하는 분석으로 변수의 특징과 변수들 간의 관계를 탐색하는 분석 방법입니다.

1절 데이터 분석 기법의 이해

데이터 처리

가. 개요

- 데이터 분석은 통계에 기반을 두고 있지만, 통계지식과 복잡한 가정이 상대적으로 적은 실용적인 분야이다.



나. 활용

- 대기업은 데이터웨어하우스(DW)와 데이터마트(DM)를 통해 분석 데이터를 가져와서 사용한다.
- 신규 시스템이나 DW에 포함되지 못한 자료의 경우, **기존 운영시스템(Legacy)**이나 **스테이징 영역(Staging Area)과 ODS(Operational Data Store)**에서 데이터를 가져와서 DW에서 가져온 내용과 결합하여 활용할 수 있다.
- 하지만 운영시스템에 직접 접근해 데이터를 활용하는 것은 매우 위험한 일이므로 거의 이루어지지 않고 있으며, 스테이징 영역(Staging Area)의 데이터는 운영시스템에서 임시로 저장된 데이터이기 때문에 가급적이면 클렌징 영역인 ODS에서 데이터의 전처리를 해서 DW나 DM과 결합하여 활용하는 것이 가장 이상적이다.

다. 최종 데이터 구조로 가공

1) 데이터마이닝 분류

- 분류값과 입력변수들을 연관시켜 인구통계, 요약변수, 파생변수 등을 산출한다.

2) 정형화된 패턴 처리

- 비정형 데이터나 소셜 데이터는 정형화한 패턴으로 처리해야 한다.

가) 비정형 데이터

- DBMS에 저장됐다가 텍스트 마이닝을 거쳐 데이터 마트와 통합한다.

나) 관계형 데이터

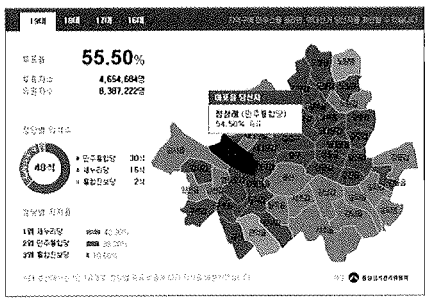
- DBMS에 저장되어 사회 신경망분석을 거쳐 분석결과 통계값이 데이터 마트와 통합되어 활용된다.

② 시각화 (시각화 그래프)

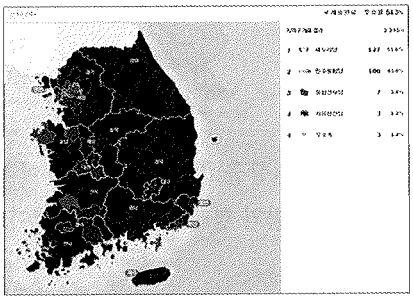
- 시각화는 가장 낮은 수준의 분석이지만 잘 사용하면 **복잡한 분석보다도 더 효율적**이다.
- 대용량 데이터를 다루는 빅데이터 분석에서 **시각화는 필수**이다.
- 탐색적 분석을 할 때 시각화는 필수이다.
- SNA 분석(사회연결망 분석)을 할 때 자주 활용된다.

③ 공간분석 (GIS)

- 공간분석(Spatial Analysis)은 공간적 차원과 **관련된 속성들을 시각화**하는 분석이다.
- 지도 위에 관련 속성들을 생성하고 **크기, 모양, 선 굵기 등으로 구분**하여 인사이트를 얻는다.



<출처 : http://media.daum.net/2012g_election/district/11/>



<출처 : 네이버 19대 총선 페이지>

④ 탐색적 자료 분석 (EDA)

가. 개요

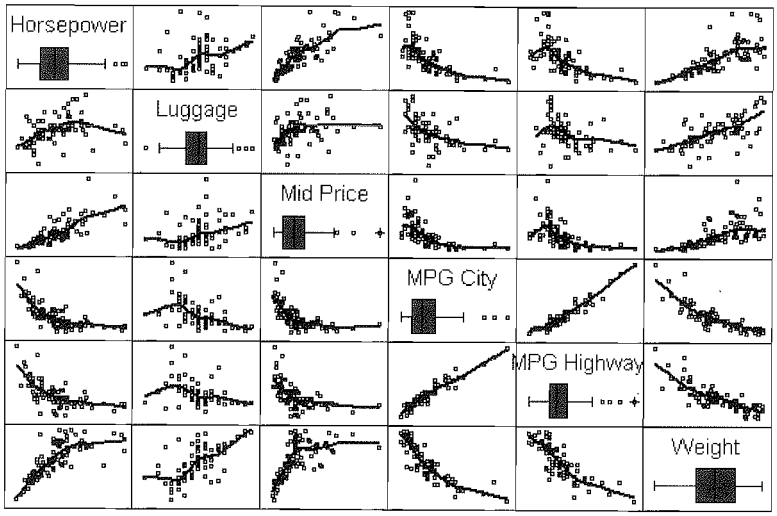
- 탐색적 분석은 다양한 차원과 값을 조합해가며 **특이한 점이나 의미 있는 사실을 도출하고 분석의 최종 목적을 달성해가는 과정**으로 데이터의 특징과 내재하는 **구조적 관계**를 알아내기 위한 기법들의 통칭이다. 프린스톤 대학의 튜키교수가 1977년 저서를 발표함으로 EDA가 등장한다.

나. EDA의 4가지 주제

- 저항성의 강조, 잔차 계산, 자료변수의 재표현, 그래프를 통한 현시성

다. 탐색적 분석의 효율 예

- 2과목 모형개발 프로세스(KDD, CRISP-DM 등)에서 언급한 바와 같이 데이터이해 단계(변수의 분포와 특성 파악)와 변수생성 단계(분석목적에 맞는 주요한 요약 및 파생변수 생성) 그리고 변수선택 단계(목적변수에 의미있는 후보변수 선택)에서 활용되고 있다.



3장에서 다시 다룰 예정이므로 간단히 이해하고 넘어갑시다.

⑤ 통계분석

가. 통계

- 어떤 현상을 종합적으로 한눈에 알아보기 쉽게 일정한 체계에 따라 숫자와 표, 그림의 형태로 나타내는 것이다.

나. 기술통계 (Descriptive Statistics)

- 모집단으로부터 표본을 추출하고 **표본이 가지고 있는 정보를 쉽게 파악**할 수 있도록 데이터를 정리하거나 요약하기 위해 하나의 숫자 또는 그래프의 형태로 표현하는 절차이다.

다. 추측(추론)통계(Inferential Statistics)

- 모집단으로부터 추출된 **표본의 표본통계량으로 부터 모집단의 특성인 모수에 관해 통계적으로 추론**하는 절차이다.

라. 활용분야

- 정부의 경제정책 수립과 평가의 근거자료로 활용(통계청의 실업률, 고용률, 물가지수)
- 농업(가뭄, 수해 또는 병충해 등에 강한 품종의 개발 및 개량)
- 의학(의학적 치료 방법의 효과나 신약 개발을 위한 임상실험의 결과 분석)
- 경영(제품 개발, 품질관리, 시장조사, 영업관리 등에 활용)
- 스포츠(선수들의 체질향상 및 개선, 경기 분석과 전략분석, 선수평가와 기용 등)

6 데이터마이닝

가. 개요

- 대표적인 고급 데이터 분석법으로 **대용량의 자료**로부터 정보를 요약하고 미래에 대한 예측을 목표로 자료에 존재하는 **관계, 패턴, 규칙 등을 탐색**하고 이를 모형화함으로써 이전에 알려지지 않은 **유용한 지식을 추출**하는 분석 방법이다.

나. 방법론

- **데이터베이스**에서의 지식탐색 : 데이터웨어하우스에서 데이터마트를 생성하면서 각 데이터들의 속성을 사전분석을 통해 지식을 얻는 방법이다.
- **기계학습**(Machine Learning) : 인공지능의 한 분야로, 컴퓨터가 학습할 수 있도록 알고리즘과 기술을 개발하는 분야로 **인공신경망, 의사결정나무, 클러스터링, 베이지안 분류, SVM 등**이 있다.
- **패턴인식**(Pattern Recognition) : 원자료를 이용해서 사전지식과 패턴에서 추출된 통계 정보를 기반으로 자료 또는 패턴을 분류하는 방법으로 **장바구니분석, 연관규칙 등**이 있다.

다. 활용분야

- 데이터베이스 마케팅(방대한 고객의 행동정보를 활용해 목표 마케팅, 고객세분화, 장바구니 분석, 추천시스템 등)
- 신용평가 및 조기경보시스템(금융기관에서 신용카드 발급, 보험, 대출 발생시 업무에 적용)
- 생물정보학(세포의 수많은 유전자를 분석하여 질병의 진단과 치료법 또는 신약 개발)
- 텍스트마이닝(전자우편, SNS 등 디지털 텍스트 정보를 통해 고객성향분석, 감성분석, 사회관계망분석 등)

3과목 | 1장

데이터 분석 개요

✓9회 기출

01 데이터가 가지고 있는 특성을 파악하기 위해 해당 변수의 분포 등을 시각화하여 분석하는 분석 방식은 무엇인가?

① 전처리분석
② 탐색적자료분석(EDA)
③ 공간분석
④ 다변량분석

02 데이터 마이닝의 모델링에 대한 설명이다. 설명이 가장 잘못된 것은?

① 데이터마이닝 모델링은 통계적 모델링이 아니므로 지나치게 통계적 가설이나 유의성에 집착하지 말아야 한다.
② 모델링 방법은 여러 가지가 있으므로 모델링 시 반드시 다양한 옵션을 줘서 모델링을 수행하여 최고의 성과를 도출하여야 한다.
③ 분석데이터를 학습 및 테스트 데이터로 6:4, 7:3, 8:2 비율로 상황에 맞게 실시한다.
④ 성능에 집착하면 분석 모델링의 주목적인 실무 적용에 반하여 시간을 낭비할 수 있으므로 훈련 및 테스트 성능에 큰 편차가 없고 예상 성능을 만족하면 중단한다.

✓10회 기출

03 모델링 성능을 평가함에 있어, 데이터마이닝에서 활용하는 평가 기준이 아닌 것은?

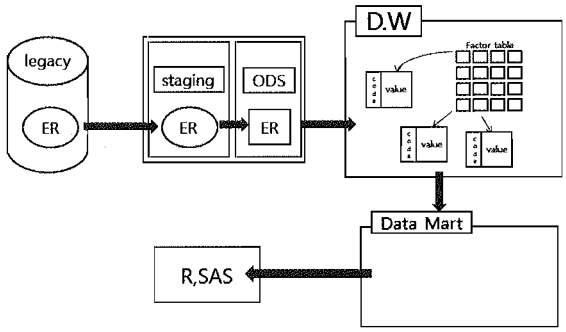
① 정확도(Accuracy)
② 리프트(Lift)
③ 디텍트 레이트(Detect Rate)
④ Throughput

✓14회 기출

04 탐색적 데이터 분석의 목적은 데이터를 이해하는 것이다. 다음 중 이에 대한 설명으로 가장 부적절한 것은?

① 데이터에 대한 전반적인 이해를 통해 분석 가능한 데이터인지 확인하는 단계이다.
② 탐색적 데이터 분석 과정은 데이터에 포함된 변수의 유형이 어떻게 되는지를 찾아가는 과정이다.
③ 데이터를 시각화하는 것만으로는 이상점(Outlier) 식별이 잘 되지 않는다.
④ 알고리즘이 학습을 얼마나 잘 하느냐 하는 것은 전적으로 데이터의 품질과 데이터에 담긴 정보량에 달려 있다.

05 아래의 그림은 데이터 처리 구조를 나타내고 있다. 그림에 대한 설명으로 잘못된 것은?



① 데이터를 분석에 활용하기 위해 데이터웨어하우스와 데이터마트에서 데이터를 가져 온다.
② 신규시스템이나 DW에 포함되지 않은 데이터는 기존 운영시스템(Legacy)에서 직접 데이터를 DW와 전처리 없이 바로 결합하면 된다.
③ ODS는 운영데이터저장소로 기존 운영시스템의 데이터가 정제된 데이터이므로 DW나 DM과 결합하여 분석에 활용할 수 있다.
④ 스테이지 영역에서 가져온 데이터는 정제되어 있지 않기 때문에 데이터를 전처리해서 DW나 DM과 결합하여 사용한다.

06 최근 시각화 기법의 활용이 높아지면서 데이터의 특성을 파악하는데 많은 기여를 하고 있다. 다음 중 최근의 시각화의 발전된 형태가 아닌 것은?

① 텍스트 마이닝에서의 워드 클라우드를 통한 그래프화
② SNA(Social Network Analysis)에서 집단의 특성과 관계를 그래프화
③ 통계소프트웨어의 기초통계정보를 엑셀에서 그래프화
④ Polygon, Heatmap, Mosaic Graph 등의 그래프 작업

07 대표적인 고급분석으로 데이터에 있는 패턴을 파악해 예측하는 분석으로 데이터가 크고 정보가 다양할수록 보다 활용하기 유리한 분석은 무엇인가?

① 시뮬레이션 ② 통계분석 ③ 데이터 마이닝 ④ 시각화

08 모집단으로부터 추출된 표본의 표본통계량으로부터 모집단의 특성인 모수에 관해 통계적으로 추론하는 통계를 무엇이라고 하는가?

① 가공 통계 ② 기술 통계 ③ 통계분석 ④ 추론 통계

09 EDA의 4가지 주제 중 틀린 것은?

① 종속변수 계산
② 저항성의 강조
③ 자료변수의 재표현
④ 그래프를 통한 현시성

10 공간적 차원과 관련된 속성들을 시각화에 추가하여 지도 위에 관련 속성들을 생성하고 크기, 모양, 선 굵기 등으로 구분하여 인사이트를 얻는 분석방법은 무엇인가?

( )

정답 및 해설

01	②	06	③
02	②	07	③
03	④	08	④
04	③	09	①
05	②	10	공간분석(Spatial analysis)

01. EDA(탐색적 자료분석)는 다양한 차원과 값을 조합해가며 특이한 점이나 의미있는 사실을 도출하고 분석의 최종목적을 달성해가는 과정이다.(**정답 : ②**)

02. 반드시 다양한 옵션을 줘서 모델링을 수행하지 않고, 충분한 시간이 있으면 다양한 옵션을 줘서 시도하는 것이고 일정 성과가 나오면 해석과 활용 단계로 진행할 수 있도록 의사결정 해야 한다. (**정답 : ②**)

03. 데이터 마이닝에서는 정확도, 정밀도, 디텍트 레이트, 리프트 등의 값으로 판단하고 시뮬레이션에서는 Throughput, Average Waiting Time, Average Queue Length, Time in System 등의 지표가 활용된다. (**정답 : ④**)

04. 상자그림(Box Plot)등을 그리면 이상치를 식별하기 쉽다. (**정답 : ③**)

05. 신규 시스템이나 스테이징 영역의 데이터는 정제되지 않았기 때문에 정제하고 DW나 DM과 결합해야 한다. (**정답 : ②**)

06. 엑셀의 그래프는 최근 시각화 기술의 발전된 형태가 아니라 기존에 기술이다. (**정답 : ③**)

07. 대용량 데이터에서 패턴을 파악해서 예측하는 분석 방법은 데이터마이닝 방법이다. (**정답 : ③**)

08. 추론(추측)통계는 모집단으로부터 추출된 표본의 표본통계량으로부터 모집단의 특성인 모수에 관한 통계적으로 추론하는 절차이다. (**정답 : ④**)

09. EDA의 4가지 주제는 저항성의 강조, 잔차 계산, 자료변수의 재표현, 그래프를 통한 현시성이다. (**정답 : ①**)

10. 지도위에 공간과 관계된 속성들을 다양한 표현으로 시각화하는 방법은 공간 분석이다.
정답 : 공간분석(Spatial analysis)

PART 03
데이터 분석
2장

R 프로그래밍 기초

8 DAY

○ 학습 목표

- 데이터 분석 환경을 이해한다.
- 데이터 분석 도구 R의 특성을 이해한다.
- R을 설치하고 GUI를 이해한다.
- R Studio를 설치하고 GUI를 이해한다.

○ 눈높이 체크

✓ **데이터 분석을 위해 활용되고 있는 분석 도구에는 어떤 것이 있을까요?**

데이터 분석에 가장 많이 활용되는 분석도구는 SPSS, SAS, R, Python, Stata 등이 있습니다.

✓ **최근 빠른 속도로 확산되고 있는 R 언어를 아시나요?**

최근 R에 대한 관심이 커지면서 많은 분야에서 R을 이용한 실험과 프로젝트가 진행되고 있습니다.

✓ **R GUI 인 R Studio를 들어보셨나요?**

여러분들이 R과 R Studio에 관해 들어보셨거나 관심을 가지고 있다면 보다 좋은 학습 성과를 얻으실 수 있을 것입니다.

1절 R 소개

① 데이터 분석 도구의 현황

가. R의 탄생

- R은 오픈소스 프로그램으로 통계·데이터마이닝과 그래프를 위한 언어이다.
- 다양한 최신 통계분석과 마이닝 기능을 제공한다.
- 세계적으로 많은 사용자들이 다양한 예제를 공유한다.
- 다양한 기능을 지원하는 많은 패키지가 수시로 업데이트 된다.

나. 분석도구의 비교

	SAS	SPSS	오픈소스 R
프로그램 비용	유료, 고가	유료, 고가	오픈소스
설치용량	대용량	대용량	모듈화로 간단
다양한 모듈 지원 및 비용	별도구매	별도구매	오픈소스
최근 알고리즘 및 기술반영	느림	다소느림	매우빠름
학습자료 입수의 편의성	유료 도서 위주	유료 도서 위주	공개 논문 및 자료 많음
질의를 위한 공개 커뮤니티	NA	NA	매우 활발

다. R의 특징

1) 오픈소스 프로그램

- 사용자 커뮤니티에 도움 요청이 쉽다.
- 많은 패키지가 수시로 업데이트 된다.

2) 그래픽 및 성능

- 프로그래밍이나 그래픽 측면 등 대부분의 주요 특징들에서 상용 프로그램과 대등하거나 월등하다.

3) 시스템 데이터 저장 방식

- 각 세션 사이마다 시스템에 데이터셋을 저장하므로 매번 데이터를 로딩할 필요가 없고 명령어 스토리도 저장 가능하다.

4) 모든 운영체제

- 윈도우, 맥, 리눅스 운영체제에서 사용 가능하다.

5) 표준 플랫폼

- S 통계 언어를 기반으로 구현된다.
- R/S 플랫폼은 통계전문가들의 사실상의 표준 플랫폼이다.

6) 객체지향언어이며 함수형 언어

- 통계 기능뿐만 아니라 일반 프로그래밍 언어처럼 자동화거나 새로운 함수를 생성하여 사용 가능하다.

  가) 객체지향 언어의 특징

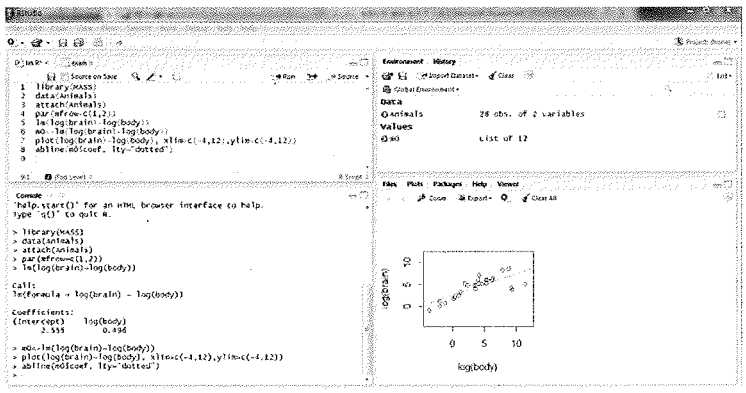
  - SAS, SPSS에서 회귀분석시, 화면에 결과가 산더미로 나오게 된다. 분석 결과를 활용하기 위해서는 추가로 프로그래밍을 하거나 별도의 작업이 필요하다.
  - R은 추정계수, 표준오차, 잔차 등 결과값을 객체(Object)에 저장하여 필요한 부분을 호출하여 쉽게 활용 가능하다.

  나) 함수형 언어의 특징

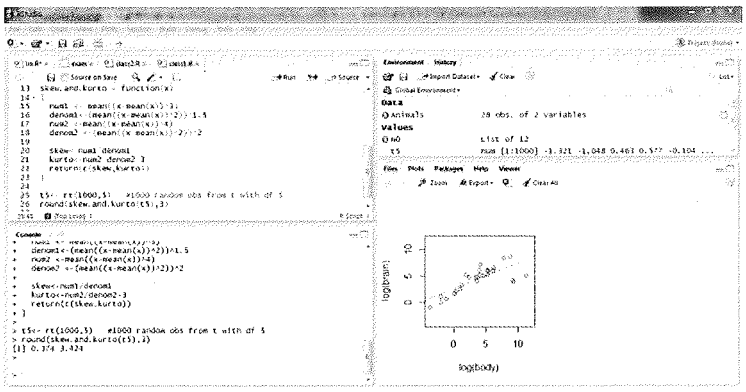
  - 더욱 깔끔하고 단축된 코드
  - 매우 빠른 코드 수행 속도
  - 단순한 코드로 디버깅 노력 감소
  - 병렬 프로그래밍으로의 전환이 더욱 용이

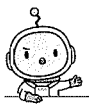
예시

- 객체 지향 언어의 설명



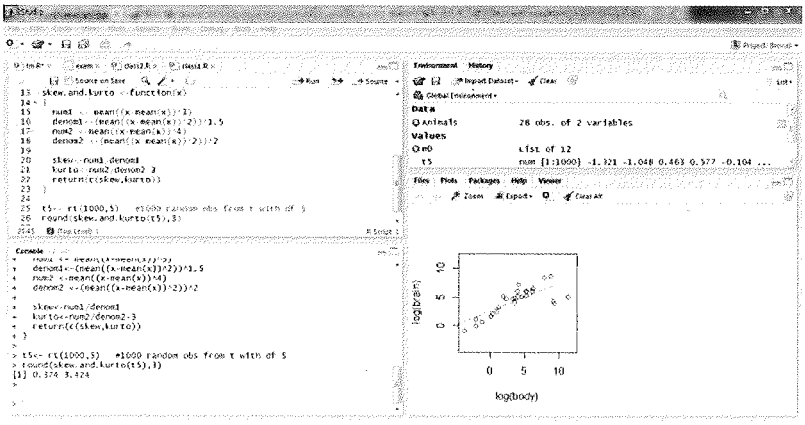
- 함수형 언어의 설명





라. R Studio

- 오픈소스이며 다양한 운영체계를 지원한다.
- R Studio는 메모리에 변수가 어떻게 되어 있는지와 타입이 무엇인지를 볼 수 있고, 스크립트 관리와 도큐먼테이션이 편리하다.
- 코딩을 해야하는 부담이 있으나 스크립트용 프로그래밍으로 어렵지 않게 자동화가 가능하다.
- **래틀(Rattle)**은 GUI가 패키지와 긴밀하게 결합돼 있어 정해진 기능만 사용 가능해 업그레이드가 제대로 되지 않으면 통합성에 문제가 발생할 수 있다.



마. R 기반의 작업 환경

- 작업환경은 업무 규모와 본인에게 익숙한 환경이 무엇인지를 기준으로 선택한다.
- 기업환경에서는 64bit 환경의 듀얼코어, 32GB RAM, 2TB 디스크, 리눅스 운영체제를 추천한다.
- R의 메모리 : 64bit 유닉스 환경 : 메모리 무제한
  x86 64bit 환경 : 128TB 까지 지원
  64bit 윈도우 환경 : 8TB 까지 지원

2절 R 기초 - 1

○ 학습 목표

- R GUI를 실행하여 프로그래밍을 할 수 있다.
- R GUI의 환경설정을 조정하고 편리한 기능들을 숙지한다.
- R 패키지를 이해하고 CRAN을 통해 다운로드하고 실행 할 수 있다.
- R 파일을 실행하고 배치작업을 할 수 있다.

○ 눈높이 체크

✓ R GUI와 R Studio를 통해 R 프로그래밍을 구동해 보신 적이 있습니까?

R 프로그램을 활용하기 위한 GUI는 상당히 많이 있습니다. 그 중 가장 많이 사용하는 것이 R Studio입니다. 그렇지만 R 프로그래밍의 기본인 R GUI를 통해 프로그래밍할 수 있어야 합니다.

✓ R GUI의 환경을 설정해 보신 적이 있습니까?

이번 강의에서는 R GUI를 통해 프로그래밍을 편리하게 사용하기 위한 환경설정을 직접 실습하고 프로그래밍을 위해 반드시 알아야 하는 내용들을 실습해 보도록 하겠습니다.

✓ R 패키지를 다운받고 실행할 수 있습니까?

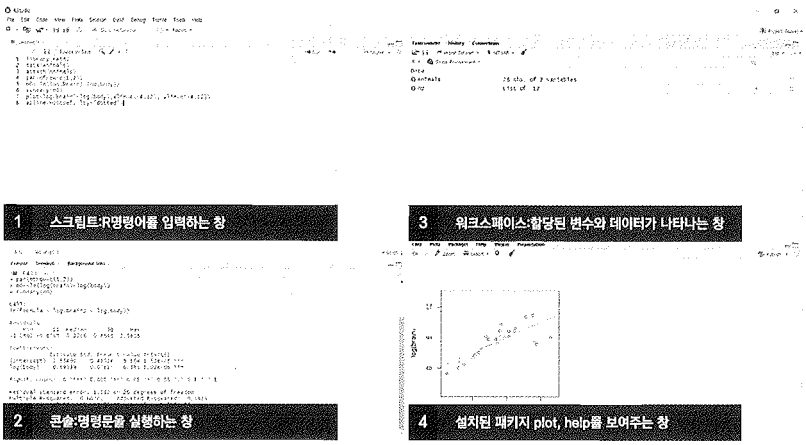
R의 가장 큰 장점은 여러 사용자가 개발한 패키지를 활용해 쉽게 데이터 분석을 할 수 있다는 것입니다. 패키지를 다운받고 실행하는 방법을 학습하고 실행함으로써 R 프로그램을 익히도록 하겠습니다.

✓ R 로 만들어진 프로그램을 실행하고 배치 작업을 경험해 보았습니까?

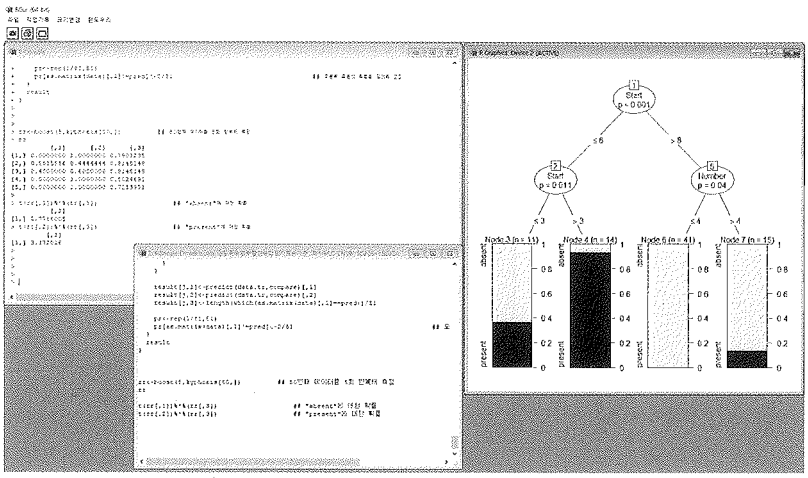
R 프로그램으로 만들어진 파일들을 실행하는 방법을 이해함으로써 반복적인 배치작업을 할 수 있는 방법을 이해할 수 있습니다.

① 통계 패키지 R

가. R Studio 구성화면



나. R GUI의 화면 구성



1) 패키지(Package)

가) 패키지란

- R 함수와 데이터 및 컴파일된 코드의 모임

(C: ₩Program Files ₩R ₩R-3.1.1 ₩library)

나) 패키지 불러들이기

① 하드디스크

- R을 설치하거나 업데이트를 통해 설치

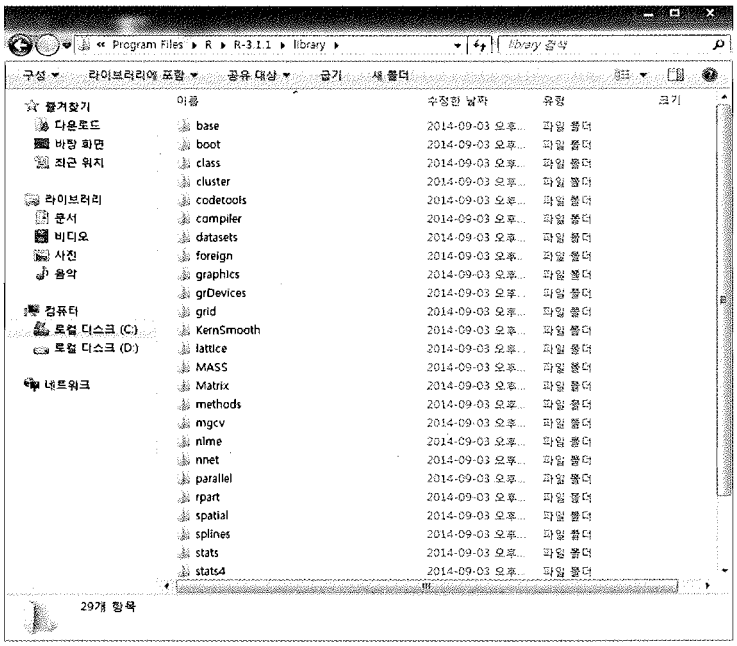
② 웹

- 2014년 CRAN 저장소에는 약 5000개의 유용한 패키지가 자동 설치

- install. packages("AID") (다운로드-C: ₩Users₩Yoon₩Documents₩R₩win-library₩3.1)
  수동설치 : install.packages("AID", "C: ₩Program Files₩R₩R-3.1.1 ₩library")

③ 패키지 도움말

- library(help=AID) : 다운로드 된 AID 패키지의 help 다큐먼트를 보여준다.
- help(package=AID) : 웹을 통해 AID 패키지의 다큐먼트를 보여준다.



2) 프로그램 파일 실행

기능	R 코드	비 고
스크립트로 프로그래밍 된 파일 실행하기	source("파일명")	오른쪽 방향키
프로그램 파일	sink(file, append, split) 함수 : R 코드 실행 결과를 특정 파일에서 출력	file : 출력할 파일명(디렉토리 포함 또는 디폴트 디렉토리)append : 파일에 결과를 덮어쓰거나 추가해서 출력 (디폴트 값(FALSE)는 덮어쓰기) split : 출력파일에만 출력하거나 콘솔창에 출력 (디폴트 값(FALSE)는 파일에만 실행 결과 출력) 예시 : sink("a_out.pdf"), sink("d:₩dataedu₩R ₩a_out.pdf")

프로그램 파일	pdf( )함수 : 그래픽 출력을 pdf 파일로 지정	예시 : pdf("a_out.pdf"), pdf("d:₩dataedu₩R ₩a_out.pdf")
	dev.off( )로 파일 닫기	

3) 배치모드 기능

가) 배치모드

- 배치모드 방식은 사용자와 인터렉션이 필요하지 않는 방식으로 매일 돌아가야 하는 시스템에서 프로세스를 자동화 할 때 유용하다.

나) 배치파일 실행 명령

- $R CMD BATCH batch.R 이라고 윈도우 도스창에서 실행한다.

다) Path 지정

- '내컴퓨터'에 오른쪽 마우스를 클릭 → 속성 → 고급시스템 설정 → 환경변수 클릭 → 변수명 path를 클릭 → R프로그램의 실행파일의 위치를 찾아서 추가 → 저장

라) 배치파일 실행

- 윈도우 창의 batch.R 실행파일이 있는 위치에서 "R CMD BATCH batch.R" 실행한다.

② 변수와 벡터 생성

- R 데이터 유형과 객체

기능	R 코드
숫자(Number)	integer, double
논리값(Logical)	True(T), False(F)
문자(Character)	"a", "abc"

2절 R 기초 - 2

○ 학습 목표

- 변수에 값을 할당하고 변수를 삭제할 수 있다.
- 기본적인 통계량을 계산할 수 있다.
- R에서의 연산자를 확인하고 우선순위를 이해할 수 있다.
- 함수의 생성 방법을 이해하고 활용할 수 있다.

○ 눈높이 체크

✓ 통계분석을 위해 변수들을 생성하고 삭제할 수 있습니까?

통계분석의 대상은 변수입니다. 통계분석을 하기 위해서는 변수를 생성하고 수정하고 삭제하는 것을 기본적으로 알고 계셔야 합니다. 이번 절에서는 R에서 변수를 다루는 방법을 학습합니다.

✓ 기본적으로 통계분석에서 활용되는 통계량과 계산 방법을 알고 계십니까?

데이터를 분석할 때 가장 기본적으로 활용되는 통계량은 평균, 중앙값, 분산, 표준편차, 공분산, 상관계수 등이 있습니다.

✓ R 프로그램의 대입연산자, 사칙연산자, 비교연산자를 이해하십니까?

일반적인 프로그래밍에서 연산을 위해서는 값을 할당하는 대입연산자와 계산에 필요한 사칙연산자, 그리고 값을 비교해서 조건문을 만들기 위한 비교연산자가 있습니다.

✓ R 프로그램에서 함수를 직접 생성하고 활용할 수 있다는 것을 이해하십니까?

R프로그램은 함수형 언어로써 프로그래머가 함수를 직접 생성해서 보다 쉽고 간단하게 데이터를 분석할 수 있도록 되어 있습니다. 함수는 입력/연산/출력으로 구성되어 있습니다.

① R 기초 중에 기초

기능	R 코드	비고
출력하기	print( ) : 출력형식을 지정할 필요 없음, 한번에 하나의 객체만 출력 cat( ) : 여러 항목을 묶어서 연결된 결과로 출력, 복합적 데이터 구조(행렬, list 등)를 출력할 수 없음	커맨드 프롬프트에 변수나 표현식을 입력 예시) print(a), cat("a","b","c")
변수에 값 할당하기 (대입 연산자)	<-, <<-, = , ->	
변수 목록보기	ls( ), ls.str( )	
변수 삭제하기	rm( )	rm(list=ls( )) : 모든 변수를 삭제할 때 사용
벡터 생성하기	c( )	벡터의 원소 중 하나라도 문자가 있으면 모든 원소의 모드는 문자 형태로 변환 됨
R 함수 정의하기	function(매개변수1,매개변수2,,,,,매개변수n) {expr 1,expr 2,.., expr m}	<expr의 특징> 지역변수 : 단순히 값을 대입하기만 하면 지역변수로 생성되고, 함수가 종료되면 지역변수는 삭제됨 조건부 실행 : if 문 반복 실행 : for문, while문, repeat 문 전역변수 : <<- 를 사용하여 전역변수로 지정할 수 있지만 추천하지 않음

② R 프로그램 소개

기능	R 코드	비고
데이터 할당	a<-1, a=1	
화면 프린트	a , print(a)	print 함수
결합	x<-c(1,2,3,4) x<-c(6.25, 3.14, 5.18) x<-c("fee","fie","fun") x<-c(x, y, z)	C함수는 문자, 숫자, 논리값, 변수를 모두 결합 가능하며 벡터와 데이터셋을 생성 가능
수열	1:5 9:-2 seq(from=0, to=20, by=2) seq(from=0, to=20, length.out=5)	콜론(:), seq 함수를 사용하여 시작값에서 최종값까지의 연속적인 숫자 생성, seq함수는 간격과 결과값의 길이를 제한 가능

반복	rep(1,time=5) rep(1:4, each=2), rep(c,each=2)	rep 함수는 숫자나 변수의 값들을 time 인자에 지정한 횟수만큼 반복
문자 붙이기	A<-paste("a","b","c", sep="-") paste(A, c("e","f")), paste(A,10, sep="")	paste 함수는 문자열을 sep 인자에 지정한 구분자로 연결시켜 줌
문자열 추출	substr("Bigdataanalysis",1,4)	substr(문자열, 시작점, 끝점) 함수는 문자열의 특정부분을 추출 가능
논리값	a<-True, a<-T, b<-False, b<-F	T 도 True 로 인식
논리 연산자	같다 == 같지 않다 != 작다, 작거나 같다 <, <= 크다, 크거나 같다 >, >=	
벡터의 원소 선택하기	V[n] : 선택하고자 하는 자리수 V[-n] : 제외하고자 하는 자리수	n은 원소의 자리수 V는 벡터의 이름

③ 벡터의 연산

연산자 우선순위	뜻	표현방법
[[[	인덱스	a[1]
$	요소 뽑아내기, 슬롯 뽑아내기	a$coef
^	지수	5^2
- +	단항 마이너스와 플러스 부호	-3, +5
:	수열 생성	1:10
%any%	특수 연산자	%/% 나눗셈 몫, %% 나눗셈 나머지, %*% 행렬의 곱
* /	곱하기, 나누기	3*5, 3/5
+ -	더하기, 빼기	3+5
== != <> <= >=	비교	3==5
!	논리 부정	!(3==5)
&	논리 "and", 단축(short-circuit) "and"	TRUE &TRUE
|	논리 "or", 단축(short-circuit) "or"	TRUE | TRUE
~	식(formula)	lm(log(brain)~log(body),data=Animals)

-> ->>	대입(왼쪽을 오른쪽으로)	3->a
=	대입(오른쪽을 왼쪽으로)	a=3
<- <<-	대입(오른쪽을 왼쪽으로)	a<-3
?	도움말	?lm

기초통계부분을 R프로그램을 통해 실행해봅시다.

④ 벡터의 기초통계

기능	R 코드	비 고
평균	mean(변수)	변수의 평균 산출
합계	sum(변수)	변수의 합계 산출
중앙값	median(변수)	변수의 중앙값 산출
로그	log(변수)	변수의 로그값 산출
표준편차	sd(변수)	변수의 표준편차 산술
분산	var(변수)	변수의 분산 산출
공분산	cov(변수1, 변수2)	변수간 공분산 산출
상관계수	cor(변수1, 변수2)	변수간 상관계수 산출
변수의 길이 값	length(변수)	변수간 길이를 값으로 출력

⑤ R 프로그래밍시 자주하는 실수

기능	R 코드	비 고
함수를 불러오고 괄호닫기	function 함수에서의 {, }, 함수의 (,)	-
윈도우 파일 경로에서 역슬래시를 두 번씩 쓰기	f: ₩dataedu ₩r ₩test.csv (f: dataedurtest.csv로 인식)	\(역슬래시, ₩)를 2번 쓰거나, /(슬래시)를 1번 써야함
<-사이를 붙여쓰기	x< -pi	Error: object "x" not found의 오류메시지
여러줄을 넘어서 식을 계속 이어갈 때	> sum<-1+2+3 > +4+5 [1] 9 >sum [1] 6	-
==대신 =를 사용하지 말 것	-	== : 비교 연산자 = : 대입 연산자

1:(n+1)대신 1:n+1로 쓰지 말 것	> n<-5; > 1:n+1; [1] 2 3 4 5 6 > 1:(n+1); [1] 1 2 3 4 5 6	-
패키지를 불러오고 library()나 require()를 수행할 것	-	-
2번 써야할 것과 1번 써야할 것을 혼돈하지 말 것	aList[[a]] vs aList[a] , && vs &, || vs | 등	-
인자의 개수를 정확히 사용할 것	mean(9,10,11)	-

MEMO

3절 입력과 출력

○ 학습 목표

- R에서의 다양한 입력(Import)과 출력(Export)방식을 이해한다.
- 다양한 구조(고정자리수, 구분자)와 형식(txt, csv등)의 외부데이터를 읽어 들일 수 있다.
- 웹(Web)에서 데이터 테이블의 데이터를 읽어 들일 수 있다.
- 복잡한 구조의 데이터 파일을 읽어 들일 수 있다.

○ 눈높이 체크

✓ **데이터 입력과 출력이 가능한 외부 데이터에 대해 알고 계신가요?**

R에서는 데이터 분석을 위해 외부에서 데이터를 불러오고 분석 결과를 외부로 출력하게 됩니다. R은 데이터베이스 뿐만 아니라 다양한 통계분석 툴의 데이터 등 다양한 데이터를 읽어오고 출력할 수 있습니다.

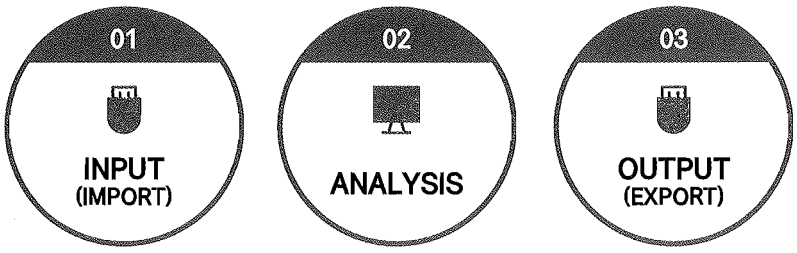
✓ **데이터의 구조와 형태가 어떤 것이 있는지 알고 계십니까?**

데이터는 구조(고정자리수, 구분자) 또는 형식(txt, csv, dat 등)으로 구분되어 있어 R로 데이터를 불러오기 위해 포맷에 따라 다양한 함수를 활용해야 합니다.

✓ **웹에서 테이블 형태의 데이터, 복잡한 구조의 데이터도 R에서 불러들일 수 있을까요?**

R에서는 웹에 있는 테이블 형태의 데이터와 데이터 구조가 아주 복잡한 데이터도 불러들여 분석할 수 있습니다. 이러한 함수들은 많은 R 사용자들이 패키지를 직접 만들어 공유할 수 있기 때문에 더욱더 발전하고 있습니다. 이것은 R의 또 다른 강력한 특성 중 하나입니다.

1 데이터 분석 과정



모든 통계 작업은 데이터에서 시작된다.

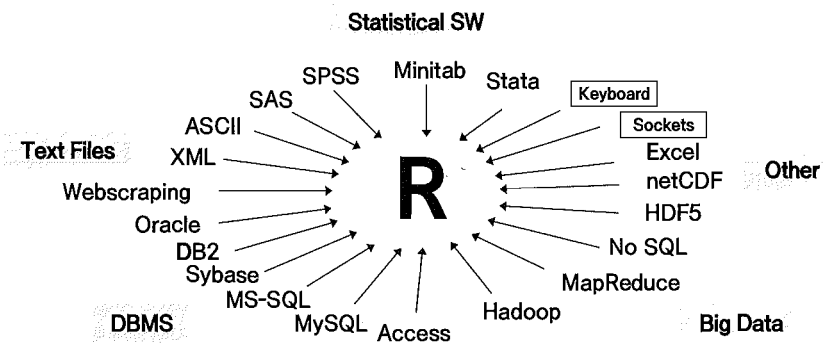
R Data Import / Export 가이드
CRM의 http://cran.r-project.org/doc/manuals/R-data.pdf

모든 통계작업은 고객에게 수치를 보고하는 것으로 끝난다.

- 분석자가 분석 목적에 맞는 적절한 분석 방법론을 선택해서 정확한 분석을 통해 얻은 결과를 통찰력을 가지고 해석함으로써 분석 과정을 마치게 된다.
- 이렇게 데이터를 분석하기 위해서는 분석자가 분석을 위해 설계된 방향으로 데이터를 정확하게 입력받는 것에서부터 시작될 수 있다.
- 그리고 입력된 데이터는 다양한 전처리 작업을 거쳐 분석이 가능한 형태로 재정리 됩니다. 우리는 이것을 데이터 핸들링이라고도 한다.
- 또한 분석된 결과를 이해하기 쉽고 잘 해석할 수 있도록 생산하는 부분을 데이터 출력이라고 할 수 있다. 출력된 결과는 보고서의 형태로 정리되어 최종 의사결정자와 고객에게 전달되게 됨으로써 통계분석 과정은 종료된다고 할 수 있다.

2 R에서의 데이터 입력과 출력

R에서 처리할 수 있는 데이터 타입은 아래와 같다.



참고

R에서 다룰 수 있는 파일 타입

Tab-delimited text, Comma-separated text, Excel file, JSON file, HTML/XML file, Database, (Other) Statistical SW's file

기능	R 코드	비 고
키보드로 데이터를 입력	1) 데이터 양이 적어 직접 입력 - c( ) : combine 함수 2) 데이터 편집기를 활용하기 : 빈데이터 프레임을 생성 → 편집기를 불러와서 편집하고 데이터 프레임에 덮어 씌우기	
출력할 내용의 자리수 정의	R의 부동소수점 표현 : 7자리로 표시 print(pi, digits=num), cat(format(pi,digits=num), " ₩n") options(digits=num)	예) 3.141593, 314.1593
파일에 출력하기	cat("출력할 내용", 변수, " ₩n", file="파일이름", append=T) sink("파일이름") … 출력할 내용 … sink()	
파일 목록 보기	list.files(), list.files(recursive=T, all.files=T)	
Cannot Open File (파일을 열 수없음) 해결하기	파일 위치 : c: ₩data ₩sample.txt R에서는 c:datasample.txt 로 인식	역슬래쉬를 슬래쉬로 바꾼다 c:/data/sample.txt 역슬래쉬를 쌍으로 표현한다 c: ₩ ₩data ₩ ₩sample.txt
고정자리수 데이터 파일 (fixed-width file) 읽기	read.fwf ("파일이름", widths=c(w1, w2,…, wn)	
테이블로 된 데이터 파일 읽기 (변수 구분자 포함)	read.table ("파일이름", sep="구분자")	(주의 1) 주소, 이름, 성 등의 텍스트를 요인으로 인식함 (해결 1) read.table("파일이름", sep="구분자", stringsASFactor=F) (주의 2) 결측치를 NA가 아닌 다른 문자열로 표현할 때 (해결 2) read.table("파일이름", sep="구분자",na.strings=".") (주의 3) 파일의 첫행을 변수명으로 인식하고자 할 때 (해결 3) read.table("파일이름", sep="구분자", header=T)

CSV 데이터 파일 읽기 (변수 구분자는 쉼표)	read.csv ("파일이름", header=T)	(주의 1) 주소, 이름, 성 등의 텍스트를 요인으로 인식함 (해결 1) read.csv("파일이름", header=T, as.is=T)
CSV 데이터 파일로 출력 (변수 구분자는 쉼표)	write.csv (행렬 또는 데이터프레임,"파일이름", row.names=F)	(주의 1) 1행을 변수명으로 자동 인식하지만 변수명이 아닐 경우 (해결 1) write.csv(dfm, "파일이름", col.names=F) (주의 2) 1열에 레코드 번호를 자동 생성 하지만 레코드 번호를 생성하지 않을 경우 (해결 2) write.csv(dfm, "파일이름", row.names=F)
웹에서 데이터 파일을 읽어 올 때 (변수 구분자는 쉼표)	read.csv ("http://www.example.com/ download/data.csv") read.table("http://www.example. com/download/data.txt")	what=numeric(0) 토큰을 숫자로 해석 what=integer(0) 토큰을 정수로 해석 what=complex(0) 토큰을 복소수로 해석 what=character(0) 토큰을 문자로 해석 what=logical(0) 토큰을 논리값으로
HTML에서 테이블 읽어 올 때	library(XML) url<-'http://www.example.com/ data/table.html' t<-readHTMLTable(url)	
복잡한 구조의 파일 (웹 테이블) 읽기	lines<-readLines("a.txt",n=num) token<-scan("a.txt", what=numeric(0)) token<-scan("a.txt",what=list (v1=character(0), v2=numeric(0)) token<-scan("a.txt", what=list (v1=character(0),v2=numeric(0), n=num, nlines=num,skip=num, na.strings=list))	

4절 데이터 구조와 데이터 프레임 - 1

○ 학습 목표

- 데이터 구조 중 벡터와 리스트 구조의 차이를 구분할 수 있다.
- 데이터 구조 중 데이터 프레임 구조를 이해한다.
- 그 밖의 R에서 활용 가능한 데이터 구조를 이해한다.
- 벡터/리스트/행렬 구조의 데이터를 다룰 수 있다.

○ 눈높이 체크

✓ 데이터 구조 중 벡터와 리스트를 알고 계신가요?

데이터 분석의 가장 기본적인 데이터 구조는 벡터입니다. 물리학에서 벡터는 힘과 방향을 나타내지만 데이터 분석에서 벡터는 여러 개의 원소를 가지는 하나의 변수라고 생각하면 됩니다.
리스트는 다른 프로그래밍 언어에서 사전(Dictionary)이나 해시(Hash), 또는 탐색표(Lookup Table)와 유사합니다만 R에서는 위치로 리스트를 인덱싱할 수 있는 장점이 있습니다.

✓ 엑셀의 시트와 SAS의 데이터셋을 다루어 보셨나요?

엑셀의 시트와 SAS의 데이터셋을 사용해본 분들은 R에서 데이터프레임의 구조와 역할을 이해하기 쉬울 것입니다. R에서는 외부데이터셋이나 대용량 데이터를 불러들일 때 데이터프레임 구조로 불러와서 데이터 분석을 실행하게 됩니다.

✓ 벡터, 스칼라, 행렬, 요인의 정의를 이해하고 계신가요?

R에서는 간단한 데이터들의 분석이 가능하도록 벡터, 스칼라, 행렬, 배열 구조의 데이터와 팩터 형의 자료들도 활용하여 분석 할 수 있습니다.

① 벡터 (Vector)

가. 벡터들은 동질적이다.

- 한 벡터의 모든 원소는 같은 자료형 또는 같은 모드(mode)를 가진다.

나. 벡터는 위치로 인덱스 된다.

- V[2]는 v벡터의 2번째 원소이다.

다. 벡터는 인덱스를 통해 여러 개의 원소로 구성된 하위 벡터를 반환할 수 있다.

- V[c(2,3)]은 v벡터의 2번째, 3번째 원소로 구성된 하위벡터이다.

라. 벡터 원소들은 이름을 가질 수 있다.

- V<-c(10,20,30); names(v)<-c("Moe", "Larry", "Curly")

  V["Larry"]

  Larry

  20

② 리스트 (List)

가. 리스트는 이질적이다.

- 여러 자료형의 원소들이 포함될 수 있다.

나. 리스트는 위치로 인덱스된다.

- L[[2]]는 L 리스트의 2번째 원소이다.

다. 리스트에서 하위 리스트를 추출할 수 있다.

- L[c(2,3)]은 L 리스트의 2번째, 3번째 원소로 이루어진 하위 리스트이다.

라. 리스트의 원소들은 이름을 가질 수 있다.

- L[["Moe"]]와 L$Moe는 둘다 "Moe"라는 이름의 원소를 지칭 한다.

③ R에서의 자료형태 (Mode)

객체	예 시	모드(Mode)
숫자	3.1415	수치형(Numeric)
숫자 벡터	c(2,3,4,5.5)	수치형(Numeric)
문자열	"Tom"	문자형(Character)
문자열 벡터	c("Tom","Yoon","Kim")	문자형(Character)
요인	factor(c("A","B","C"))	수치형(Numeric)
리스트	list("Tom","Yoon","Kim")	리스트(List)
데이터 프레임	data.frame(x=1:3,y=c("Tom","Yoon","Kim"))	리스트(List)
함수	print	함수(Function)

데이터 프레임은 데이터 구조가 행과 열로 이루어지지만 행렬이라기보다는 리스트라고 볼 수 있습니다. 데이터프레임은 엑셀의 시트와 같은 역할을 합니다. 다만 엑셀 시트에서는 열에 대한 자료형태를 각각 구분해주어야하는 번거로움이 있지만 R의 데이터 프레임에서는 각각의 열에 대해 문자형인지 수치형인지 자동적으로 구분되어 편리합니다. 데이터 프레임은 메모리상에서 구동이 된다는 것, 각 변수(열)마다 다른 자료형태로 구분되어 있다는 것을 기억하시길 바랍니다~

④ 데이터 프레임 (Dataframe)

가. 특징

- 데이터 프레임은 강력하고 유연한 구조. SAS의 데이터셋을 모방해서 만들어진다.
- 데이터 프레임의 리스트의 원소는 벡터 또는 요인이다.
- 그 벡터와 요인은 데이터 프레임의 열이다.
- 벡터와 요인들은 동일한 길이이다.
- 데이터 프레임은 표 형태의 데이터 구조이며, 각 열은 서로 다른 데이터 형식을 가질 수 있다.
- 열에는 이름이 있어야 한다.

나. 데이터 프레임의 원소에 대한 접근방법

```
b[1] ; b["empno"]
b[[i]] ; b[["empno"]]
b$empno
```

R 프로그램을 통해 실습해 보고 싶다면 동영상강의를 참고해주세요!

6 그 밖의 데이터 구조들

가. 단일값(Scalar)

- R에서는 원소가 하나인 벡터로 인식/처리

```
>pi
[1] 3.1415
>length(pi)
[1] 1
```

나. 행렬(Matrix)

- R에서는 차원을 가진 벡터로 인식

```
>a<-1:9
>dim(a) <-c(3,3)
>a
          [,1]    [,2]   [, 3]
  [1,  ]   1       4       7
  [2,  ]   2       5       8
  [3,  ]   3       6       9
```

다. 배열(Array)

- 행렬에 3차원 또는 n차원까지 확장된 형태
- 주어진 벡터에 더 많은 차원을 부여하여 배열을 생성

```
>b<-1:12
 >dim(b)<-c(2,3,2)
```

라. 요인(Factor)

- 벡터처럼 생겼지만, R에서는 벡터에 있는 유일값(Unique Value)의 정보를 얻어내는데, 이 유일값들을 요인의 수준(Level)이라고 한다.
- 요인의 두 가지 주된 사용처로 범주형 변수, 집단분류가 있다.

6 벡터, 리스트, 행렬 다루기

- 행렬(Matrix)는 R에서 차원을 가진 벡터이며, 텍스트마이닝과 소셜네트워크 분석 등에 활용한다.
- 재활용 규칙(Recycling Rule) : 길이가 서로 다른 두 벡터에 대해 연산을 할 때, R은 짧은 벡터의 처음으로 돌아가 연산이 끝날 때까지 원소들을 재활용 한다.

본래 빈칸이었던 부분이 재활용규칙에 의해 차례로 7, 8, 9가 적용됩니다. 그래서 왼쪽표와 같은 결과가 나오게 되죠. 재활용규칙은 벡터의 연산에 큰 영향을 미치므로 잘 알아두는 게 좋습니다.

a <- seq(1,6)	b <- seq(7,9)
1	7
2	8
3	9
4	
5	
6	

a+b	cbind(a,b)
8	1 7
10	2 8
12	3 9
11	4 7
13	5 8
15	6 9

기능	R 코드	비 고
벡터에 데이터 추가	v<-c(v, newItems) v[length(v)+1]<-newItem	
벡터에 데이터 삽입	append(vec, newvalues, after=n)	
요인 생성	f<-factor(v), v: 문자열 또는 정수 벡터 f<-factor(v, levels)	
여러 벡터를 합쳐 하나의 벡터와 요인으로 만들기	comb<- stack(list (v1=v1,v2=v2,v3=v3))	
벡터 내 값 조회	벡터[c(1,3,5,7)] 벡터[-c(2,4)]	벡터 내 1, 3, 5, 7번째 값 조회 벡터 내 2, 4 번째 값을 제외하고 조회
리스트	list(숫자, 문자, 함수)	list 함수의 인자로는 숫자, 문자, 함수가 포함
리스트 생성하기	L<-list(x,y,z) L<-list(valuename1 = data, valuename2 = data, valuename3 = data) L<-list(valuename1 = vec, valuename2 = vec, valuename3 = vec)	
리스트 원소 선택	L[[n]] : n번째 원소 , L[c(n1,n2,n3,⋯,nk)]:목록	
이름으로 리스트 원소 선택	L[["name"]], L$name	
리스트에서 원소 제거	L[["name"]]<-NULL	
NULL 원소를 리스트에서 제거	L[sapply(L, is.null)]<-NULL, L[L==o]<-NULL, L[is.na(L)]<-NULL	

기능	R 코드	비고
행렬	matrix(data, 행 수,열 수), a<-matrix(data,2,3) , d<-matrix(0,4,5) e<-matrix(1:20, 4,5)	data 대신 숫자를 입력하면 행렬의 값이 동일한 수치값 부여
차원	dim(행렬), dim(a)	a 행렬의 차원은 2행 3열
대각(Diagonal)	diag(행렬), diag(b)	b 행렬의 대각선의 값 반환
전치(Transpose)	t(행렬), t(a)	a 행렬의 전치행렬을 반환
역	solve(matrix)	
행렬의 곱	행렬 %*% t(행렬), a %*% t(a)	행렬의 곱
행 이름 부여	rownames(a)<-c("행이름1", "행이름2","행이름3")	행의 이름 할당
열 이름 부여	colnames(a)<-c("열이름1", "열이름2")	열의 이름 할당
행렬의 연산 +, -	f+f, f-f, f+1, f-1	행렬 간의 덧셈, 뺄셈 행렬 상수 간 덧셈, 뺄셈
행렬의 연산 *	f%*%f f*3	행렬 간의 곱 행렬 상수 간 곱
행렬에서 행, 열 선택하기	vec<-matrix[1,] vec<-matrix[, 3]	

4절 데이터 구조와 데이터 프레임 - 2

○ 학습 목표

- 데이터 프레임의 구조를 이해한다.
- 데이터 프레임에서 열과 행 데이터를 추출/제거/변경할 수 있다.
- 여러 데이터 프레임을 분할/결합/재생산할 수 있다.
- 모든 데이터 구조를 변경하여 활용할 수 있다.

○ 눈높이 체크

✓ 데이터 프레임의 구조와 정의를 이해 하시나요?

데이터 프레임은 다변량 데이터 분석을 위해 가장 많이 활용되는 데이터 구조입니다. 여러 변수들을 활용하여 소기의 목적에 맞는 데이터 분석을 하기 위해서는 데이터 프레임의 구조와 정의를 잘 이해해야 합니다.

✓ 데이터 셋에서 특정 변수 또는 특정 행들을 추출/제거/수정 하고 분석해 보셨나요?

데이터 프레임의 행과 열을 추출/제거/수정함으로써 데이터를 분석할 수 있는 최적의 상태로 자료를 유지해야 하며 이러한 과정은 데이터 전처리와 데이터 클렌징에서 가장 많이 활용됩니다.

✓ 여러 데이터 셋들을 결합/분할/추출하여 통계분석을 해본 경험이 있으신가요?

분석하고자 하는 대상 데이터를 여러 데이터셋에서 결합/분할/추출하여 분석을 위한 Training set과 Validation set 등을 분할하여 최적의 결과를 얻을 수 있는 분석적 구조를 마련하는 것이 중요합니다.

• 데이터에서 각각의 변수에 해당하는 열들의 모임으로 R에서 활용하는 코드들은 아래와 같다.

기능	R 코드	비고
데이터 프레임	data.frame(벡터,벡터,벡터)	벡터들로 데이터셋 생성
레코드 생성	new<-data. frame(a=1,b=2,c=3,d="a")	레코드 생성시 숫자, 문자를 함께 사용 가능
열 데이터(변수)로 데이터 프레임 만들기	dfm<-data. frame(v1,v2,v3,f1,f2) dfm<-as.data.frame(list. of.vectors)	
데이터셋 행결합	rbind(dfrm1, dfrm2) newdata<- rbind(newdata,new)	두 데이터 프레임을 행으로 결합
데이터셋 열결합	cbind(dfrm1, dfrm2) cbind(newdata,newcol) #newcol=1:150	두 데이터 프레임을 열로 결합
데이터 프레임 할당	N=1,000,000 dtfm<-data. frame(dosage=numeric(N), lab=character(N), respones=numeric(N))	
데이터 프레임 조회1	dfrm[dfrm$gender="m"]	데이터셋 내 성별이 남성만 조회
데이터 프레임 조회2	dfrm[dfrm$변수1>4 & dfrm$변수2>5, c(변수3, 변수4)]	데이터셋의 변수1과 변수2의 조건에 만족하는 레코드의 변수3과 변수4만을 조회
데이터 프레임 조회3	dfrm[grep("문자", dfrm$변수1, gnore.case=T), c("변수2, 변수3")]	데이터셋의 변수1 내 "문자"가 들어 있는 케이스들의 변수2, 변수3 값을 조회
데이터셋 조회	subset(dfrm, select=변수, subset=변수>조건)	데이터셋의 특정변수의 값이 조건에 맞는 변수셋 조회, subset은 벡터와 리스트에서도 선택 가능
데이터 선택	lst1[[2]], lst1[2] , lst1[2,], lst1[,2] lst1[["name"]], lst1$name, lst1[c("name1", "name2",… ."name k")]	
데이터 병합	merge(df1,df2, by= "df1과 df2의 공통 열 이름")	공통변수로 데이터셋 병합
열 네임 조회	colnames(변수)	변수의 속성들을 조회

행, 열 선택	subset(dfm, select=열이름) subset(dfm, select=c(열이름1, 열이름2, …, 열이름n)) subset(dfm, select=열이름, subset=(조건))	열이름에 " " 표시 안함, 조건에 맞는 행의 열 자료만 선택
이름으로 열 제거	subset(dfm, select=-"열이름")	
열 이름 바꾸기	colnames(dfm) <- newnames	
NA 있는 행 삭제	NA_cleaning <- na.omit(dfm)	
데이터 프레임 두 개 합치기	열 (cbind_dfm <- cbind(dfm1,dfm2) 행 (rbind_dfm <- rbind(dfm1,dfm2)	(유의사항1) cbind는 행의 개수가 동일해야 함 -recycling rule (유의사항2) rbind는 열의 개수와 열의 이름이 동일해야 함
두 개의 데이터 프레임을 동일한 변수 기준으로 합치기	merge(dfm1,dfm2, by="T_name") merge(dfm1,dfm2, by="T_name",all=T)	

② 자료형 데이터 구조 변환

기능	R 코드
데이터 프레임의 내용에 쉽게 접근하기	with(dfm, expr) attach(dfm) detach(dfm)
자료형 변환하기	as.character() as.complex() as.numeric() 또는 as.double() as.integer() as.logical()
데이터 구조 변환하기	as.data.frame() as.list() as.matrix() as.vector()

③ 데이터 구조 변경

벡터 → 리스트	as.list(vec)	행렬 → 벡터	as.vector(mat)
벡터 → 행렬	1열짜리 행렬 : cbind(vec) 또는 as.matrix(vec) 1행짜리 행렬 : rbind(vec) n x m 행렬 : matrix(vec,n,m)	행렬 → 리스트	as.list(mat)
벡터 → 데이터 프레임	1열짜리 데이터프레임 : as.data.frame(vec) 1행짜리 데이터프레임 : as.data.frame(rbind(vec))	행렬 → 데이터 프레임	as.data.frame(mat)
리스트 → 벡터	unlist(lst)	데이터 프레임 → 벡터	1열짜리 데이터 프레임 : dfm[[1]] or dfm[,1] 1행짜리 데이터 프레임 : dfm[1,]
리스트 → 행렬	1열짜리 행렬 : as.matrix(lst) 1행짜리 행렬 : as.matrix(rbind(lst)) n x m 행렬 : matrix(lst,n,m)	데이터 프레임 → 리스트	as.list(dfm)
리스트 → 데이터 프레임	목록 원소들이 데이터의 열이면 : as.data.frame(lst) 리스트 원소들이 데이터의 행이면 : rbind(obs[[1]],obs[[2]])	데이터 프레임 → 행렬	as.matrix(dfm)

④ 벡터의 기본 연산

기능	R 코드	비고
벡터 연산	벡터1 + 벡터2 벡터2 - 벡터2 벡터1 * 벡터2 벡터1 ^ 벡터2	덧셈 연산 뺄셈 연산 곱셈 연산 승수 연산
함수 적용	sapply(변수, 연산함수) sapply(a,log)	연산 및 적용 함수를 통해 변수에 적용
파일 저장1	write.csv(변수이름, "파일이름.csv") write.csv(a,"test.csv")	파일로 저장
파일 저장2	save(변수이름, file="파일이름.Rdata") save(a, file="a.Rdata")	R파일로 저장
파일 읽기	read.csv("파일이름.csv") read.csv("a.csv")	파일 읽기

파일 불러오기	load("파일.R") load("a.R") source("a.R")	R파일 불러오기
데이터 삭제	rm(변수) rm(list=ls(all=TRUE))	변수를 메모리에서 삭제 모든 변수를 메모리에서 삭제

⑤ 그 외에 간단한 함수

기능	R 코드	비 고
데이터 불러오기	data( ) data(데이터셋)	R에 내장된 데이터셋 리스트를 보여줌. 데이터셋을 불러들임
데이터셋 요약	summary(데이터셋)	데이터셋 변수 내용을 요약
데이터셋 조회	head(데이터셋)	6개 레코드까지 데이터 조회
패키지 설치	install.packages("패키지 명")	패키지를 설치
패키지 불러오기	library("패키지 명")	패키지를 불러들임
작업 종료	q( )	작업을 종료
워킹디렉토리 지정	setwd("~/")	R 데이터와 파일 등을 로드하거나 저장할 때 워킹 디렉토리를 지정

5절 데이터 변형

주요 코드 ①

기능	R 코드	비 고
요인으로 집단 정의	v<-c(24,23,52,46,75,25) w<-c(87,86,92,84,77,68) f<-factor(c("A","A","B","B","C","A"))	
벡터를 여러 집단으로 분할 (벡터의 길이만 같으면 됨)	groups<-split(v,f) groups<-split(w,f) groups<-unstack(data.frame(v,f))	두 함수 모두 벡터로 된 리스트를 반환
데이터 프레임을 여러집단으로 분할	MASS 패키지, Cars93 데이터셋 활용 library(MASS) sp<-split(Cars93$MPG.city, Cars93$Origin) median(sp[[1]])	
리스트의 각 원소에 함수 적용	lapply(결과를 리스트 형태로 반환) list<-lapply(l,func) sapply(결과를 벡터 또는 행렬로 반환) vec<-sapply(l,func)	
행렬에 함수 적용	m<-apply(mat, 1, func) m<-apply(mat, 2, func)	
데이터프레임에 함수 적용	dfm<-lapply(dfm, func) dfm<-sapply(dfm, func) dfm<-apply(dfm, func) : 데이터프레임이 동질적인 경우만(모두 문자, 숫자) 활용가능 데이터프레임을 행렬로 변환 후 함수 적용	
대용량 데이터의 함수적용	<sapply를 통한 간단한 R코딩> cors<- sapply(dfm,cor,y=targetVariabele) mask<-(rank(-abs(cors))<=10) best.pred<-dfm[,mask] lm(targetVariabele~bes.pred)	많은 변수가 있는 데이터에서의 다중회귀분석 1. 데이터 프레임에서 타겟 변수를 정한다. 2. 타겟변수와 상관계수를 구한다. 3. 상관계수가 높은 상위 10개의 변수를 입력변수로 선정 4. 타겟변수와 입력변수로 다중회귀분석을 실시한다.

집단별 함수 적용	tapply(vec, factor, func)	데이터가 집단(factors)에 속해 있을 때 합계/평균 구하기
행집단 함수 적용	by(drm,factor,func) 요인별 선형회귀선 구하기 model(dfm, factor, function(df) lm(종속변수~독립변수1+독립변수 2+…+독립변수k, data=df))	
병력 벡터, 리스트들 함수 적용	mapply (factor, vec1, vec2, vec3, …, vec k) mapply (factor, list1, list2, list3, …,list k)	

② 문자열, 날짜 다루기

기능	R코드	비고
문자열 길이	nchar("단어")	단어나 문장 또는 벡터내 원소의 문자열 길이를 반환 [주의] length(vec) 문자열의 길이가 아닌 벡터의 길이를 반환
문자열 연결	paste("단어1", "단어2", sep="-")paste("the pi is approximately", pi) paste(vec, "loves me", collapse=", and")	
하위문자열 추출	substr("statistics",1,4)	문자열의 1자리에서 4자리까지 추출
구분자로 문자열 추출	strsplit(문자열, 구분자)	
하위 문자열 대체	sub(old, new, string) gsub(old, new, string)	
쌍별 조합	mat<-outer(문자열1, 문자열2, paste, sep="")	
날짜 변환1	Sys.Date() as.Date()	현재 날짜를 반환 날짜 객체로 반환
날짜 변환2	format(Sys.date(), format=%m/%d/%y)	
날짜 조회	format(Sys.Date(), '%a') format(Sys.Date(), '%b') format(Sys.Date(), '%B') format(Sys.Date(), '%d') format(Sys.Date(), '%m') format(Sys.Date(), '%y') format(Sys.Date(), '%Y')	요일 조회 축약된 월이름 조회 전체 월이름 조회 두자리 숫자의 일 조회 두자리 숫자의 월 조회 두자리 숫자의 연도 조회 네자리 숫자의 연도 조회

날짜 일부 추출	d<-as.Date("2014-12-25") p<-as.POSIXlt(d) p$yday start<-as.Date("2014-12-01") end<-as.Date("2014-12-25") seq(from=start, to=end, by=1)

MEMO

3과목 | 2장

R 프로그래밍 기초

✓16회 기출

01 다음 중 R의 데이터 구조 중 벡터에 대한 설명으로 적절한 것은?

① 벡터는 행과 열을 갖는 m x n 형태의 직사각형에 데이터를 나열한 데이터 구조이다.

② 벡터는 하나의 스칼라 값 또는 하나 이상의 스칼라 원소들을 갖는 단순한 형태의 집합이다.

③ 벡터는 행렬과 유사한 2차원 목록 데이터 구조이다.

④ 벡터는 숫자로만 구성되어야 한다.

✓20회 기출

02 다음 중 아래의 프로그램을 통해 생성된 벡터 xy에 대한 설명으로 옳지 않은 것은?

```
> x<-c(1:4)
> y<-c("apple","banana","orange")
> xy<-c(x,y)
```

① xy는 문자형 벡터이다.

② xy의 길이는 7이다.

③ xy[1]+xy[2]의 결과는 3이다.

④ xy[5:7]은 y와 동일하다.

✓16회 기출

03 다음 중 연속형 변수의 경우 4분위수, 최소값, 최대값, 중앙값, 평균 등을 출력하고 범주형 변수의 경우 각 범주에 대한 빈도수를 출력하여 데이터의 분포를 파악할 수 있게 하는 함수로 적절한 것은?

① summary 함수

② ddply 함수

③ cast 함수

④ aggregate 함수

✓7회 기출

04 다음 중 나머지 세 개의 명령과 결과가 다른 것은?

① z=c(1:3, NA)
is.na(z)

② z<-c(1:3, NA)
is.na(z)

③ z= c(1:3, NA)
z==NA

④ c(1,1,1,2) ==2

✓7회 기출

05 아래의 R 프로그래밍을 통해 객체 a 에 할당되는 모드가 다른 것을 고르시오.

① a<-c("Tom", "Yoon", "Kim")

② a<-c(pi, "pi", 3.14)

③ a<-c(3.14, pi, TRUE)

④ a<-c("A","B","A","A","B")

✓11회 기출

06 다음 중 결과가 다른 R코드는?

① a<-seq(1,10,1)

② b<-c(1,10)

③ c<-1:10

④ d<-seq(10,100,10)/10

✓11회 기출

07 다음 중 아래의 R코드를 수행한 결과에 대한 설명으로 옳은 것은?

```
> c(2, 4, 6, 8) + c(1, 3, 5, 7, 9)
```

① 경고 메시지와 함께 결과가 출력된다.

② 4개의 숫자로 이루어진 벡터가 출력된다.

③ 9개의 숫자로 이루어진 벡터가 출력된다.

④ 에러 메시지가 출력되고, 명령 수행이 중단된다.

✓18회 기출

08 R의 데이터 구조와 저장형식에 관한 설명으로 가장 부적절한 것은?

① as.numeric 함수에 논리형 벡터를 입력하면 TRUE에 대응하는 원소는 1, FALSE에 대응하는 원소는 0인 숫자형 벡터로 변형된다.

② 숫자형 행렬에서 원소 중 하나를 문자형으로 변경하게 되면 해당 행렬의 모든 원소가 문자형으로 변경된다.

③ 데이터 프레임은 각 열 별로 서로 다른 데이터 타입을 가질 수 있다.

④ 행렬을 as.vector 함수에 입력하면 행 방향으로 1행부터 차례로 원소를 나열하는 벡터가 생성된다.

✓18회 기출

09 R의 데이터 구조 중 2차원 목록 데이터 구조이면서 각 열이 서로 다른 데이터 타입을 가질 수 있는 데이터 구조로 적절한 것은?

① 벡터

② 행렬

③ 배열

④ 데이터프레임

✓10회 기출

10 아래의 R코드가 의미하는 것은?

```
> mean(x, na.rm=T)
```

① 이상값을 제외한 X의 평균

② 결측값을 제외한 X의 평균

③ 이상값을 포함한 X의 평균

④ 결측값을 포함한 X의 평균

✓18회 기출

11 R에서 제공하는 데이터 가공, 처리를 위한 패키지의 설명으로 가장 부적절한 것은?

① data.table 패키지는 데이터 프레임 처리함수 인 ddply함수를 제공한다.

② reshape 패키지는 melt와 cast를 이용하여 데이터를 재구성할 수 있다.

③ sqldf 패키지는 R에서 표준 SQL 명령을 실행하고 결과를 가져올 수 있다.

④ plyr 패키지는 데이터의 분리, 결합 등 필수적인 데이터 처리 기능을 제공한다.

✓10회 기출

12 아래 R코드를 수행한 결과로 적절한 것은?

```
> "+"(2,3)
```

① 에러 메시지가 출력된다.

② 경고 메시지가 출력된다.

③ 숫자 5가 출력된다.

④ 두 개의 원소로 이루어진 벡터가 출력된다.

✓18회 기출

13 R에서 y=c(1,2,3,NA)일 때 3*y의 실행 결과는?

① 에러가 발생하고 결과가 출력되지 않는다.

② 3 6 9 0

③ 3 6 9 3

④ 3 6 9 NA

✓9회 기출

14 R에서 결측값을 가리키는 것으로 가장 적절한 것은?

① Inf ② NaN ③ NA ④ dim

✓15회 기출

15 Carseats 데이터 프레임은 400개 상점에서 판매 중인 유아용 카시트의 재료이고, Sales 변수는 해당 상점에서 판매된 카시트의 수를 나타낸다. 다음 중 R 패키지에서 Sales 변수의 표준편차를 계산하기 위한 식으로 가장 부적절한 것은?

① stdev(Carseats$Sales)

② sd(Carseats$Sales)

③ sqrt(var(Carseats$Sales))

④ var(Carseats$Sales)^(1/2)

✓20회 기출

16 다음 중 아래 R 코드의 결과로 적절한 것은?

```
> s<-c("Monday", "Tuesday", "Wednesday")
> substr(s,1,2)
```

① "Mo", "Tu", "We"

② "Monday" "Tuesday"

③ "Mo" "Tu"

④ "Monday"

17 아래 그림과 같이 두개의 데이터 프레임 dfm1, dfm2 를 T_name 이라는 변수로 결합하고자 할 때, 사용되는 함수는 어느 것인가?

T_name	x	y
T1	1.4	3.2
T2	1.8	3.4
T3	1.5	3.9
T4	1.4	3.2
T5	1.6	3.4
T6	1.5	3.9

+

T_name	z
T1	5.7
T3	5.8
T5	6.9

=

T_name	x	y	z
T1	1.4	3.2	5.7
T3	1.5	3.9	5.8
T5	1.6	3.4	6.9

① cbind(dfm1, dfm2, by="T_name")

② rbind(dfm1, dfm2, by="T_name")

③ merge(dfm1,dfm2, by="T_name")

④ subset(dfm1,dfm2,by ="T_name")

✓12회 기출

18 아래 프로그램의 실행 결과로 다음 중 적절한 것은 무엇인가?

```
calculate<-function(a) {
  y=1
  for(i in 1:a) {
    y=y*i
  }
  print(y)
}


calculate(4)
```

① 24 ② 20 ③ 12 ④ 6

✓19회 기출

19 아래 프로그램을 통해 생성된 xy에 대한 설명으로 부적절한 것은?

```
> x<-c(1:5)
> y<-seq(10,50,10)
> xy<-rbind(x,y)
```

① 2X5 행렬이다.

② xy[1,]은 x와 동일하다.

③ xy[,1]은 y와 동일하다.

④ Matrix 타입의 개체이다.

✓22회 기출

20 R에서 matrix 명령어를 활용하여 벡터를 행렬로 아래와 같이 변환하였다고 할 때 생성된 mx의 결과로 옳은 것은 ?

```
mx = matrix(c(1,2,3,4,5,6), ncol=2, byrow=T)
```

①

	[,1]	[,2]
[1,]	1	2
[2,]	3	4
[3,]	5	6

②

	[,1]	[,2]
[1,]	1	4
[2,]	2	5
[3,]	3	6

③

	[,1]	[,2]	[,3]
[1,]	1	2	3
[2,]	4	5	6

④

	[,1]	[,2]	[,3]
[1,]	1	3	5
[2,]	3	4	6

✓20회 기출

21 R에서 데이터 타입이 같지 않은 객체들을 하나의 객체로 묶어놓을 수 있는 자료구조는 어떤 것인가?

① 행렬(Matrix)

② 배열(Array)

③ 리스트(List)

④ 문자열(String)

✓23회 기출

22 다음 중 2019/08/23을 "2019-08-23"으로 나타내는 코드로 올바른 것은?

① as.Date('08/23/2019', '%m/%d/%Y')

② as.Date('08/23/2019', '%m/%D/%Y')

③ as.Date('08/23/2019', '%M/%d/%Y')

④ as.Date('08/23/2019', '%M/%D/%Y')

✓20회 기출

23 아래 R 코드로 생성되는 행렬 A에서 일부 원소를 추출하기 위한 코드 중 나머지 보기와 결과가 다른 것은?

```
A <- cbind( c(1,2,3), c(4,5,6), c(7,8,9) )
colnames(A) <- c("A", "B", "C")
rownames(A) <- c("r1", "r2", "r3")
```

① A[,"A"]

② A[-c(2,3),]

③ A[,1]

④ A[,-(2:3)]

✓19회 기출

24 R에서 새로운 패키지를 설치 및 사용하고자 할 때 명령어와 순서로 적절한 것은 ?

① install.packages("패키지명") → library(패키지명)

② install.packages(패키지명) → library("패키지명")

③ library("패키지명") → install.packages("패키지명")

④ ibrary(패키지명) → install.packages("패키지명")

✓10회 기출

25 아래 R 코드의 출력 결과는?

```
> f <- function(x,a) return((x-a)^2)
> f(1:2,3)
```

( )

✓8회 기출

26 R에서 다음의 명령을 수행했을 때 출력되는 결과는?

```
x<-c(1,2,3,NA)
mean(x)
```

( )

✓7회 기출

27 R에서 다음의 명령을 수행했을 때 출력되는 결과는?

```
x<-1:100
sum(x>50)
```

( )

28 A반과 B반 학생들이 동일한 과목을 들었다고 하자. A반과 B반 학생 모두를 대상으로 과목 별 성적의 평균을 구하려고 할 때, A반 학생 데이터와 B반 학생 데이터를 class 라는 변수를 기준으로 합치려고 한다. R로 프로그래밍 하시오.

( )

29 아래의 표와 같이 여러 학과 학생들의 과목별 성적을 데이터 프레임으로 구성하였다. 데이터 프레임명은 test 라고 할 때, 경영학과 학생들의 데이터만 조회하고자 한다. R로 프로그래밍 하시오.

학과	학년	성별	이름	실용컴퓨터	영어회화	한문	총점
경영학과	1	여	김지영	85	75	86	246
경영학과	1	여	이소연	75	65	78	218
경영학과	1	남	이진혁	96	77	67	240
데이터정보학과	3	남	김영수	45	78	56	179
데이터정보학과	1	남	김민수	86	87	84	257
데이터정보학과	1	여	박미혜	100	92	96	288
데이터정보학과	1	남	최성호	87	95	92	274
영문학과	4	여	김동수	68	75	78	221
영문학과	2	남	이민지	99	86	86	271

( )

30 SQL을 활용하거나 SAS에서 PROC SQL로 작업하던 사용자들에게 R 프로그램에서 지원해 주는 패키지는 무엇인가?

( )

3과목 | R 프로그래밍 기초

정답 및 해설

01	②	11	①	21	③
02	③	12	③	22	①
03	①	13	④	23	②
04	③	14	③	24	①
05	③	15	①	25	4 1
06	②	16	①	26	NA
07	①	17	③	27	50
08	④	18	①	28	merge(A,B,by="class")
09	④	19	③	29	subset(test, 학과=="경영학과")
10	②	20	①	30	sqldf()

01. 벡터는 하나의 스칼라 값 또는 하나 이상의 스칼라 원소들을 갖는 단순한 형태의 집합으로 한 벡터의 모든 원소는 같은 자료형(숫자 또는 문자)로 구성된다. 벡터는 행렬 구조로 나타나지 않는다. (**정답 : ②**)

02. xy는 문자형 벡터로 문자형은 서로 연산을 할 수 없으므로 출력결과에는 에러가 나타난다. (**정답 : ③**)

03. R에서의 summary 함수는 수치형 변수의 경우 최대값, 최소값, 평균, 1사분위수, 2사분위수(중앙값), 3사분위수를 출력하며, 명목형 변수의 경우 명목값, 데이터의 개수를 출력하는 함수이다. (**정답 : ①**)

04. ①, ②, ④의 결과는 모두 FALSE FALSE FALSE TRUE 이지만, ③의 경우에는 NA NA NA NA가 나타난다. (**정답 : ③**)

05. ①, ②, ④의 결과는 모두 character 이지만, ③의 경우에는 numeric이다. (**정답 : ③**)

06. ①, ③, ④의 결과는 모두 1 2 3 4 5 6 7 8 9 10이지만, ②의 경우에는 1 10이다. (**정답 : ②**)

07. 아래의 R 코드를 실행시키면 '두 객체의 길이가 서로 배수관계에 있지 않습니다'라는 경고 메시지가 뜨고 결과도 출력된다. (**정답 : ①**)

08. 행렬을 as.vector 함수에 입력하면 열방향으로 1열부터 차례로 원소를 나열하는 벡터가 생성된다. (**정답 : ④**)

09. 데이터프레임은 표 형태의 데이터 구조이며, 각 열은 서로 다른 데이터 형식을 가질 수 있다. (**정답 : ④**)

10. 해당 R 코드 중 na.rm은 결측치를 제외하느냐에 대한 물음이며, T는 TRUE로서 결측치를 제외하겠다는 의미이다. (**정답 : ②**)

11. data.table 패키지는 큰 데이터를 탐색, 연산, 병합하는데 아주 유용하다. ddply는 plyr패키지에서 지원한다. (**정답 : ①**)

12. 아래의 코드를 실행하면 숫자 5가 출력된다. (**정답 : ③**)

13. 각 값에 3이 곱해져 3 6 9 NA 가 출력된다. (**정답 : ④**)

14. Inf는 무한대, NaN은 Not a Number, dim은 행렬의 차원을 나타낸다. (**정답 : ③**)

15. R에서 표준편차를 계산하기 위해 사용하는 함수가 아닌 것은 stdev()함수이다. (**정답 : ①**)

16. 아래의 코드를 실행하면 "Mo", "Tu", "We"가 나타난다. (**정답 : ①**)

17. 두 개의 테이블을 하나로 변경할 때 merge 함수를 사용한다. (**정답 : ③**)

18. Calculate(4)를 실행 했을 때, (1). y=1, i=1 → y=1, (2). y=1, i=2 → y=2, (3). y=2, i=3 → y=6, (4). y=6, i=4 → y=24이므로 24가 출력된다. (**정답 : ①**)

19. y는 10 20 30 40 50이라는 값이 출력되지만 xy[,1]을 실행하면 x와 y 각각 1 10이라는 값이 출력된다. (**정답 : ③**)

20. 벡터를 행을 기준으로 2열로 매트릭스를 생성한다. (**정답 : ①**)

21. 리스트는 타입이 같지 않은 객체들을 하나의 객체로 묶어놓을 수 있는 자료구조이다. (**정답 : ③**)

22. 2019를 전체 표현하기 위해서는 %Y, 08과 23을 표현하기 위해서는 각각 %m, %d의 format를 가져야한다. (**정답 : ①**)

23. ②의 경우는 $\begin{matrix} A & B & C \\ 1 & 4 & 7 \end{matrix}$ 출력되고 나머지는 $\begin{matrix} r1 & r2 & r3 \\ 1 & 2 & 3 \end{matrix}$ 으로 출력된다. (**정답 : ②**)

24. install.packages("패키지명")로 패키지를 설치하고 library(패키지명)로 패키지를 불러와 사용할 수 있다. (**정답 : ①**)

25. **정답 : 4 1**

26. **정답 : NA**

27. **정답 : 50**

28. **정답 : merge(A, B, by="class")**

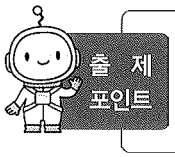
29. **정답 : subset(test, 학과=="경영학과")**

30. **정답 : sqldf()**

PART 03
데이터 분석
3장

데이터 마트

15 DAY



최소한 2문제 이상 출제됩니다. 요약변수와 파생변수에 대한 내용이 보기를 통해 나올 수 있고, Reshape이나 다른 R패키지를 활용하여 데이터 마트를 어떻게 구성할 수 있는지 묻기도 합니다. 이번 장에서 등장하는 용어의 정의와 R프로그램에서 사용할 여러 함수에 관해 알아두도록 합시다.

○ 학습 목표

- 데이터 마트를 구성하는 요약변수와 파생변수를 구분할 수 있다.
- reshape 패키지를 활용하여 데이터 마트를 생성할 수 있다.
- sqldf 패키지와 plyr 패키지를 활용하여 데이터를 핸들링할 수 있다.
- data.table 패키지를 이해하고 활용할 수 있다.

○ 눈높이 체크

✔ **요약변수와 파생변수에 대해 알고 계신가요?**

데이터 마트를 구성할 때 가장 중요한 부분 중 하나가 요약변수와 파생변수를 생성하는 부분입니다. 모형을 개발할 때 문제를 가장 잘 해석할 수 있는 변수를 찾는 것은 모형 개발에서 가장 중요한 핵심단계입니다. 그래서 데이터를 특정 기준에 따라 사칙연산을 통해 만들어 낸 변수가 요약변수이고 사용자의 노하우를 기반으로 새롭게 만들어 낸 변수가 파생변수입니다.

✔ **R프로그램에서 reshape 패키지를 들어 보셨나요?**

reshape 패키지는 데이터 마트를 생성할 수 있도록 데이터를 녹이고(melt) 다시 형상화(cast)할 수 있는 R 패키지로, 분석용 마트 설계에서 잘 활용됩니다.

✔ **R프로그램에서 SQL은 어떻게 활용할 수 있을까요?**

SAS에서 SQL을 활용할 수 있듯이 R 프로그램에서도 SQL을 사용하기 위해 sqldf라는 패키지를 통해 SQL을 활용할 수 있습니다. sqldf 함수를 사용하면 모든 SQL 문장을 거의 똑같은 형식으로 사용할 수 있게 됩니다.

✔ **data.table 패키지를 들어보셨나요?**

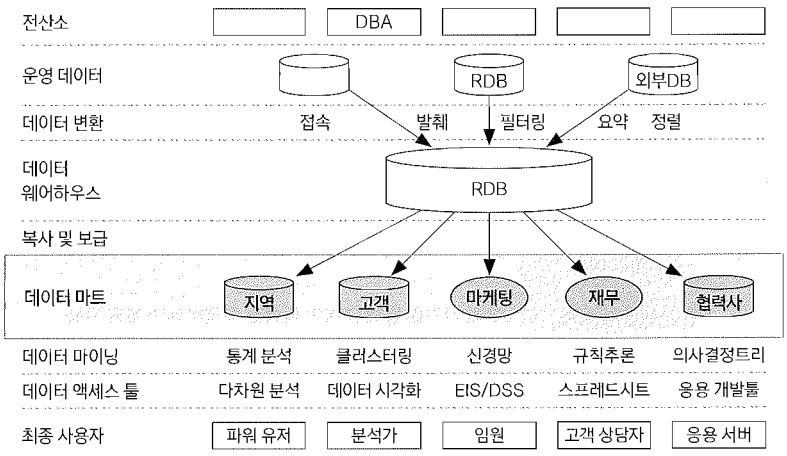
data.table 패키지는 dataframe 과 같은 구조를 가지고 있으나 key를 활용해서 훨씬 빠른 연산이 가능하게 만든 패키지입니다.

1절 데이터 변경 및 요약

가. 데이터 마트

- **데이터 웨어하우스와 사용자 사이의 중간층에 위치**한 것으로, 하나의 주제 또는 하나의 부서 중심의 데이터 웨어하우스라고 할 수 있다.
- 데이터 마트 내 대부분의 데이터는 데이터 웨어하우스로부터 복제되지만, 자체적으로 수집될 수도 있으며, 관계형 데이터 베이스나 다차원 데이터 베이스를 이용하여 구축 한다.

<출처 : 컴퓨터 인터넷IT용어대사전>



- CRM(Customer Relationship Management) 관련 업무 중에서 핵심 - 고객 데이터 마트 구축
- 동일한 데이터 셋을 활용할 경우, 최신 분석기법들을 이용하면 분석가의 역량에서는 분석 효과가 크게 차이가 나지 않기 때문에 데이터 마트를 어떻게 구축하느냐에 따라 분석 효과는 크게 차이 난다.

나. 요약변수

- 수집된 정보를 **분석에 맞게 종합한 변수**이다.
- 데이터 마트에서 가장 기본적인 변수로 **총 구매 금액, 금액, 횟수, 구매여부 등 데이터 분석을 위해 만들어지는 변수**이다.

1 R reshape를 이용한 데이터 마트 개발

데이터 마트를 만들 때 가장 중요한 데이터들은 데이터 웨어하우스로부터 받아오는 데이터입니다. 받아온 데이터를 처리과정을 통해 분석에 적절하게 활용할 수 있는 자료로 변환을 해야 합니다. 이렇게 만들어진 변수는 요약변수와 파생변수로 나뉩니다. 요약변수와 파생변수의 정의와 예시에 대해 알아야 합니다. 요약변수와 파생변수에 대한 내용을 섞어 문제가 출제되는 경향이 높기 때문입니다. 두 변수가 어떠한 차이가 있는지 체크합시다.

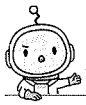
요약변수를 잘 만든다면 분석의 중요한 변수로 활용이 가능합니다. 요약변수를 잘 만드는 것이 가장 중요하다고 할 수 있죠.

- 많은 모델에 공통으로 사용될 수 있어 **재활용성이 높다.**
- 합계, 횟수와 같이 간단한 구조이므로 자동화하여 상황에 맞게 또는 일반적인 자동화 프로그램으로 구축 가능하다.
- 요약변수의 단점은 얼마 이상이면 구매하더라도 기준값의 의미 해석이 애매할 수 있다. 이러한 경우, 연속형 변수를 그룹핑해 사용하는 것이 좋다.

예시를 공부할 때 요약변수의 예시를 완벽히 숙지한 후 나머지는 파생변수라고 생각하면 더 쉬울 것입니다.

예시

기간별 구매 금액, 횟수 여부	고객의 구매 패턴을 볼 수 있는 변수
위클리 쇼퍼	구매 시기를 통해 고객의 특성을 추정하는데 활용 가능
상품별 구매 금액, 회수 여부	고객의 라이프 스테이지와 라이프 스타일 등을 이해하는데 크게 도움이 됨
상품별 구매 순서	고객에 대한 이해와 해석력을 높일 수 있음
유통 채널별 구매 금액	온라인과 오프라인 사용 고객에게 모두 사용하도록 유도하는데 활용
단어 빈도	텍스트 자료에서 단어들의 출현 빈도를 데이터화하여 사용
초기 행동변수	고객 가입 또는 첫 거래 초기 1개월 간 거래 패턴에 대한 변수로 1년후에 어떤 행동을 보일지를 평가하는 지표로 활용
트렌드 변수	추이값을 나타내는 변수
결측값과 이상값 처리	결측값과 이상값은 무리해서 처리하려고 하면 시간과 위험이 커질 수 있으므로 데이터의 내용을 파악하여 처리해야 함
연속형 변수의 구간화	분석후 적용 단계를 고려한 데이터 분석을 위해 연령이나 비용 등 연속형 변수를 구간화 하는 것이 필요하다. 반드시 10, 100, 1000 단위로 구간화하지 말고 의미있는 구간으로 구간화



다. 파생변수

- 사용자(분석자)가 **특정 조건을 만족**하거나 **특정 함수에 의해 값을 만들어 의미를 부여한 변수**이다.
- 매우 **주관적일 수 있으므로 논리적 타당성을 갖추어 개발**해야 한다.
- **세분화, 고객행동 예측, 캠페인 반응 예측**에 매우 잘 활용된다.
- 파생변수는 상황에 따라 특정 상황에만 유의미하지 않게 대표성을 나타나게 할 필요가 있다.

예시

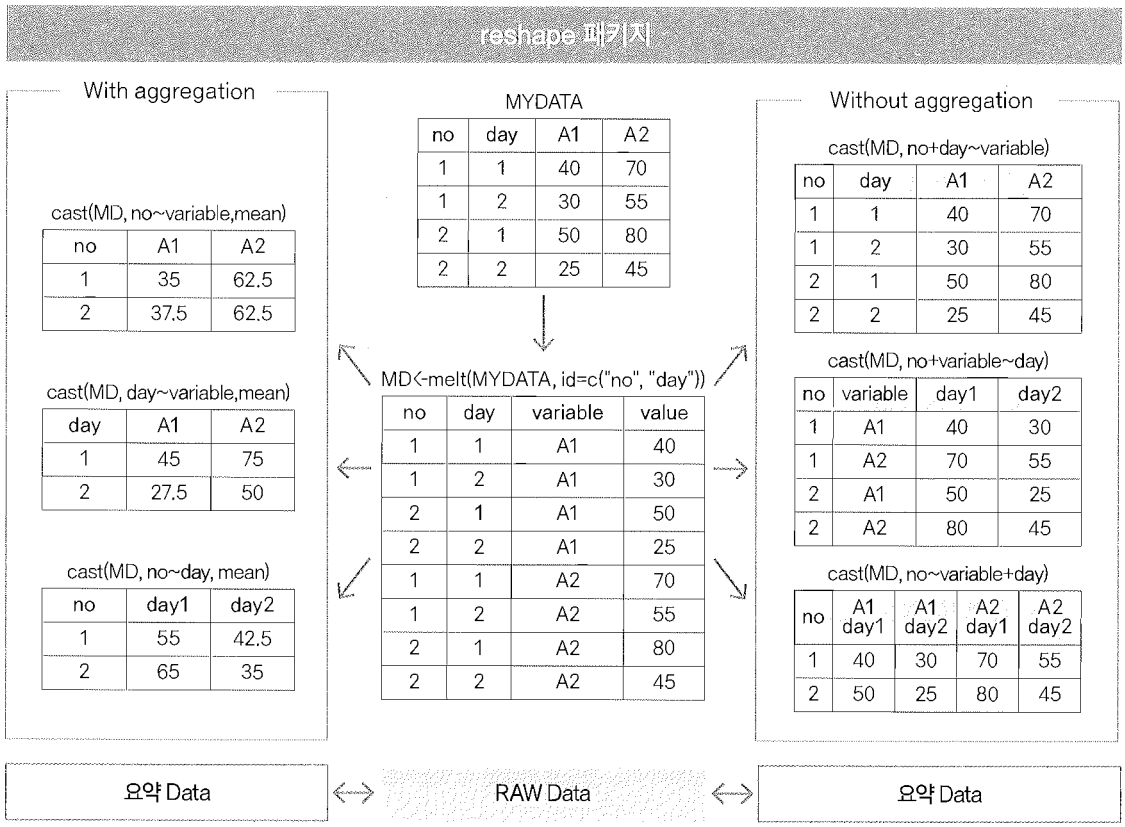
근무시간 구매지수	근무시간대에 거래가 발생하는 비율을 산출하여 활용
주 구매 매장 변수	고객의 주거래 매장을 예측하여 적절한 분석에 활용
주 활동 지역 변수	고객의 정보나 거래내용을 통해 주 활동지역을 예측하여 분석에 활용
주 구매상품 변수	상품을 추천하는데 활용 (1순위 상품을 구매하고 2순위 상품을 구매하지 않은 고객에게 추천)
구매상품 다양성 변수	고객이 다양한 상품이나 같은 브랜드 등을 구매하는 성향을 파악하여 분석에 필요한 변수로 변환
선호하는 가격대 변수	각자의 취향, 소득, 서비스 등에 따라 많이 투자하는 상품군이 있는데 주로 패션 분야에 중요하게 적용
시즌 선호고객 변수	각자 의미 있게 생각하는 날 소비가 많이 이루어지기 때문에 패턴을 파악하여 분석에 활용(주로 유통업)
라이프 스테이지 변수	고객이 속한 라이프 스테이지를 예측하여 행동을 이해하고 그들의 니즈와 가치를 파악하는데 활용
라이프스타일 변수	고객의 라이프스타일을 보고 상품구매를 유도하는데 활용
행사민감 변수	같은 상품도 행사를 할 때 구매하는 사람이 있고 행사와 관련 없이 구매하는 사람이 있는데 이런 행동 패턴을 파악하여 활용
휴면가망 변수	고객은 늘 구매하지 않기 때문에 고객의 취향이나 관심사가 변해 구매하지 않거나 경쟁사의 상품을 선호하게 되는 경우가 있는데 이를 파악하여 사전 대응에 활용
최대가치 변수	고객의 가치를 판단하여 어느 정도를 판매할 수 있는지를 예측하는데 활용
최적 통화 시간	콜센터에 걸려온 시간으로 고객의 직업 등을 고려한 통화시간을 예측하여 통화를 시도

라. reshape 의 활용

- reshape 패키지에는 melt( )와 cast( )라는 2개의 핵심 함수가 있다.(철을 녹이고 다시 틀에 넣어 모양을 만드는 과정에 비유하여, **녹이는 함수를 melt( ), 모양을 만드는 함수를 cast( )**로 사용한다.)

- 다음의 예시는 reshape 패키지의 주요 기능인 melt를 이용해 airquality데이터의 Month, id를 기준으로 모든 데이터를 표준형식으로 변환한다.
- 변수를 조합해 변수명을 만들고 변수들을 시간, 상품 등의 차원에 결합해 다양한 요약변수와 파생변수를 쉽게 생성하여 데이터 마트를 구성할 수 있게 한다.

reshape에 활용되는 cast와 melt함수의 R코딩 방식은 시험에 자주 나오니 반드시 학습하세요~



- melt() : 원데이터 형태로 만드는 함수
- cast() : 요약 형태로 만드는 함수

예시

- airquality data
  6개 변수 ("Ozone" "Solar.R" "Wind" "Temp" "Month" "Day"), 153개 자료

```
> head(airquality)
Ozone Solar.R Wind Temp Month Day
1 41 190 7.4 67 5 1
2 36 118 8.0 72 5 2
3 12 149 12.6 74 5 3
4 18 313 11.5 62 5 4
5 NA NA 14.3 56 5 5
6 28 NA 14.9 66 5 6
. . .
```

- melt 함수
  melt( ) : 쉬운 casting을 위해 적당한 형태로 만들어주는 함수
  melt(data, id = ...)

```
> melt(airquality, id=c("Month", "Day"), na.rm=T)
Month Day variable value
1 5 1 Ozone 41
2 5 2 Ozone 36
3 5 3 Ozone 12
4 5 4 Ozone 18
5 5 6 Ozone 28
6 5 7 Ozone 23
. . .
117 5 1 Solar.R 190
118 5 2 Solar.R 118
119 5 3 Solar.R 149
. . .
```

- cast 함수
  cast( ) : 데이터를 원하는 형태로 계산 또는 변형 시켜주는 함수
  cast(data, formula = ... ~ variable, fun)

```
> cast(aqm, Day ~ Month ~ variable)
, , variable = Ozone
Month
Day 5 6 7 8 9
1 41 NA 135 39 96
2 36 NA 49 9 78
3 12 NA 32 16 73
4 18 NA NA 78 91
. . .
, , variable = Solar.R
Month
Day 5 6 7 8 9
1 190 286 269 83 167
2 118 287 248 24 197
3 149 242 236 77 183
4 313 186 101 NA 189
. . .
```

```
, , variable = Temp
Month
Day 5 6 7 8 9
1 67 78 84 81 91
2 72 74 85 81 92
3 74 67 81 82 93
4 62 84 84 86 93
. . .
```

출 제 포인트 sql명령문과 sqldf를 활용한 명령문의 차이를 확인해봅시다!

② sqldf를 이용한 데이터 분석

- sqldf는 R에서 sql의 명령어를 사용 가능하게 해주는 패키지이다.
- SAS에서의 PROC SQL과 같은 역할을 하는 패키지다.

예시

- sql에서 사용하는 명령어 : select * from [data frame]
  → R에서 사용하는 명령어 : sqldf("select * from [data frame]")
- sql에서 사용하는 명령어 : select * from [data frame] numrows 10
  → R에서 사용하는 명령어 : sqldf("select * from [data frame] limit 10")
- sql에서 사용하는 명령어 : select * from [data frame] where [col] like 'char%'
  → R에서 사용하는 명령어 : sqldf("select * from [data frame] where [col] like 'char%' ")

- head([df]) : sqldf("select * from [df] limit 6")
- subset([df], grep1("qn%", [col])): sqldf("select * from [df] where [col] like 'qn%' ")
- subset([df], [col] %in% c("BF", "HF") : sqldf("select * from [df] where [col] in('BF', 'HF')")
- rbind([df1], [df2]) : sqldf("select * from [df1] union all select * from [df2]")
- merge([df1], [df2]) : sqldf("select * from [df1], [df2]")
- df[order([df]$[col], decreasing=T),] : sqldf("select * from [df] order by [col] desc")

- iris데이터를 활용한 예시

```
> sqldf("select * from iris")
Loading required package: tcltk
Sepal_Length Sepal_Width Petal_Length Petal_Width Species
1 5.1 3.5 1.4 0.2 setosa
2 4.9 3.0 1.4 0.2 setosa
3 4.7 3.2 1.3 0.2 setosa
4 4.6 3.1 1.5 0.2 setosa
5 5.0 3.6 1.4 0.2 setosa
. . .
```

plyr패키지의 활용방식과 R코딩을 확인하세요~

③ plyr을 이용한 데이터 분석

- plyr은 apply 함수에 기반해 데이터와 출력변수를 동시에 배열로 치환하여 처리하는 패키지이다.
- split-apply-combine : 데이터를 분리하고 처리한 다음, 다시 결합하는 등 필수적인 데이터 처리 기능을 제공한다.

	array	data frame	list	nothing
array	aaply	adply	alply	a_ply
data frame	daply	ddply	dlply	d_ply
list	laply	ldply	llply	l_ply
n replicates	raply	rdply	rlply	r_ply
function arguments	maply	mdply	mlply	m_ply

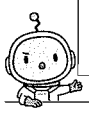
예시

- test data 불러오기

```
> test.data
year value
1 2011 31
2 2011 84
3 2011 66
. . .
9 2012 95
10 2012 83
11 2013 91
. . .
```

- test.data를 이용하여 sd와 mean의 비율인 변동계수 CV(Coefficient of Variation)를 산출

```
> dd.test <- ddply(test.data, "year", function(x) {
+ m.value <- mean(x$value)
+ sd.value <- sd(x$value)
+ cv <- round(sd.value/m.value, 4)
+ data.frame(cv.value=cv)
+  })
> dd.test
year cv.value
1 2011 0.4396
2 2012 0.3716
3 2013 0.6599
4 2014 0.6760
```



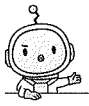
데이터 테이블과 데이터 프레임을 비교하여 묻는 문제가 출제 될 수 있습니다. 데이터 테이블의 특징을 꼭 기억하도록 합시다.

④ 데이터 테이블

- data.table 패키지는 R에서 **가장 많이 사용하는 데이터 핸들링 패키지** 중 하나이다.
- data.table은 큰 데이터를 **탐색, 연산, 병합 하는데 아주 유용**하다.
- 기존 data.frame 방식보다 **월등히 빠른 속도**이다.
- 특정 Column을 key 값으로 색인을 지정한 후 데이터를 처리한다.
- **빠른 그루핑과 Ordering, 짧은 문장 지원 측면에서 데이터 프레임보다 유용**하다.

예시

```
> install.packages("data.table")
> library(data.table)
> DF <- data.frame(x = runif(2.6e+07), y = rep(LETTERS, each = 10000))
> df <- data.frame(x = runif(2.6e+07), y = rep(letters, each = 10000))
> system.time(x <- DF[DF$y == "C", ])
  사용자 시스템 elapsed
  1.88 0.40 2.30
> DT <- as.data.table(DF)
> setkey(DT, y)
> system.time(x <- DT[J("C"), ])
  사용자 시스템 elapsed
  0.03 0.00 0.03
```



2절 데이터 가공

가. 개요

Data Exploration

- 데이터 분석을 위해 구성된 데이터의 변수들의 상태를 파악한다.

나. 종류

1) head(데이터셋), tail(데이터셋)

- 시작 또는 마지막 6개 record만 조회하는 함수

2) summary(데이터셋)

가) 수치형 변수 : 최댓값, 최솟값, 평균, 1사분위수, 2사분위수(중앙값), 3사분위수

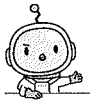
나) 명목형 변수 : 명목값, 데이터 개수

head(), summary() 함수는 시험에 자주 출제 되니 내용을 파악하세요

예시

- R의 diamonds data를 이용한 예시(head 함수)

```
> require(ggplot2)
> data(diamonds)
> dia.data <- diamonds
> head(dia.data)
carat cut color clarity depth table price x y z
1 0.23 Ideal E SI2 61.5 55 326 3.95 3.98 2.43
2 0.21 Premium E SI1 59.8 61 326 3.89 3.84 2.31
3 0.23 Good E VS1 56.9 65 327 4.05 4.07 2.31
4 0.29 Premium I VS2 62.4 58 334 4.20 4.23 2.63
```



예시

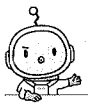
- R의 diamonds data를 이용한 예시(summary 함수)

```
> summary(dia.data)
```

carat	cut	color	clarity	depth
Min. :0.2000	Fair : 1610	D: 6775	SI1 :13065	Min. :43.00
1st Qu.:0.4000	Good : 4906	E: 9797	VS2 :12258	1st Qu.:61.00
Median :0.7000	Very Good:12082	F: 9542	SI2 : 9194	Median :61.80
Mean :0.7979	Premium :13791	G:11292	VS1 : 8171	Mean :61.75
3rd Qu.:1.0400	Ideal :21551	H: 8304	VVS2 : 5066	3rd Qu.:62.50
Max. :5.0100		I: 5422	VVS1 : 3655	Max. :79.00
		J: 2808	(Other): 2531	

table	price	x	y	z
Min. :43.00	Min. : 326	Min. : 0.000	Min. : 0.000	Min. : 0.000
1st Qu.:56.00	1st Qu.: 950	1st Qu.: 4.710	1st Qu.: 4.720	1st Qu.: 2.910
Median :57.00	Median : 2401	Median : 5.700	Median : 5.710	Median : 3.530
Mean :57.46	Mean : 3933	Mean : 5.731	Mean : 5.735	Mean : 3.539
3rd Qu.:59.00	3rd Qu.: 5324	3rd Qu.: 6.540	3rd Qu.: 6.540	3rd Qu.: 4.040
Max. :95.00	Max. :18823	Max. :10.740	Max. :58.900	Max. :31.800

※ Min : 최솟값, 1st Qu : 1사분위수, Median : 중위수, Mean : 평균, 3st Qu : 3사분위수, Max : 최댓값



16 DAY

② 변수 중요도

가. 개요

- 변수 선택법과 유사한 개념으로 모형을 생성하여 사용된 변수의 중요도를 살피는 과정이다.

나. 종류

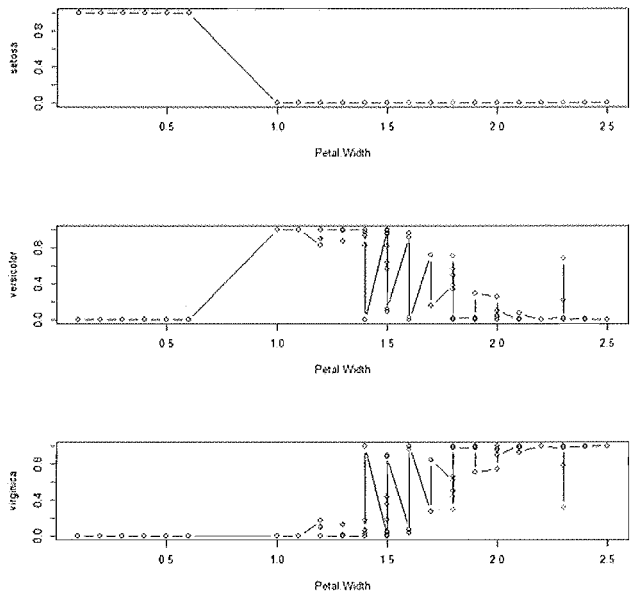
1) klaR 패키지

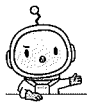
- 특정 변수가 주어졌을 때 클래스가 어떻게 분류되는지에 대한 에러율을 계산해주고, 그래픽으로 결과를 보여주는 기능을 한다.
- greedy.wilks( ) : 세분화를 위한 stepwise forward 변수선택을 위한 패키지, 종속변수에 가장 영향력을 미치는 변수를 wilks lambda를 활용하여 변수의 중요도를 정리
  (Wilk's Lambda = 집단내분산/총분산)

예시

- plineplot()을 이용한 iris데이터 예제

```
> iris2 <- iris[,c(1,3,5)]
> plineplot(Species ~., data=iris2, method="lda",x=iris[,4],
   xlab="Petal.Width")
[1] 0.03333333
```





출 제 포인트

변수의 구간화에서 자주 활용되는 방식에는 Binning과 의사결정나무가 있습니다.

변수의 구간화

가. 개요

- **연속형** 변수를 분석 목적에 맞게 활용하기 위해 구간화하여 모델링에 적용한다.

  ※ 일반적으로 10진수 단위로 구간화하지만, 구간을 5개로 나누는 것이 보통이며, 7개 이상의 구간을 잘 만들지 않는다.

- 신용평가모형, 고객 세분화와 같은 시스템에서 모형에 활용하는 각 변수들을 구간화해서 구간별로 점수를 적용하는 스코어링 방식으로 많이 활용되고 있다.

예시

타자들의 연봉데이터(단위:백만원)

630	400	180	550	162	500	270	200
192	200	200	80	310	200	135	300
160	70	220	350	700	88	100	250
400	400	70	160	185	170	85	202
80	72	100	350	85	500	140	100

구간	연봉(단위:백만원)
1	~20미만
2	20이상~40미만
3	40이상~60미만
4	60이상~80미만
5	80이상~100미만
6	100이상~

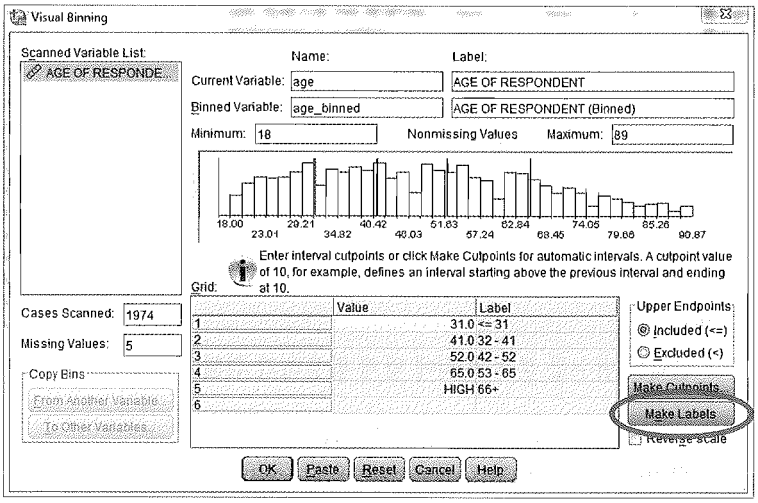
나. 구간화 방법

1) Binning

- 신용평가모형의 개발에서 연속형 변수(부채비율 등)를 범주형 변수로 구간화 하는데 자주 활용되고 있는 방법이다.

예시

연속형 변수를 오름차순 정렬한 후 각각 동일한 개수의 레코드를 50개의 깡통(bin)에 나누어 담고 각 깡통의 부실율을 기준으로 병합하면서 최종 5~7개의 깡통으로 부실율의 역전이 생기지 않게 합치면서 구간화 한다. <아래 그림은 나이를 Binning 적용하여 5개의 Bin으로 구간화하고 있다.



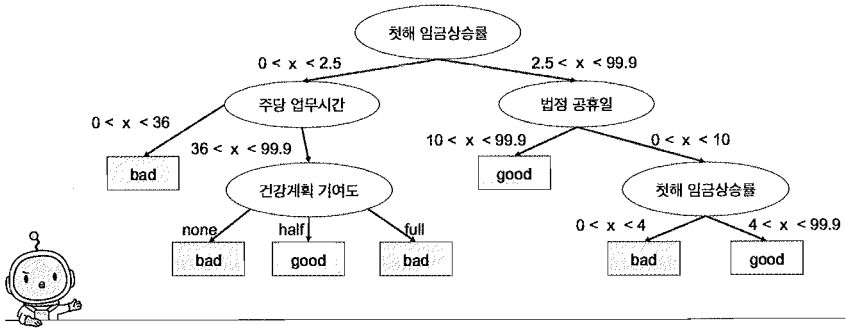
2) 의사결정나무

- 세분화 또는 예측에 활용되는 의사결정나무 모형을 사용하여 입력변수들을 구간화할 수 있다. 의사결정나무에서는 동일한 변수를 여러 번의 분리 기준으로 사용이 가능하기 때문에 연속 변수가 반복적으로 선택될 경우, 각각의 분리 기준값으로 연속형 변수를 구간화할 수 있다.

예시

아래 그림은 올해 임금상승률을 예측하는 의사결정나무이며, 이 모형에서 중복으로 사용된 "첫해 임금상승률" 변수의 경우, 2.5%이하, 2.5%초과~4%이하, 4%초과~99.9%이하, 99.9%초과로 구간화 할 수 있다.

구간	첫해 임금상승률
1	~ 2.5이하
2	2.5초과 ~ 4.0이하
3	4.0초과 ~ 99.9이하
4	99.9초과 ~



3절 기초 분석 및 데이터 관리

① 데이터 EDA (탐색적 자료 분석)

- 데이터의 분석에 앞서 전체적으로 데이터의 특징을 파악하고 데이터를 다양한 각도로 접근한다.
- summary()를 이용하여 데이터의 기초통계량을 확인한다.

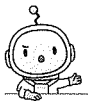
예시

- diamond 데이터의 summary 결과

```
> summary(dia.data)
```

carat	cut	color	clarity	depth
Min. :0.2000	Fair : 1610	D: 6775	SI1 :13065	Min. :43.00
1st Qu.:0.4000	Good : 4906	E: 9797	VS2 :12258	1st Qu.:61.00
Median :0.7000	Very Good:12082	F: 9542	SI2 : 9194	Median :61.80
Mean :0.7979	Premium :13791	G:11292	VS1 : 8171	Mean :61.75
3rd Qu.:1.0400	Ideal :21551	H: 8304	VVS2 : 5066	3rd Qu.:62.50
Max. :5.0100		I: 5422	VVS1 : 3655	Max. :79.00
		J: 2808	(Other): 2531	

table	price	x	y	z
Min. :43.00	Min. : 326	Min. : 0.000	Min. : 0.000	Min. : 0.000
1st Qu.:56.00	1st Qu.: 950	1st Qu.: 4.710	1st Qu.: 4.720	1st Qu.: 2.910
Median :57.00	Median : 2401	Median : 5.700	Median : 5.710	Median : 3.530
Mean :57.46	Mean : 3933	Mean : 5.731	Mean : 5.735	Mean : 3.539
3rd Qu.:59.00	3rd Qu.: 5324	3rd Qu.: 6.540	3rd Qu.: 6.540	3rd Qu.: 4.040
Max. :95.00	Max. :18823	Max. :10.740	Max. :58.900	Max. :31.800



② 결측값 인식

- 결측값은 NA, 99999999, ' '(공백), Unknown, Not Answer 등으로 표현되는 것으로 결측값을 처리하기 위해서 시간을 많이 사용하는 것은 비효율적이다.
- 결측값 자체의 의미가 있는 경우도 있는데 예를 들면 쇼핑몰 가입자 중 특정 거래 자체가 존재하지 않는 경우와 인구통계학적 데이터(Demographic Data)에서 아주 부자이거나 아주 가난한 경우 자신의 정보를 잘 채워 넣지 않기 때문에 가입자의 특성을 유추하여 활용할 수 있다.
- 결측값 처리는 전체 작업속도에 많은 영향을 준다.

예시

- 결측값이 있는 자료의 평균 구하기

```
>x<-c( 1, 2, 3, NA)
> mean(x)
[1] NA
> mean( x, na.rm=T)
[1] 2
```

- na.rm을 이용해 NA를 제거한 후 평균을 구할 수 있다.

레코드를 삭제하면 데이터 수가 줄어 활용할 수 있는 변수도 작아집니다. 즉 데이터활용의 효율성이 떨어진다는 것이죠. 이러한 단점을 보완하기 위해 평균대치법, 단순확률 대치법을 이용합니다.

3 결측값 처리 방법

가. 단순 대치법(Single Imputation)

1) Completes Analysis

- 결측값이 존재하는 레코드를 **삭제**한다.

2) 평균대치법(Mean Imputation)

- 관측 또는 실험을 통해 얻어진 **데이터의 평균으로 대치**한다.
- **비조건부** 평균 대치법 : 관측데이터의 평균으로 대치
- **조건부** 평균 대치법(Regression Imputation) : 회귀분석을 활용한 대치법

Y_1	Y_2	Y_3	$\hat{Y}_3$
10	15	20	20
12	25	30	30
15	35	40	40
25	49	57	57
30	49	59	59
35	55	65	65
37	47	70	70
40	60	$?_1$	76.89
42	65	$?_2$	81.67
50	70	$?_3$	92.39

$$\hat{Y} = \beta_0 + \beta_1 y_1 + \beta_2 y_2 + \varepsilon,\ i = 1,2,...,7$$

$$\beta_0 = 3.69,\ \beta_1 = 0.099,\ \beta_2 = 0.56$$

$$?_1 = 3.69 + 0.099 \times 40 + 0.56 \times 60 = 76.89$$

3) 단순확률 대치법(Single Stochastic Imputation)

- 평균대치법에서 추정량 표준 오차의 과소 추정문제를 보완하고자 고안된 방법으로 Hot deck 방법, **Nearest Neighbor 방법** 등이 있다.

R프로그램에서 결측값의 확인 및 처리방식을 기억합시다.

나. 다중 대치법(Multiple Imputation)

- 단순대치법을 한 번만 하지 않고 m번의 대치를 통해 m개의 가상적 완전 자료를 만드는 방법이다.
- 1단계 : 대치(Imputation Step), 2단계 : 분석(Analysis Step), 3단계 : 결합(Combination Step)
- Amelia-time series cross sectional data set(여러 국가에서 매년 측정된 자료)에서 Bootstrapping Based Algorithm 을 활용한 다중 대치법이다.

④ R에서 결측값 처리

가. 관련 함수

함수	내용
complete.cases( )	데이터내 레코드에 결측값이 있으면 FALSE, 없으면 TRUE로 반환
is.na( )	결측값을 NA로 인식하여 결측값이 있으면 TRUE, 없으면 FALSE로 반환
DMwR 패키지의 centralImputation( )	NA 값에 가운데 값(Central Value)으로 대치, 숫자는 중위수, 요인(Factor)은 최빈값으로 대치
DMwR 패키지의 knnImputation( )	NA 값을 k최근 이웃 분류 알고리즘을 사용하여 대치하는 것으로, k개 주변 이웃까지의 거리를 고려하여 가중 평균한 값을 사용
Amelia 패키지의 amelia( )	• time-series-cross-sectional data set(여러 국가에서 매년 측정된 자료)에서 활용 ※ 랜덤포레스트(Random Forest)모델은 결측값이 존재할 경우, 바로 에러가 발생 • RandomForest 패키지의 rfImpute( ) 함수를 활용하여 NA 결측값을 대치한 후 알고리즘에 적용

Bad Data로 판명된 데이터는 삭제하는 것이 바람직합니다.

⑤ 이상값(Outlier) 인식과 처리

가. 이상값이란?

- 의도하지 않게 잘못 입력한 경우(Bad Data)
- 의도하지 않게 입력되었으나 분석 목적에 부합되지 않아 제거해야 하는 경우(Bad Data)

- 의도하지 않은 현상이지만 분석에 포함해야 하는 경우
- 의도된 이상값(Fraud, 불량)인 경우
- 이상값을 꼭 제거해야 하는 것은 아니기 때문에 분석의 목적이나 종류에 따라 적절한 판단이 필요하다.

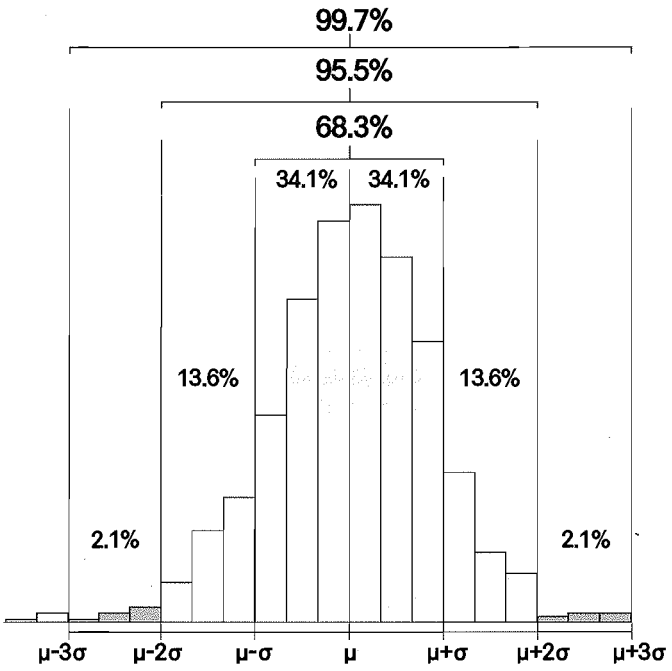
이상치 사용 분야
사기 탐지, 의료(특정환자에게 보이는 예외적인 증세), 네트워크 침입탐지 등

ESD는 단답형으로 자주 출제 되는 문제이므로 숙지하시고 넘어가시기 바랍니다.

나. 이상값의 인식 방법

1) ESD(Extreme Studentized Deviation)

- 평균으로부터 3 표준편차 떨어진 값(각 0.15%)



<출처 : 위키피디아>

2) 기하평균-2.5×표준편차 < data < 기하평균 +2.5×표준편차

3) 사분위수 이용하여 제거하기(상자 그림의 outer fence 밖에 있는 값 제거)
이상값 정의 : Q1-1.5(Q3-Q1)< data< Q3+1.5(Q3-Q1)를 벗어나는 데이터

출제 포인트 극단값 절단 방법을 활용해 데이터를 제거하는 것 보다는 극단값 조정 방법을 이용하는 것이 데이터 손실율도 적고, 설명력도 높아집니다.

다. 극단값 절단(Trimming) 방법

1) 기하평균을 이용한 제거

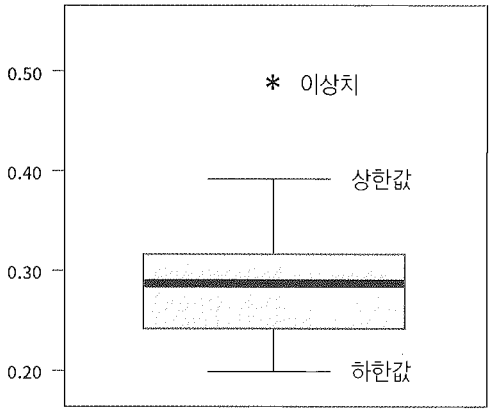
- geo_mean

2) 하단, 상단 % 이용한 제거

- 10% 절단(상하위 5%에 해당되는 데이터 제거)

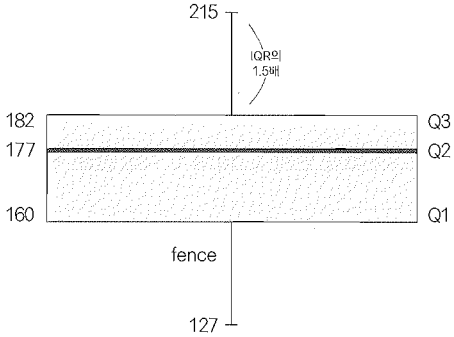
라. 극단값 조정(Winsorizing) 방법

- 상한값과 하한값을 벗어나는 값들을 **하한, 상한값으로 바꾸어 활용**하는 방법이다.



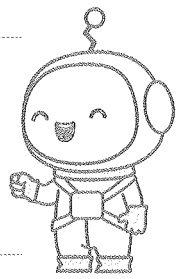
예시

- 상자수염그림(Box Plot을 통한 예시)



- 이상치를 구하기 위해 3Q에서 1Q를 뺀 Inter Quartile Range(IQR)을 구한결과 22로 나타났다. (182-160 = 22)
- IQR의 약 1.5배(33)이 최소, 최대값에서 벗어날 경우 이상점으로 구분한다.

MEMO



3과목 | 3장

데이터 마트

✓22회 기출

01 데이터의 한 부분으로 특정 사용자가 관심을 갖고 있는 데이터를 담은 비교적 작은 규모의 데이터 웨어하우스는 무엇이라고 하는가?

① 데이터 베이스
② 데이터 마트
③ 데이터 마이닝
④ 데이터 프레임

02 변수를 조합해 변수명을 만들고 변수들을 시간, 상품 등의 차원에 결합해 다양한 요약변수와 파생변수를 쉽게 생성하여 데이터 마트를 구성할 수 있는 패키지는 무엇인가?

① ETL ② reshape ③ OLAP ④ rattle

✓10회 기출

03 파생변수는 사용자가 특정 조건을 만족하거나 특정 함수에 의해 값을 만들어 의미를 부여한 변수이다. 다음 중 파생변수의 설명으로 적절한 것은?

① 파생변수는 매우 주관적인 변수일 수 있으므로 논리적 타당성을 갖춰야 한다.
② 파생변수는 많은 모델에서 공통적으로 많이 사용될 수 있다.
③ 파생변수는 재활용성이 높다.
④ 파생변수는 다양한 모델을 개발해야 하는 경우, 효율적으로 사용할 수 있다.

✓10회 기출

04 많은 기업에서 평균거래주기를 3~4배 이상 초과하거나 다음 달에 거래가 없을 것으로 예상되는 고객을 (㉠)으로 정의하고 있다. 다음 중 (㉠)에 가장 적절한 것은?

① 신규고객 ② 우량고객 ③ 가망고객 ④ 휴면고객

✓6회 기출

05 아래 표는 데이터의 변경을 통해 새로운 구조의 데이터셋을 구성하고자 할때 사용하는 R 프로그램 중 melt함수와 cast 함수의 예시이다. 데이터셋 MD를 새로운 데이터 형태로 변경하기 위한 cast 함수를 활용한 R 프로그램 중 옳은 것은?

〈DATA 명 : MD〉

ID	Time	Variable	Value
1	1	X1	5
1	2	X1	3
2	1	X1	6
2	2	X1	2
1	1	X2	6
1	2	X2	5
2	1	X2	1
2	2	X2	4

〈새로운 데이터〉

ID	Variable	Time1	Time2
1	X1	5	3
1	X2	6	5
2	X1	6	2
2	X2	1	4

① cast(md, id~variable +time)

② cast(md, id+variable~time)

③ cast(md, id+time~variable)

④ cast(md, id~variable, mean)

✓9회 기출

06 아래의 정의가 가리키는 데이터 마트의 구성요소로 가장 적절한 것은?

> 특정한 의미를 갖는 작위적 정의에 의한 변수로, 사용자가 특정 조건을 만족하거나 특정 함수에 의해 값을 만들어 의미를 부여한 변수

① 반응변수 ② 파생변수 ③ 설명변수 ④ 요약변수

✓7회 기출

07 아래의 왼쪽 자료를 오른쪽의 형태로 변환하기 위한 명령어로 적절한 것은?

>head(airquality, 10)

	Ozone	Solar.R	Wind	Temp	Month	Day
1	41	190	7.4	67	5	1
2	36	118	8	72	5	2
3	12	149	12.6	74	5	3
4	18	313	11.3	62	5	4
5	NA	NA	14.3	56	5	5
6	28	NA	14.9	66	5	6
7	23	299	8.6	65	5	7
8	19	99	13.8	59	5	8
9	8	19	20.1	61	5	9
10	NA	194	8.6	69	5	10

> aqm

	month	day	variable	value
1	5	1	Ozone	41
2	5	2	Ozone	36
3	5	3	Ozone	12
4	5	4	Ozone	18
5	5	6	Ozone	28
6	5	7	Ozone	23
7	5	8	Ozone	19
115	9	29	Ozone	18
116	9	30	Ozone	20
117	5	1	Solar.R	190
118	5	2	Solar.R	118
119	5	3	Solar.R	149
564	9	28	Temp	75
565	9	29	Temp	76
566	9	30	Temp	68

① aqm<-melt(airquality, id=c("Month","Day"), na.rm=TRUE)

② aqm<-melt(airquality, id=c("Month","Day"))

③ aqm<-melt(airquality, id=c("Ozone","Solar.R","Wind","Temp"), na.rm=TRUE)

④ aqm<-melt(airquality, id=c ("Ozone","Solar.R","Wind","Temp"))

08 "iris"라는 데이터셋에서 데이터의 내용을 조회할 때, R프로그램으로 적절한 것은?

① plyr("select*from iris")

② sql("select*from iris")

③ mysql("select*from iris")

④ sqldf("select*from iris")

✓14회 기출

09 chickwts 데이터 프레임은 여섯가지 종류의 닭 사료 첨가물(feed)과 각 사료를 먹인 닭의 무게(weight)를 변수로 가진다. 아래의 (1)의 기초통계량과 각 feed별 weight의 평균을 계산하여, 아래 (2)와 같은 결과물을 만들기 위한 코드로 다음 중 가장 적절한 것은?

(1)

```
> head(chickwts)
  weight      feed
1    179 horsebean
2    160 horsebean
3    136 horsebean
4    227 horsebean
5    217 horsebean
6    168 horsebean
> summary(chickwts)
     weight             feed
 Min.   :108.0   casein   :12
 1st Qu.:204.5   horsebean:10
 Median :258.0   linseed  :12
 Mean   :261.3   meatmeal :11
 3rd Qu.:323.5   soybean  :14
 Max.   :423.0   sunflower:12
```

(2)

```
        feed groupmean
1     casein  323.5833
2 horsebean  160.2000
3    linseed  218.7500
4  meatmeal  276.9091
5    soybean  246.4286
6  sunflower  328.9167
```

① ddply(chickwts, ~feed, groupmean=mean(weight))

② ddply(chickwts, weight~feed, summarize, groupmean=mean(weight))

③ ddply(chickwts, ~feed, summarize, groupmean=mean(weight))

④ ddply(chickwts, weight~feed, groupmean=mean(weight))

✓7회 기출

10 다음 중 결측치에 대한 설명으로 가장 부적절한 것은?

① 해당 칸이 비어있는 경우 결측치 여부는 알기 쉽다.

② 관측치가 있지만 실상은 Default 값이 기록된 경우에도 결측치로 처리해야 하는 것이 바람직하다.

③ 결측치가 있는 경우 다양한 대치(Imputation)방법을 사용하여 완전한 자료로 만든 후 분석을 진행할 수 있다.

④ 결측치가 20% 이상인 경우에는 해당 변수를 제거하고 분석해야 한다.

11 다음은 결측값을 확인하고 결측값을 대치하는데 활용되는 R 함수들이다. 설명이 잘못된 것을 고르시오.

① complete.cases( ) : 데이터 내 레코드에 결측값이 있으면 TRUE, 없으면 FALSE를 반환하는 함수

② is.na( ) : 결측값이 NA인지 여부를 판단하여 반환하는 함수

③ knnImputation( ) : NA 값을 k 최근 이웃 분류 알고리즘을 사용하여 대치하는 함수로 k개 주변 이웃까지의 거리를 고려하여 가중 평균한 값을 대치해 주는 함수

④ rfImpute( ) : 랜덤 포레스트 모형의 경우, 결측값이 있으면 에러를 발생하기 때문에 랜덤포레스트 패키지에서 NA 결측값을 대치하도록 하는 함수

✓25회 기출

12 결측값은 관측되어 얻어지는 실험 자료에서 종종 나타나는 현상이다. 결측값을 분석할 수 있는 통계분석 방법론으로 대치법이 있다. 다음 중 결측값을 처리하는 방법에 대한 설명 중 부적절한 것은?

① Complete Analysis는 불완전 자료를 모두 삭제하고 완전한 관측치만으로 자료를 분석하는 방법이다. 그러나 부분적 관측자료를 사용하므로 통계적 추론의 타당성 문제가 있다.

② 평균대치법은 자료의 평균값으로 결측값을 대치하여 불완전한 자료를 완전한 자료로 만들어 분석하는 방법이다.

③ 단순확률대치법은 평균대치법에서 추정량 표준오차의 과소 추정문제를 보완하고자 고안된 방법이다.

④ 다중대치법은 단순대치법을 한 번만 하지 않고 m번 대치를 통해 m개의 가상적 완전 자료를 만들어서 분석하는 방법이다. 추정량의 과소추정이나 계산의 난해성 문제가 보완된 방법이다.

✓23회 기출

13 이상치를 찾는 것은 데이터 분석에서 데이터 전처리를 어떻게 할지 검정할 때 사용할 수 있다. 다음 중 상자그림을 이용하여 이상치를 판정하는 방법에 대한 설명으로 가장 부적절한 것은?

① IQR = Q3 − Q1이라고 할 때, Q1 − 1.5*IQR < x < Q3 + 1.5*IQR을 벗어나는 x를 이상치라고 규정한다.

② 평균으로부터 3*표준편차 벗어나는 것들을 비정상이라 규정하고 제거한다.

③ 이상치는 변수의 분포에서 벗어난 값으로 상자 그림을 통해 확인할 수 있다.

④ 이상치는 분포를 왜곡할 수 있으나 실제 오류인자에 대해서는 통계적으로 실행하지 못하기 때문에 제거여부는 실무자들을 통해서 결정하는 것이 바람직하다.

✓11회 기출

14 다음 중 이상값 검색을 활용한 응용시스템으로 가장 적절한 것은?

① 장바구니분석 시스템
② 데이터 마트
③ 교차판매 시스템
④ 부정사용방지 시스템

✓5회 기출

15 이상치에 대한 설명으로 가장 부적절한 것은?

① 군집분석을 이용하여 다른 데이터들과 거리상 멀리 떨어진 데이터를 이상치로 판정한다.
② 데이터를 측정과정이나 입력하는 과정에서 잘못 포함된 이상치는 삭제한 후 분석한다.
③ 설명변수의 관측치에 비해 종속변수의 값이 상이한 값을 이상치라 한다.
④ 통상 평균으로부터 표준편차의 3배가 되는 점을 기준으로 이상치를 정의한다.

16 다음은 이상값(Outlier)에 대한 설명이다. 잘못 설명한 내용을 고르시오.

① 부정사용방지 시스템이나 부도예측시스템에서는 이상값(Outlier)이라도 의미가 있으므로 제거하지 않는다.
② 이상값 인식에 있어서 가장 많이 활용하는 방법은 ESD(Extreme Studentized Deviation)으로 평균에서 3 표준편차를 벗어나는 경우 이상값으로 인식하는 방법이다.
③ 이상값의 처리에 있어서 극단값 절단 방법과 조정 방법이 있으며 조정의 경우, 제거 방법에 비해 데이터 손실율이 높아 설명력이 낮아지는 단점이 있다.
④ 의도하지 않게 잘못 입력된 데이터인 경우 Bad Data 에 해당되며 이러한 경우, 데이터를 제거하여 분석한다.

✓20회 기출

17 R에서 반복문을 다중으로 사용할 경우 계산 시간이 현저하게 떨어지는 단점이 있다. 다음 함수 중 Multi-Core를 사용하여 반복문을 사용하지 않고도 매우 간단하고 빠르게 처리할 수 있는 데이터 처리 함수를 포함하고 있는 패키지로 적절한 것은?

① plyr
② sqldf
③ caret
④ party

18 데이터 전처리 단계에서 데이터의 이상치(Outlier) 에 대한 설명으로 틀린 것은?

> 이상치(Outlier) 탐지의 목적은 대부분의 객체들과 다른 객체들을 찾는 것이다. 이상치 탐지는 속성값들의 일반적인 값들과 상당히 편차가 큰 값을 가지므로 편차 탐지(Deviation Detection)라고도 한다. 그러나 이상치는 반드시 비정상적인 객체를 의미하지는 않는다.

① 최댓값과 최솟값
② 데이터 입력 시 오타로 인해 잘못 입력된 경우
③ 분석 목적에 부합되지 않아 제거해야 하는 경우
④ 부정사용방지 시스템에서 의도된 이상 값

✓9회 기출

19 아래는 이상치(Outlier) 탐지에 대한 설명이다. 다음 중 이상치를 유용하게 사용하는 분야의 예로 부적절한 것은?

① 사기탐지 - 도난당한 신용카드의 구매 행위는 원 소유자의 행위와 다를 수 있다. 평상시의 행위와 다른 구매패턴을 조사하여 사기를 탐지할 수 있다.
② 환경파괴 - 자연 세계에서는 환경에 중요한 영향을 줄 수 있는 홍수, 가뭄 같은 사건들이 있다. 그러나 이러한 사건은 정상적인 환경에서 발생하는 사건으로 해석할 수 있다.
③ 의료 - 특정 환자에게 보이는 예외적인 증세나 검사 결과는 잠재적인 건강 문제를 나타낸다.
④ 침입탐지 - 컴퓨터 네트워크에 대한 공격은 보편화되었다. 침입의 다수는 네트워크에 대한 예외적인 행위를 감시하는 경우에 탐지할 수 있다.

✓22회 기출

20 평균으로부터 t Standard Deviation 이상 떨어져 있는 값들을 이상값(Outlier)으로 판단하고 t는 3으로 설정하는 이상값 검색 알고리즘은?

( )

정답 및 해설

01	②	11	①
02	②	12	④
03	①	13	②
04	④	14	④
05	②	15	②
06	②	16	③
07	①	17	①
08	④	18	①
09	③	19	②
10	②	20	ESD(Extreme Studentized Deviation)

01. 데이터 마트란 데이터 웨어하우스와 사용자 사이의 중간층에 위치한 것으로 하나의 주제 또는 하나의 부서 중심의 데이터 웨어하우스이다. (**정답 : ②**)

02. reshape 패키지는 데이터를 원하는 형태로 바꿔주는 melt함수와 원하는 부분만을 선택하는 cast함수로 구성되어 있다. (**정답 : ②**)

03. 파생변수는 사용자가 특정 조건을 만족하거나 특정 함수에 의해 값을 만들어 의미를 부여한 변수로서, 매우 주관적일 수 있으므로 논리적 타당성을 갖추어 개발해야 한다. (**정답 : ①**)

04. (**정답 : ④**)

05. csat 함수는 모양을 만드는 함수로서 오른쪽의 각 ID와 Variable에 대해 Time의 Value를 확인하는 것이므로 cast(md, id+variable~time)이 정답이다. (**정답 : ②**)

06. 아래의 정의는 파생변수에 대한 설명이다. (**정답 : ②**)

07. melt 함수는 데이터를 재구성하기 위한 함수로서 id는 month와 day이고 각 variable별로 value 값을 나타내고 NA값은 na.rm=TRUE로 제외했음을 알 수 있다. (**정답 : ①**)

08. sqldf 패키지는 R에서 sql의 명령어를 사용가능하게 해주는 패키지이다. (**정답 : ④**)

09. 각 feed별 weight의 평균을 계산하기 위해서는 ~feed, summarize, mean이라는 명령어가 있는 것이 정답이다. (정답 : ③)

10. 관측치가 기록된 값을 결측치로 처리하여 분석에 활용하는 것은 옳지 않다. Default 값이 기록된 경우라도 그 값의 의미를 가지고 있기 때문에 결측치로 처리하면 분석에 큰 오류로 작용할 수도 있다. (정답 : ②)

11. complete.cases 함수는 레코드에 결측값이 없으면 TRUE, 있으면 FALSE 를 반환하는 함수이다. (정답 : ①)

12. 다중대치법은 추정량의 표준오차의 과소추정 또는 계산의 난해성 문제가 보완된 방법이다. (정답 : ④)

13. '이상치'라고 규정한 자료는 분석에서 제외를 할 수 있지만 무조건적으로 제거할 수는 없다. (정답 : ②)

14. 이상값을 검색하여 한 집단에서 매우 크거나, 매우 작으면 의심되는 대상이므로 부정사용방지 시스템에 활용이 가능하다. (정답 : ④)

15. 이상치는 분석에 의미가 있을 수 있으므로 제거하면 안된다. (정답 : ②)

16. 이상치를 절단이나 조정하는 경우 제거방법에 비해 데이터의 손실율이 낮아지기 때문에 설명력이 높아지는 장점이 생긴다. (정답 : ③)

17. plyr는 데이터 처리에 필요한 R 패키지로 데이터를 분할하고 분할된 결과에 함수를 적용한 뒤 결과를 재조합하는 함수를 포함한다. (정답 : ①)

18. 최대값과 최소값은 이상치(Outlier)로 볼 수 없다. (정답 : ①)

19. 이상치 탐지에 활용할 수 있는 분야는 사기탐지, 의료, 침입탐지 등에 활용이 가능하지 환경 파괴에는 적용하기 어렵다. (정답 : ②)

20. 정답 : ESD(Extreme Studentized Deviation)