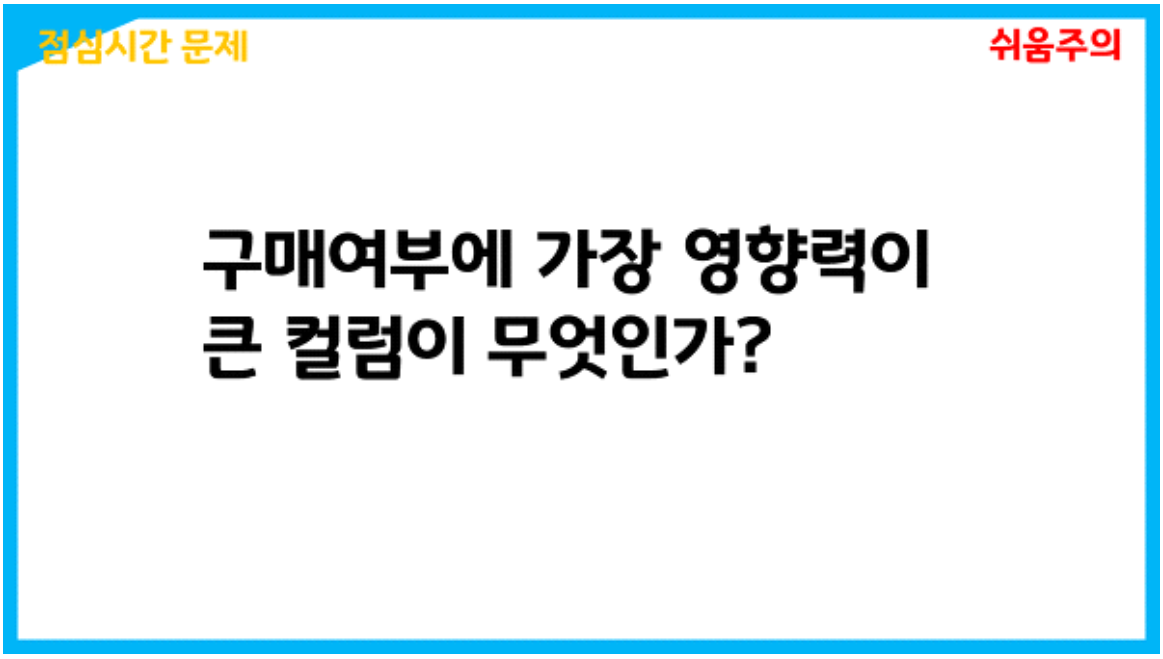
수업 전체 일정

2020년 6월 26일 금요일 오전 9:46

번호	과목	일정	시험	책
1	SQL	4월 1일 ~ 4월 24일	4월 29일	
2	SQL튜닝	4월 27일 ~ 5월 8일		
3	파이썬	5월 11일 ~ 6월 5일	6월 12일	파이썬200제 -정보문화사 통계학 -성안당 통계학 개론 -Knou
4	리눅스	6월 1일 ~ 6월 5일	6월 19일	
5	R을 이용한 머신러닝	6월 8일 ~ 7월 3일	**7월 21일 (발표)**	R을 활용한 기계학습 -에이콘
6	파이썬을 이용한 머신러닝	7월 6일 ~ 7월 17일	8월 7일	파이썬 머신러닝 판다스 데이터 분석 -정보문화사
7	딥러닝과 케글 도전	7월 20일 ~ 8월 21일	8월 27일	밑바닥부터 시작하는 딥러닝 -한빛미디어
8	강화학습	8월 24일 ~ 9월 17일	9월 24일	파이썬과 케라스 배우는 강화학습 -위키북스
9	최종 프로젝트	9월 21일 ~ 10월5일	10월 6일 **(발표)**	

6월 26일 점심시간 문제. 정보획득량

2020년 6월 26일 금요일 오전 9:28



6월 26일 점심시간 문제. 아래의 데이터 프레임을 생성하고 맨 끝의 컬럼인 buy_yn(구매여부)에 가장
영향력이 큰 컬럼이 카드유무(card_yn) 인지 intro_yn(지인소개) 인지 before_buy_yn(전에 구매한 경험) 인지
확인하시오

```
> buy
  cust_name card_yn intro_yn before_buy_yn buy_yn
1     SCOTT       Y        Y             Y      Y
2     SMITH       Y        Y             Y      Y
3     ALLEN       N        N             Y      N
4     JONES       Y        N             N      Y
5      WARD       Y        Y             Y      Y
>
```

```
buy <- data.frame(
  cust_name=c('SCOTT','SMITH','ALLEN','JONES','WARD'),
  card_yn=c('Y','Y','N','Y','Y'),
  intro_yn=c('Y','Y','N','N','Y'),
  before_buy_yn=c('Y','Y','Y','N','Y'),
  buy_yn=c('Y','Y','N','Y','Y') )
```

From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SeFi&contentval=0004Szzzzzzzzzzzzzzzzzzzzzzzz&datanum=276&page=1&prev_page=0&firstbbsdepth=&lastbbsdepth=zzzzzzzzzzzzzzzzzzzzzzzzzzzzz&listnum=20>

7월 21일 (발표) 데이터 구하는 방법

2020년 6월 26일 금요일 오전 9:49

<7월 21일 발표 준비>

데이터 (은행 대출, 유방암, 독버섯, 와인, 화장품 구매자, 독감, 영화 장르)

책에 있는 데이터는 양질의 데이터이기 때문에 결과와 정확도가 매우 잘 나왔지만,
현업에서는 그런 데이터를 접하기 쉽지 않다.

프로젝트를 진행하면서 현업에 들어갔을 때 부딪히는 어려움을 간접적으로 경험해 보는 과정이 필요하다.

1. 데이터 수집하는 곳 -> **케글**, 공공데이터포털, UCI 머신러닝 데이터셋, 직접 웹스크롤링

돼지삼형제에 비유하자면 직접 웹스크롤링 하면 벽돌 집을 지을 수 있다.
(1. 짚 2. 나무 3. 벽돌)

2. 머신러닝 알고리즘 -> 배웠던 것을 다 적용시켜서 정확도 90%를 넘기도록 할 것
(논문에 실릴 수 있는 정확도가 약 86% 정도)

3. 파워포인트 또는 R 마크 다운 분석 보고서를 만들고 발표 (10~15분 발표, Q/A 시간)

카페의 7기 게시판 PPT 자료 참고
카페의 9기 게시판 R 마크다운 자료 참고

ppt에서는 코드를 반드시 넣을 필요는 없음 -> 코드 스크립트는 따로 제출함

R 마크 다운 -> 현업에서 분석보고서를 만들 때 자신만의 R 마크 다운 포맷을 가지고 있으면 금방 보고서를 만들 수 있음 (좋은 예시 : 9기 신소정님)

'과정 끝나고 제대로 만들어봐야지~' 라는 생각 -> 끝나고 절대 못만든다. 발표 있을 때 열심히 하자.

규칙 기반 알고리즘

2020년 6월 26일 금요일 오전 10:10

5장에 의사결정트리와 같이 분류 규칙을 둔 이유는 2개가 비슷한 알고리즘인데 **둘 다 정보획득량을 사용**한다.

공통점 : 정보획득량을 사용한다.
차이점 : **분류규칙이 훨씬 간단하면서 성능이 좋다**.

종류		알고리즘 기반
knn	------>	유클리드 거리공식
나이브베이즈	------>	나이브 베이즈 확률
의사결정트리	------>	정보획득량
분류규칙	------>	정보획득량

분류 규칙 알고리즘의 종류 2가지

2020년 6월 26일 금요일 오전 10:14

1. one R 알고리즘 :
   "하나의 사실(조건)만 가지고 간단하게 분류하는 알고리즘"
   하지만 하나의 사실만 가지고 분류하다 보니 간단하지만 오류가 많다.

예 : **가슴통증의 유무에 따라** 심장질환이 있는지 분류하고자 하면 가슴통증 하나만 보고 심장질환이 있다고 분류하기에는 오류가 많아진다. 왜냐하면 식도염, 폐질환 가슴통증이기 때문이다.

-> 가슴통증 유무 하나만 고려

1. Riper 알고리즘
   "복수 개의 사실(조건)을 가지고 분류하는 알고리즘"

예 : **가슴통증이 있으면서 호흡곤란이 있으면** 심장질환이다.

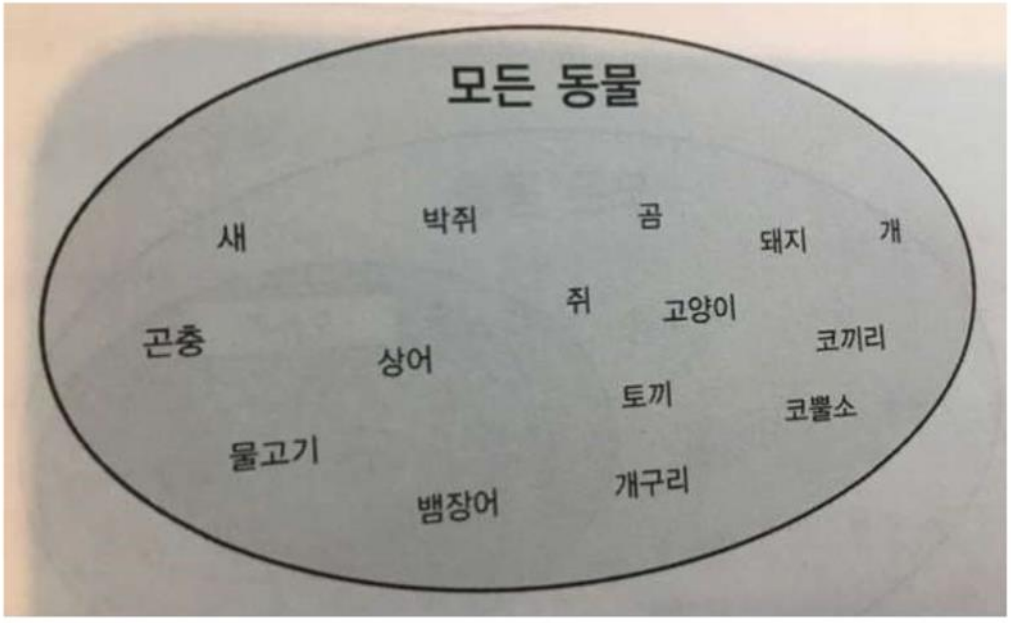
-> 가슴통증 + 호흡곤란 까지 고려

분류 규칙 알고리즘 PPT

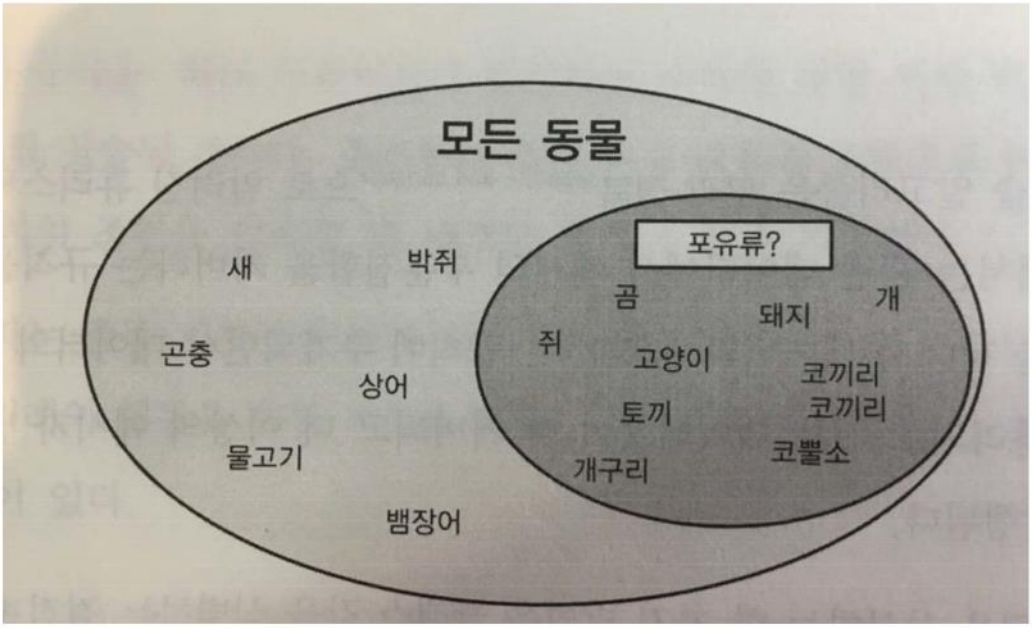
2020년 6월 26일 금요일 오전 10:19

규칙기반 ppt

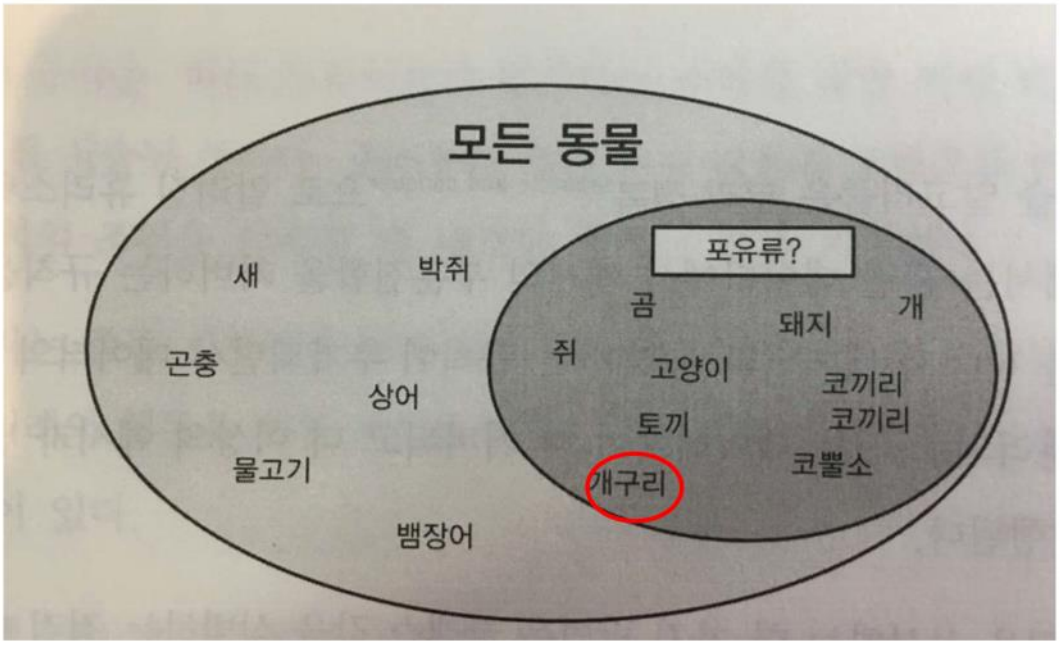
포유류인지 아닌지를 분류



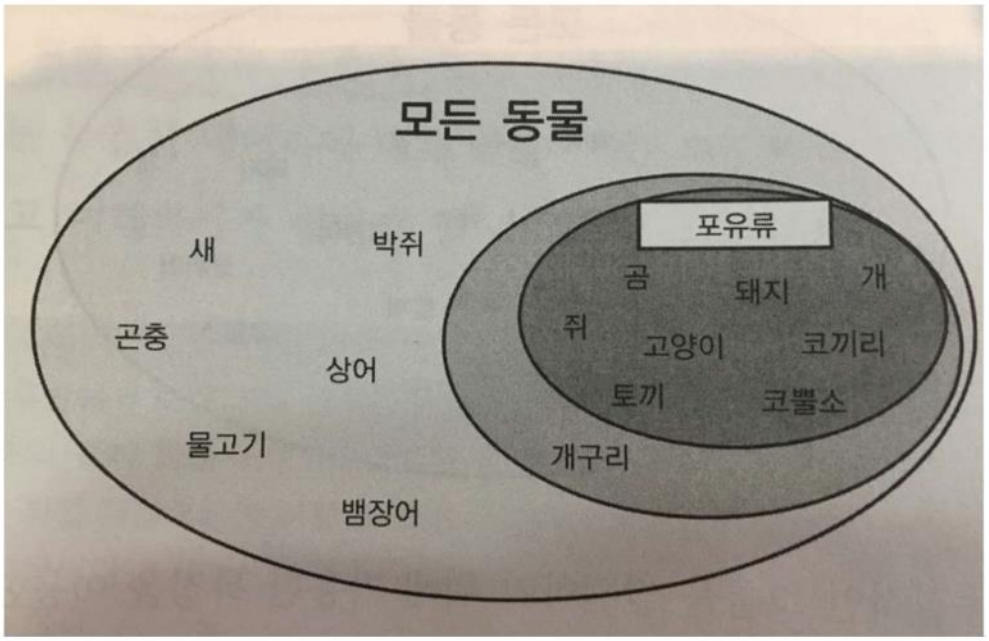
포유류인지 아닌지를 분류



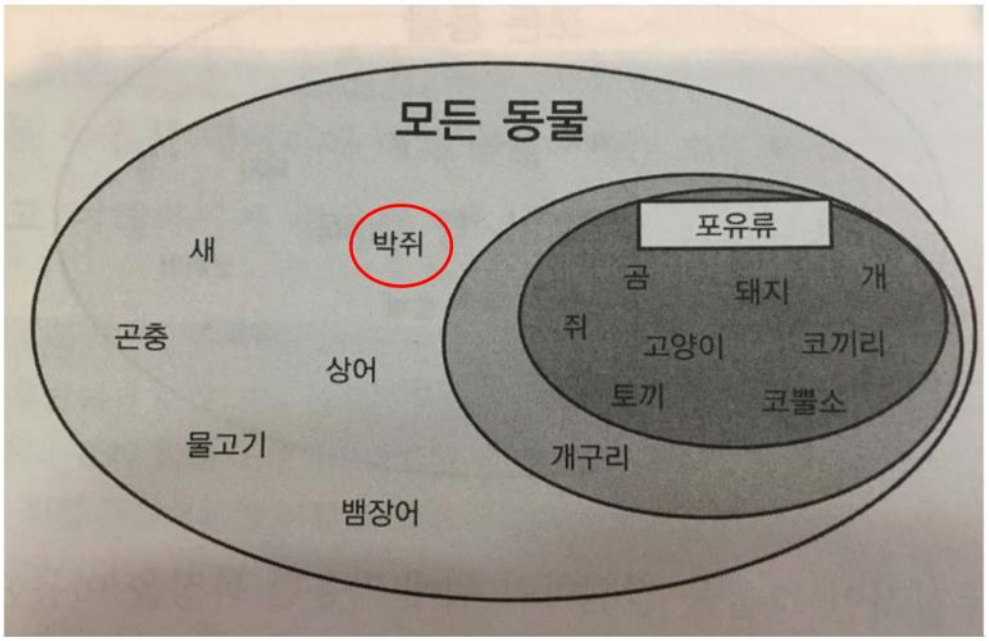
개구리는 양서류 아닌가 ?



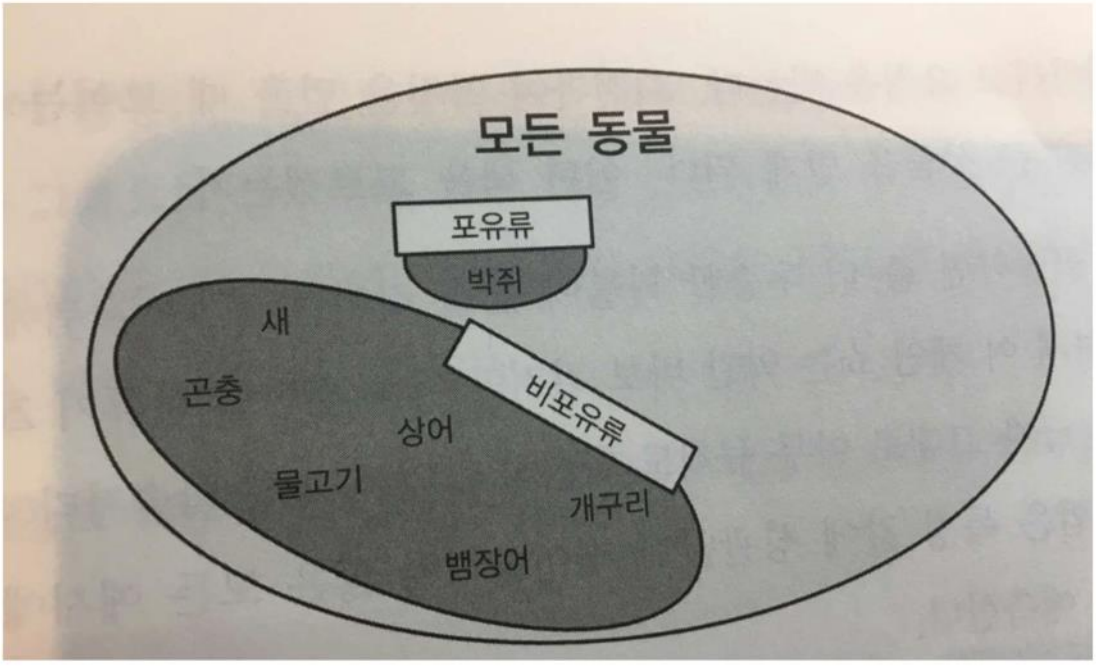
포유류는 땅으로 걷고 꼬리가 이어야 한다

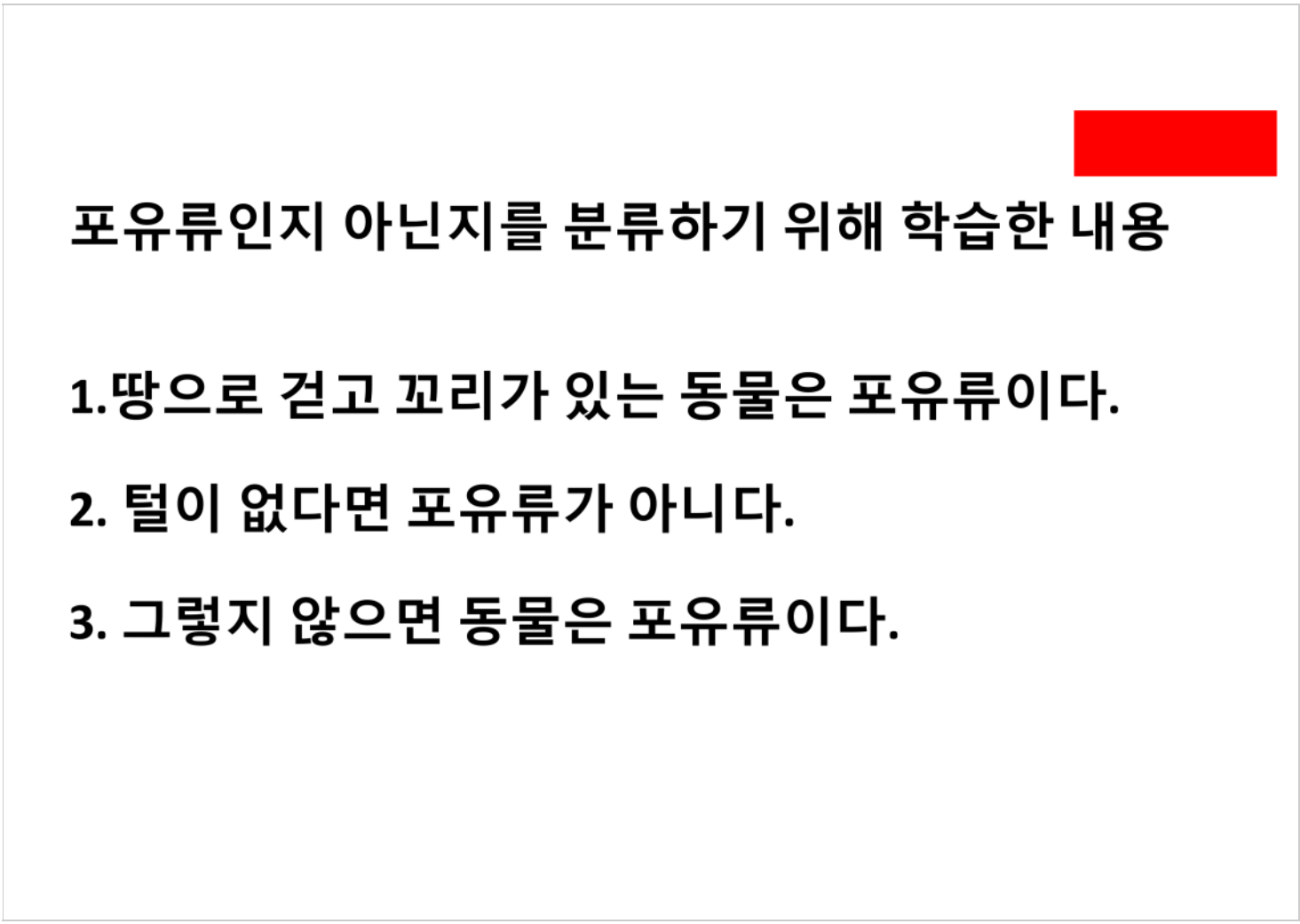


박쥐는 어떻게 할 것인가?



털이 없다면 포유류가 아니다 로 분류





-> 여러가지 사실(조건)을 가지고 분류를 하면 정확도가 많이 향상된다.

1. oneR 알고리즘 : p 223

"하나의 사실만 가지고 간단하게 분류하는 알고리즘"

간단하긴 하지만 오류가 많아진다.

예: 가슴통증의 유무에 따라 심장질환이 있는지 분류

가슴통증 하나만 보고 심장질환이 있다고 분류하기에는 오류가 많이진다. 왜냐하면 식도염, 폐질환도 가슴통증이 있기 때문이다.

2. Riper 알고리즘 : p 226

" 복수개의 사실(조건) 을 가지고 분류하는 알고리즘 "

예: 하늘을 날고 털이 있다면 그것은 포유류이다.
땅을 걷고 털이 있다면 그것은 포유류이다.

독버섯 데이터를 분류하는 실습

2020년 6월 26일 금요일 오전 10:48

1. 독버섯 데이터의 정보 획득량
2. 독버섯 데이터를 oneR 알고리즘으로 분류
3. 독버섯 데이터를 Riper 알고리즘으로 분류

문제 243. mushrooms.csv 파일을 R로 로드해서 각 컬럼변수의 정보획득량을 구하시오.

```
setwd('d:₩₩data')
mushrooms <- read.csv('mushrooms.csv', header = T, stringsAsFactors = T)
brary(FSelector)
weights <- information.gain(mushrooms)
weights
```

```
> weights
                          attr_importance
cap_shape                      0.033823296
cap_surface                    0.019817239
cap_color                      0.024987459
bruises                        0.133347298
odor                           0.628043316
gill_attachment                0.009818449
gill_spacing                   0.069926895
gill_size                      0.159530856
gill_color                     0.289026795
stalk_shape                    0.005210230
stalk_root                     0.093448465
stalk_surface_above_ring       0.197356746
stalk_surface_below_ring       0.188462888
stalk_color_above_ring         0.175952066
stalk_color_below_ring         0.167336519
veil_type                      0.000000000
veil_color                     0.016508698
ring_number                    0.026653359
ring_type                      0.220435714
spore_print_color              0.333199258
population                     0.139986632
habitat                        0.108708771
```

문제 244. 위의 결과가 정보 획득량이 높은것부터 출력되게 하시오.

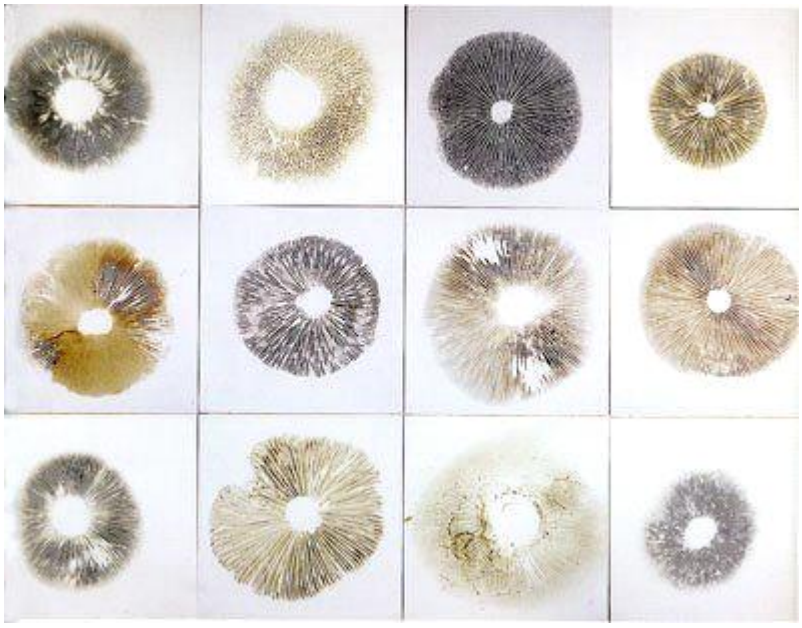
```
> str(weights)
'data.frame':    22 obs. of  1 variable:
 $ attr_importance: num  0.0338 0.0198 0.025 0.1333 0.628 ...
```

```
setwd('d:₩₩data')
mushrooms <- read.csv('mushrooms.csv', header = T, stringsAsFactors = T)
library(FSelector)
weights <- information.gain(mushrooms)
weights

library(doBy)
orderBy(~-attr_importance, weights)
```

```
> library(doBy)
> orderBy(~-attr_importance, weights)
                     attr_importance
odor                      0.628043316
spore_print_color         0.333199258
gill_color                0.289026795
ring_type                 0.220435714
stalk_surface_above_ring  0.197356746
stalk_surface_below_ring  0.188462888
stalk_color_above_ring    0.175952066
stalk_color_below_ring    0.167336519
gill_size                 0.159530856
population                0.139986632
bruises                   0.133347298
habitat                   0.108708771
stalk_root                0.093448465
gill_spacing              0.069926895
cap_shape                 0.033823296
ring_number               0.026653359
cap_color                 0.024987459
cap_surface               0.019817239
veil_color                0.016508698
gill_attachment           0.009818449
stalk_shape               0.005210230
veil_type                 0.000000000
```

-> ODOR, 독버섯을 분류할 때 버섯의 향기가 큰 영향을 준다는 것을 알 수 있음

※ 설명 : 독버섯 데이터에서 정보획득량이 가장 높은 것은 odor(버섯향)이 가장 크고, spore_print_color, grill_color 순으로 나타남



-> The spore print is the powdery deposit obtained by allowing spores of a fungal fruit body to fall onto a surface underneath.

규칙 기반 분류 알고리즘 (oneR 실습)

2020년 6월 26일 금요일 오전 11:01

" 독버섯 데이터 "

1. 버섯 데이터를 R 로 로드한다.

```
mushroom <- read.csv("mushrooms.csv", stringsAsFactors=T)
```

2. mushroom 데이터를 훈련 데이터와 테스트 데이터로 나눈다
(훈련 데이터 75%, 테스트 데이터 25%)

```
set.seed(11)

dim(mushroom)

train_cnt <- round( 0.75 * dim(mushroom)[1])
train_index <- sample(1:dim(mushroom)[1], train_cnt, replace=F)

mushroom_train <- mushroom[train_index,  ]
mushroom_test  <- mushroom[-train_index, ]
```

3. 규칙기반 알고리즘인 oneR 을 이용해서 독버섯과 일반버섯을 분류하는 모델을 생성한다.

```
install.packages("OneR")
library(OneR)

model1 <- OneR(type~. ,  data=mushroom_train)

model1
summary(model1)
```

```
> model1    # 분류를 위해 어떻게 코드를 만들었는지 보여준다.
Call:
OneR.formula(formula = type ~ ., data = mushroom_train)
Rules:
If odor = almond   then type = edible
If odor = anise    then type = edible
If odor = creosote then type = poisonous
If odor = fishy    then type = poisonous
If odor = foul     then type = poisonous
If odor = musty    then type = poisonous
If odor = none     then type = edible
If odor = pungent  then type = poisonous
```

```
If odor = spicy    then type = poisonous
Accuracy:
6001 of 6093 instances classified correctly (98.49%)
```

-> 모델을 확인해보면 if문을 만들어서 모델을 구축했음을 알 수 있다.

```
> summary(model1)
Contingency table:
          odor
type       almond anise creosote fishy   foul musty   none pungent spicy  Sum
  edible     * 303 * 295       0     0      0     0 * 2527       0     0 3125
  poisonous      0     0    * 134 * 442 * 1639  * 24     92   * 196 * 441 2968
  Sum          303   295      134   442   1639    24   2619     196   441 6093
---
Maximum in each column: '*'
Pearson's Chi-squared test:
X-squared = 5737.7, df = 8, p-value < 2.2e-16
```

4. 위에서 생성한 모델을 가지고 테스트 데이터로 결과를 확인한다.

```
result1 <- predict( model1, mushroom_test[   , -1] )

library(gmodels)

CrossTable( mushroom_test[ , 1],  result1)
```

```
> CrossTable( mushroom_test[ , 1],  result1)
Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|
Total Observations in Table:  2031
| result1
mushroom_test[, 1] |    edible | poisonous | Row Total |
-------------------|-----------|-----------|-----------|
            edible |      1083 |         0 |      1083 |
                   |   406.238 |   490.576 |           |
                   |     1.000 |     0.000 |     0.533 |
                   |     0.975 |     0.000 |           |
                   |     0.533 |     0.000 |           |
-------------------|-----------|-----------|-----------|
         poisonous |        28 |       920 |       948 |
                   |   464.088 |   560.437 |           |
                   |     0.030 |     0.970 |     0.467 |
                   |     0.025 |     1.000 |           |
                   |     0.014 |     0.453 |           |
```

```
-------------------|-----------|-----------|-----------|
      Column Total |      1111 |       920 |      2031 |
                   |     0.547 |     0.453 |           |
-------------------|-----------|-----------|-----------|
```

-> 결론 : 머신러닝 모델이 식용으로 예측했는데 **독버섯인 것이 26개** 있다.

From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SZTZ&datanum=2083>

규칙 기반 분류 알고리즘 (JRip 실습)

2020년 6월 26일 금요일 오전 11:13

```
install.packages("RWeka")
library(RWeka)
model2 <-  JRip(type~ ., data=mushroom_train)
model2
```

```
> model2 <-  JRip(type~ ., data=mushroom_train)
> model2
JRIP rules:
===========
(odor = foul) => type=poisonous (1639.0/0.0)
(gill_size = narrow) and (gill_color = buff) => type=poisonous (883.0/0.0)
(gill_size = narrow) and (odor = pungent) => type=poisonous (196.0/0.0)
(odor = creosote) => type=poisonous (134.0/0.0)
(spore_print_color = green) => type=poisonous (56.0/0.0)
(stalk_surface_below_ring = scaly) and (stalk_surface_above_ring = silky) => type=poisonous
(48.0/0.0)
(habitat = leaves) and (cap_surface = scaly) and (population = clustered) => type=poisonous
(10.0/0.0)
(cap_surface = grooves) => type=poisonous (2.0/0.0)
 => type=edible (3125.0/0.0)
Number of Rules : 9
```

-> oneR과는 다르게 ***JRip는 조건이 훨씬 복잡****해졌음*

```
summary(model2)
```

```
> summary(model2)  # 작은 이원교차표가 하나 보임
=== Summary ===
Correctly Classified Instances        6093               100      %
Incorrectly Classified Instances         0                0      %
Kappa statistic                          1
Mean absolute error                      0
Root mean squared error                  0
Relative absolute error                  0      %
Root relative squared error              0      %
Total Number of Instances             6093
=== Confusion Matrix ===
a    b   <-- classified as
 3125    0 |    a = edible
    0 2968 |    b = poisonous
```

```
result2 <- predict( model2, mushroom_test[   , -1] )
library(gmodels)
```

CrossTable(mushroom_test[, 1], result2)

```
> result2 <- predict( model2, mushroom_test[    , -1] )
> library(gmodels)
> CrossTable( mushroom_test[ , 1],  result2)
Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|
Total Observations in Table:  2031
| result2
mushroom_test[, 1] |    edible | poisonous | Row Total |
-------------------|-----------|-----------|-----------|
         edible |      1083 |         0 |      1083 |
                |   442.493 |   505.507 |           |
                |     1.000 |     0.000 |     0.533 |
                |     1.000 |     0.000 |           |
                |     0.533 |     0.000 |           |
-------------------|-----------|-----------|-----------|
       poisonous |         0 |       948 |       948 |
                |   505.507 |   577.493 |           |
                |     0.000 |     1.000 |     0.467 |
                |     0.000 |     1.000 |           |
                |     0.000 |     0.467 |           |
-------------------|-----------|-----------|-----------|
    Column Total |      1083 |       948 |      2031 |
                |     0.533 |     0.467 |           |
-------------------|-----------|-----------|-----------|
```

-> 정확도가 거의 100% 그래서 굉장히 강력한 알고리즘이다

From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SZTZ&datanum=2084>

자동화 코드 만들기

2020년 6월 26일 금요일 오전 11:42

문제 245. 머신러닝을 활용하는 자동화 코드가 잘 수행되는지 확인하시오.

r()

```
> setwd('d:₩₩data')
> r <- function() {source('my_func.R')}
> r()
data.table 1.12.8 using 4 threads (see ?getDTthreads).  Latest news: r-datatable.com
숫자를 선택하세요 ~
1: 산포도 그래프
2: 히스토그램 그래프
3: 사분위수 그래프
4: 유방암 진단
5: 독감 진단
```

문제 246. mushrooms_test1.csv 과 mushrooms_test2.csv 라는 이름으로 독버섯과 식용버섯 데이터를 각각 만드시오.

type,cap_shape,cap_surface,cap_color,bruises,odor,gill_attachment,gill_spacing,gill_size,gill_color,stalk_shape,stalk_root,stalk_surface_above_ring,stalk_surface_below_ring,stalk_color_above_ring,stalk_color_below_ring,veil_type,veil_color,ring_number,ring_type,spore_print_color,population,habitat
poisonous,convex,smooth,brown,yes,pungent,free,close,narrow,black,enlarging,equal,smooth,smooth,white,white,partial,white,one,pendant,black,scattered,urban

-> 컬럼명과 같이 복사한다.

type,cap_shape,cap_surface,cap_color,bruises,odor,gill_attachment,gill_spacing,gill_size,gill_color,stalk_shape,stalk_root,stalk_surface_above_ring,stalk_surface_below_ring,stalk_color_above_ring,stalk_color_below_ring,veil_type,veil_color,ring_number,ring_type,spore_print_color,population,habitat
edible,convex,smooth,yellow,yes,almond,free,close,broad,black,enlarging,club,smooth,smooth,white,white,partial,white,one,pendant,brown,numerous,grasses

-> 독버섯, 식용버섯 각각 하나씩 데이터를 만든다.

mushrooms_test1.csv -> 독버섯
mushrooms_test2.csv -> 식용버섯

문제 247. 독버섯에 대한 정확도 100%가 나오는 jriper 실습 코드를 가져와서 위의 테스트 csv 파일을 로드해서 예측하고 결과가 나오게 코드를 수정하시오.

<수정 전>
```
install.packages("RWeka")
library(RWeka)
model2 <-  JRip(type~ ., data=mushroom_train)
result2 <- predict( model2, mushroom_test[   , -1] )
library(gmodels)
CrossTable( mushroom_test[ , 1],  result2)
```

<수정 후>
```
# install.packages("RWeka")
library(RWeka)

# 모델 생성
model2 <-  JRip(type~ ., data=mushroom_train)

# 파일 로드
fname <- file.choose()    # 윈도우 탐색기 여는 코드를 추가
table <- read.csv(fname, header=T, stringsAsFactor=F )

# 결과 예측
result2 <- predict( model2, table[ , -1] )
result2
```

-> 그런데 독버섯을 식용버섯으로 잘못 예측하는 오류 발생
(다음주에 선생님이 다시 알려주실 예정)

문제 248. 문제 247번에 구현한 코드를 Riper_func 라는 함수로 생성하시오.

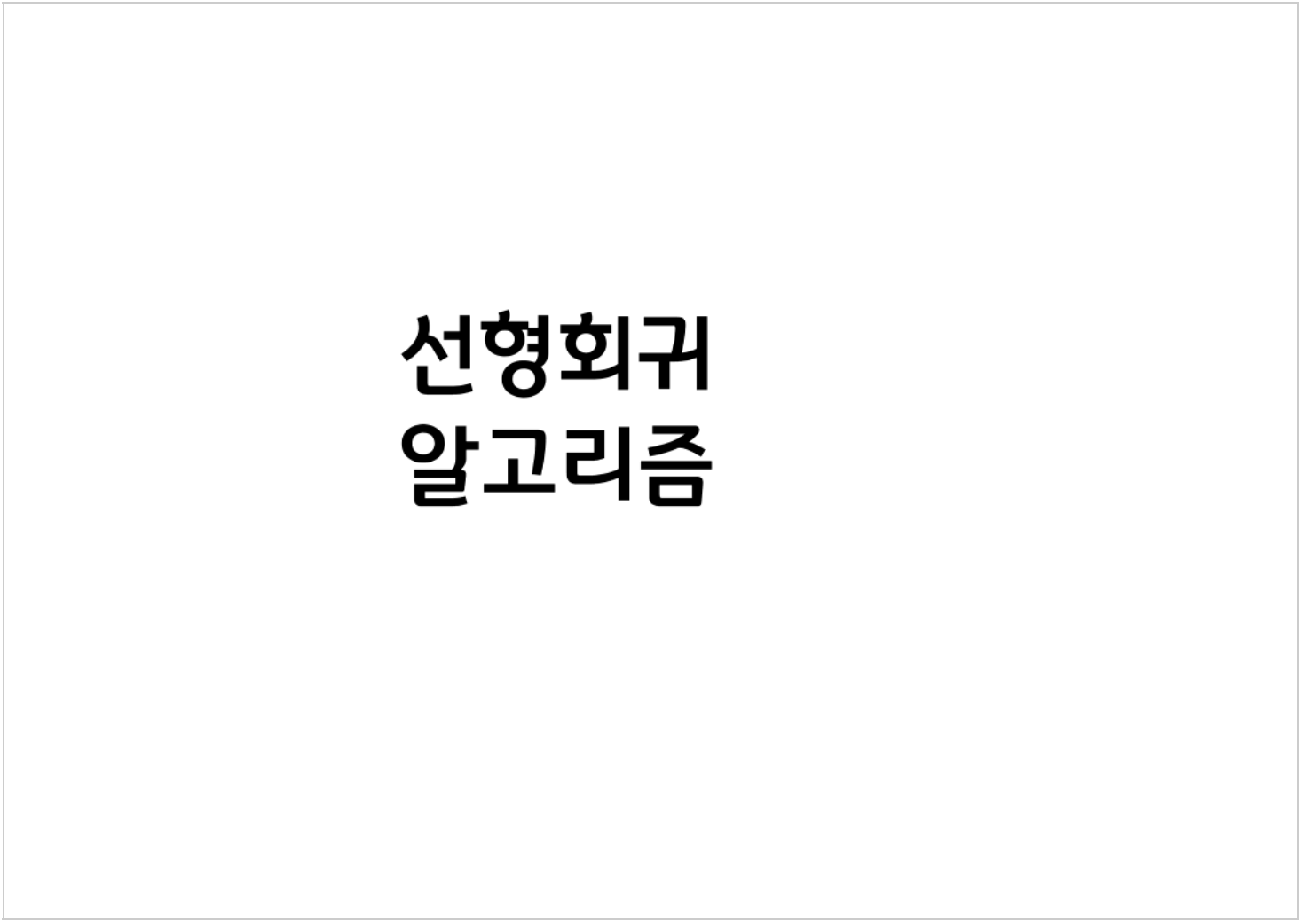
```
Riper_func()
```

결과 : 식용입니다.

회귀분석 PPT

2020년 6월 26일 금요일 오후 1:45

회귀분석 알고리즘



회귀의 사전적 뜻은 ?

회귀, 回歸

/회-/훼-/

명사

한 바퀴 돌아서 본디의 자리나 상태로 돌아오는 것. 순화어는 `돌아옴'.

돌 회, 돌아갈 귀

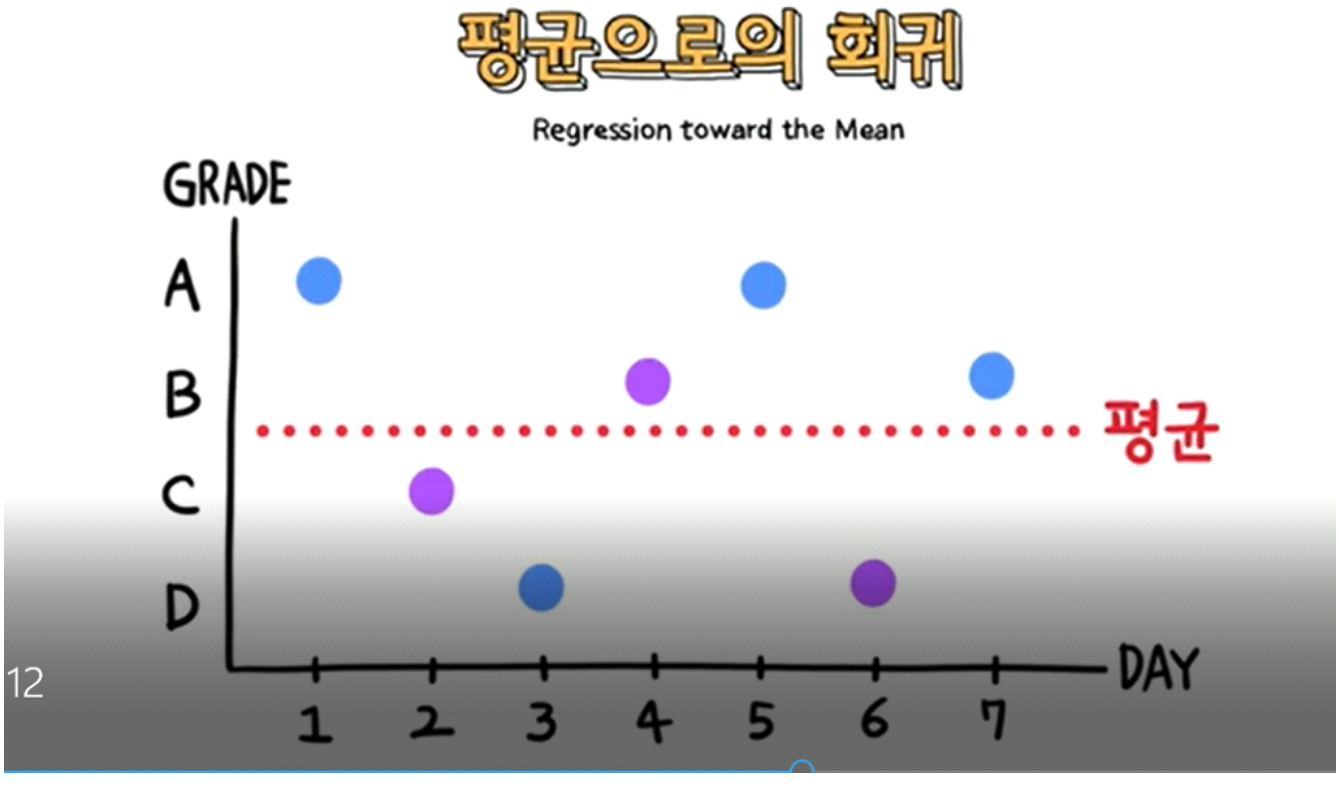
그런데 한자의 뜻과는 틀리게
그냥 무엇인가 예측하는 기법이다

데이터의 새로운 표본에는
평균으로 돌아가려는 특성이 있는데,
이 돌아가려는 특성만 있을뿐이고
이 돌아가려는 특성을 분석하지는 않고
그냥 무엇인가를 예측하는 기법입니다

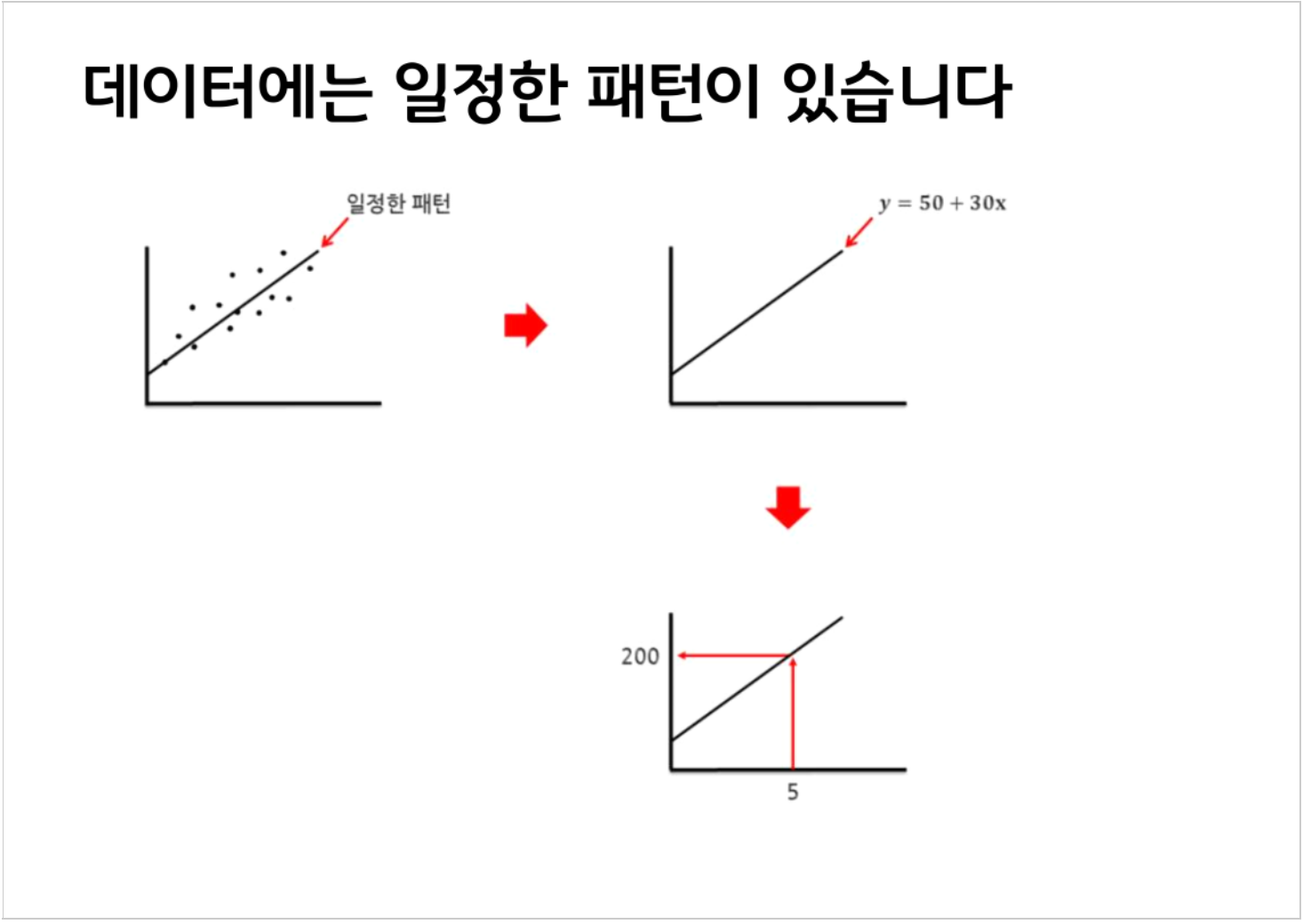
영상을 하나 보시겠습니다

'2년 차 징크스'는 왜 생기는 걸까?

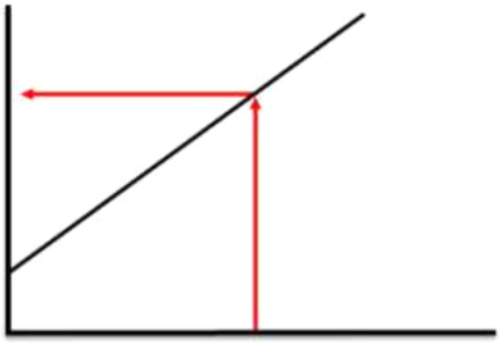
세상의 모든 법칙 - '2년 차 징크스'는 왜 생기는 걸까



From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SeFi&contentval=0004hzzzzzzzzzzzzzzzzzzzzzzz&datanum=291&page=1&prev_page=0&firstbbsdepth=&lastbbsdepth=zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz&listnum=20>

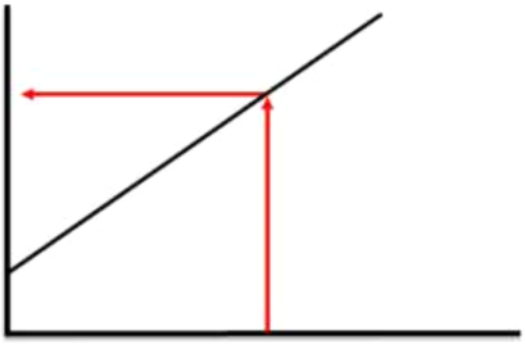


2. 필요한 전력량은
얼마인지?



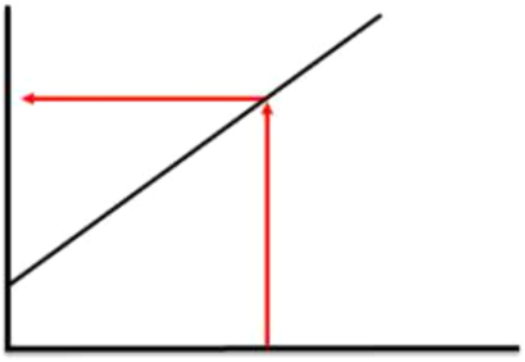
1. 인구수가 x명일 때,

2. 몸무게는
얼마인지?



1. 키가 x일 때,

2. 암 발생률은
얼마인지?



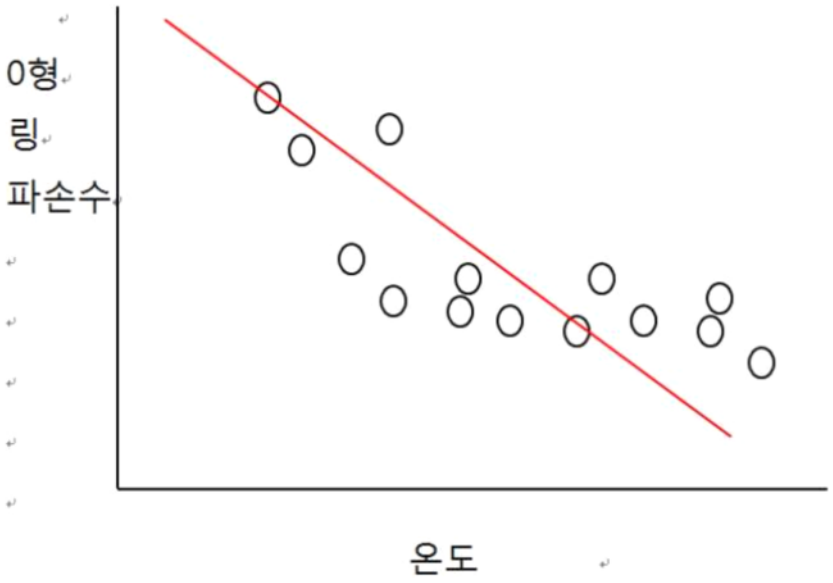
1. 흡연율이 x일 때,

우주 왕복선 챌린져호의 폭파원인에 대한 데이터 분석이 있었습니다

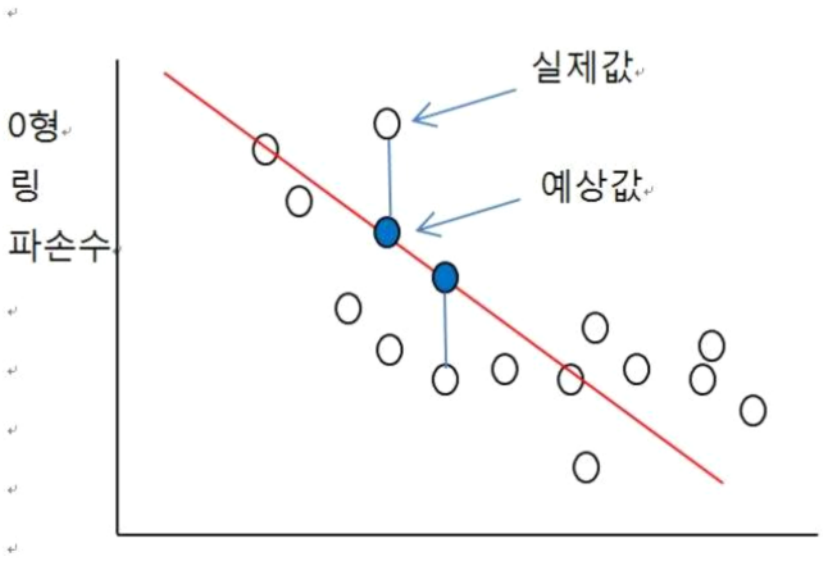
https://www.youtube.com/watch?v=rCVX9eXvgRo&feature=youtu.be

우주 왕복선 챌린져호의 폭파원인은 o형링의 파손이었습니다

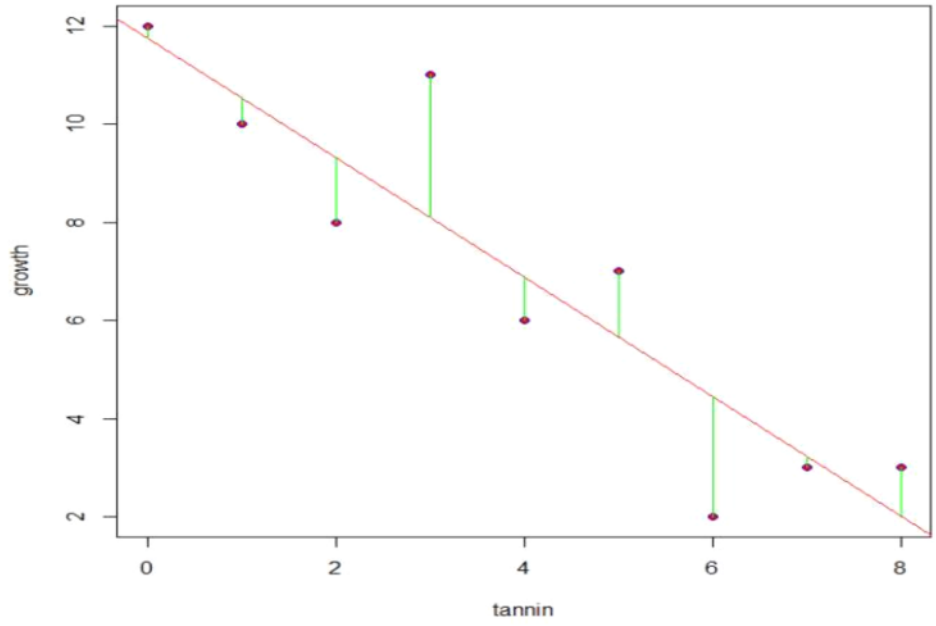
온도가 낮아질수록 O형링의 파손수가 높아집니다



빨간 직선이 각 값들의 평균을 나타내는 회귀직선입니다



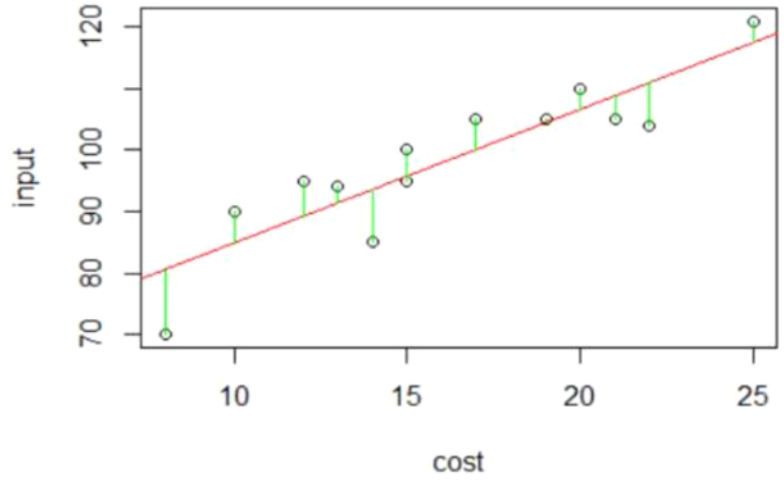
회귀 직선 공식은
사료회사에서도 필요합니다



-> 탄닌 함유량이 얼마나 되어야 에벌레의 성장이 증가하는지에 대한 관계

회귀 직선 공식은
광고회사에서도 필요합니다

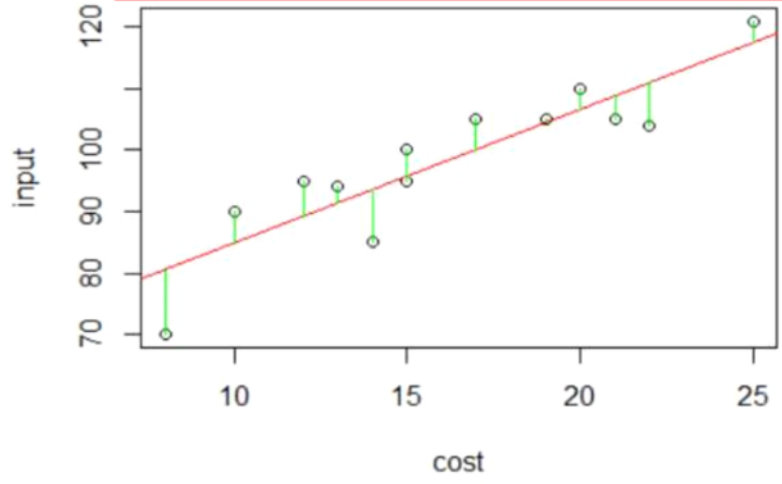
$$input = 2.18649 \times cost + 62.92913$$



-> 광고 비용을 얼마나 들였을 때 매출이 얼마나 늘어나는지에 대한 관계

우리는 이 회귀직선의 방정식을 알아내야 합니다

$$input = 2.18649 \times cost + 62.92913$$



6장. 회귀분석

2020년 6월 26일 금요일 오후 2:03

회귀분석은 **하나의 변수가 나머지 다른 변수들과의 선형관계를 갖는가의 여부를 분석하는 방법**으로 하나의 종속변수 (예측하고자 하는 값)와 독립변수 사이의 관계를 명시하는 것이다.

예시 : 집값에 가장 영향을 주는 요소가 무엇인가?

- 독립변수 : 종속변수에 영향을 주는 변수 (아파트 평수, 역세권 여부, 학군 등)
- 종속변수 : 집값

회귀식 : y = ax + b
-> 직선의 방정식으로 표현할 수 있다.
우리는 기울기인 a와 절편인 b를 알아내야 한다.

***최소 제곱 추정법**을 이용해서 알아낸다. (257p - 보통 최소 제곱 추정 참고)*

최소 제곱 추정법

2020년 6월 26일 금요일 오후 2:07

최적의 a(기울기)와 b(절편)을 결정하기 위해서는 **정규 최소제곱으로 알려진 추정기법을 사용**한다.

실제값과 예측값 사이의 수직 직선인 **오차(잔차)를 제곱해서 구한 총합을 알아야**한다.

회귀선 예측은 실제값보다 잔차만큼 차이가 난다.

문제 249. 어느 실험실에서 10시간, 20시간, 30시간, 40시간마다 물질의 방사능 수치를 측정한 자료가 있을 때 35시간에 물질의 방사능 수치는?

x축 : 시간
y축 : 방사능 수치

```
x = c(10, 20, 30, 40)
y = c(300, 250, 200, 150)

func_nuclear(35)    # 175
```

y = ax + b 에서
175 = 35 a + b 이다.

여기서 a와 b값만 구하면 된다.

$$a = \frac{\mathrm{cov}(x, y)}{\mathrm{var}(x)}$$

a는 공분산값을 x에 대한 분산값으로 나누면 구할 수 있다.

```
x = c(10, 20, 30, 40)
y = c(300, 250, 200, 150)
a = cov(x,y)/var(x)
a     # -5
```

```
> a = cov(x,y)/var(x)
> a
[1] -5
```

y = -5x + b

b는 x의 평균을 구하고, y의 평균을 구해서 위의 식에 대입하여 구해본다.

```
x_mean = mean(x)
y_mean = mean(y)
b = y_mean - a*x_mean
```

회귀분석 실습 1

2020년 6월 26일 금요일 오후 2:46

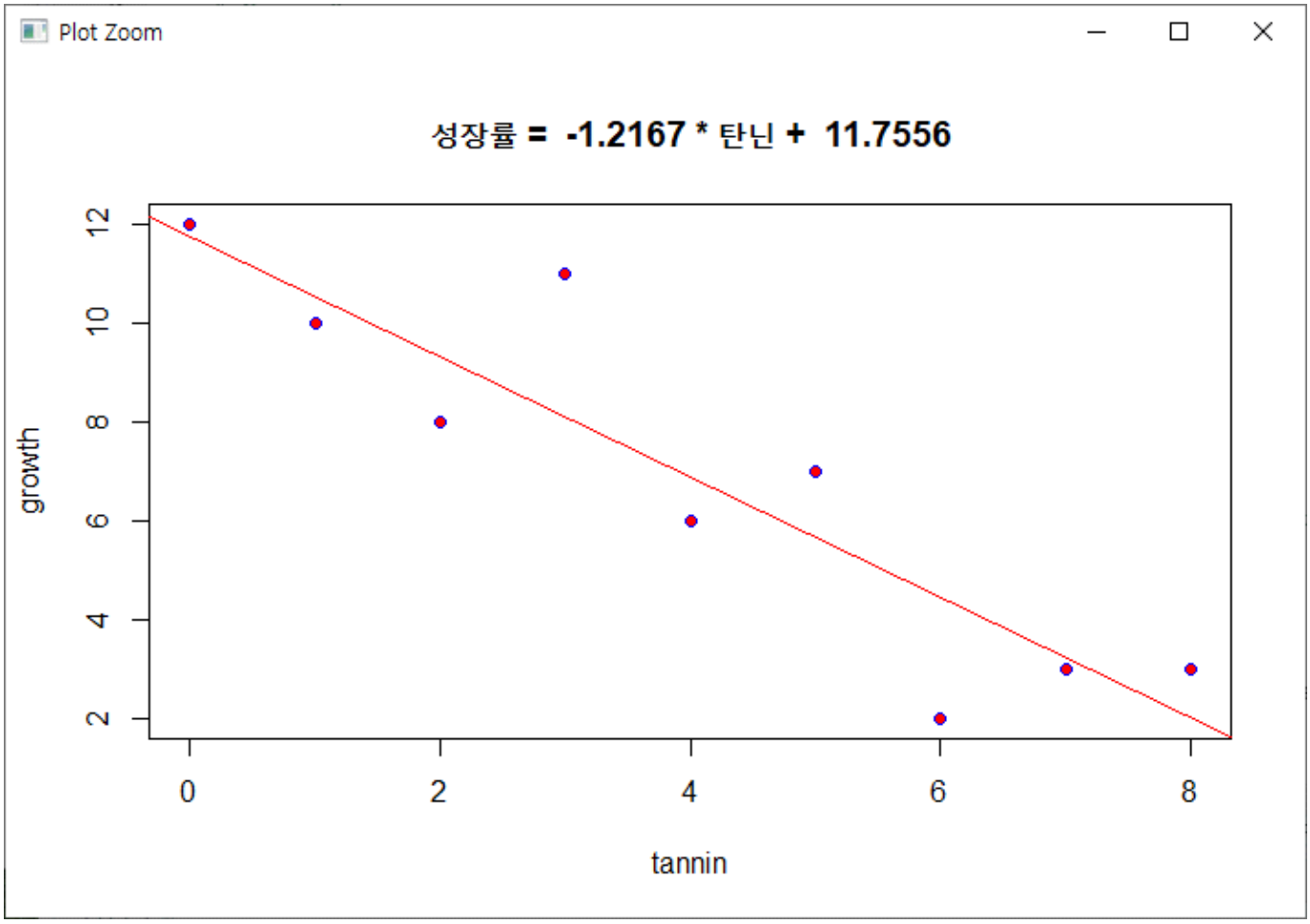
■ 애벌래의 성장 추이와 탄닌과의 관계

■ 애벌래의 성장 추이와 탄닌과의 관계가 어떻게 되는지 탄닌 포함량이 많을수록 애벌래의 성장이 증가하는지 감소하는지를 나타내는 회귀 방정식을 구하고 시각화 하시오

```
# 1. 데이터를 로드한다.
reg <- read.table("regression.txt", header=T)
# reg
# growth        tannin
    # 12            0
    # 10            1
    # 8             2
    # 11            3
    # 6             4
    # 7             5
    # 2             6
    # 3             7
    # 3             8

# 데이터설명 : 에벌레 사료의 탄닌(영양제)의 양에 따라
            # 애벌레의 성장률이 어떻게 되는지 조사한 데이터이다.
            # 맛이 없어서 잘 안먹어서 성장이 떨어지는 결과가 나타난다.

# 2. 데이터를 시각화 한다.
attach(reg)
plot(growth~tannin, data = reg, pch=21, col='blue', bg='red')    # 그래프 생성
       # y ~ x

# 3. 회귀분석을 해서 회귀 계수인 기울기와 절편을 구하시오.
m <- lm (  growth ~  tannin, data=reg)
    # ↑        ↑         ↑
 # 회귀함수 종속변수 독립변수
# m    # intercept = 절편, tannin = 기울기    # y = -1.217x + 11.756 이라는 방정식을 구할 수 있다.

# 4. 위의 산포도 그래프에 회귀 직선을 그린다.
abline(m, col='red')
```

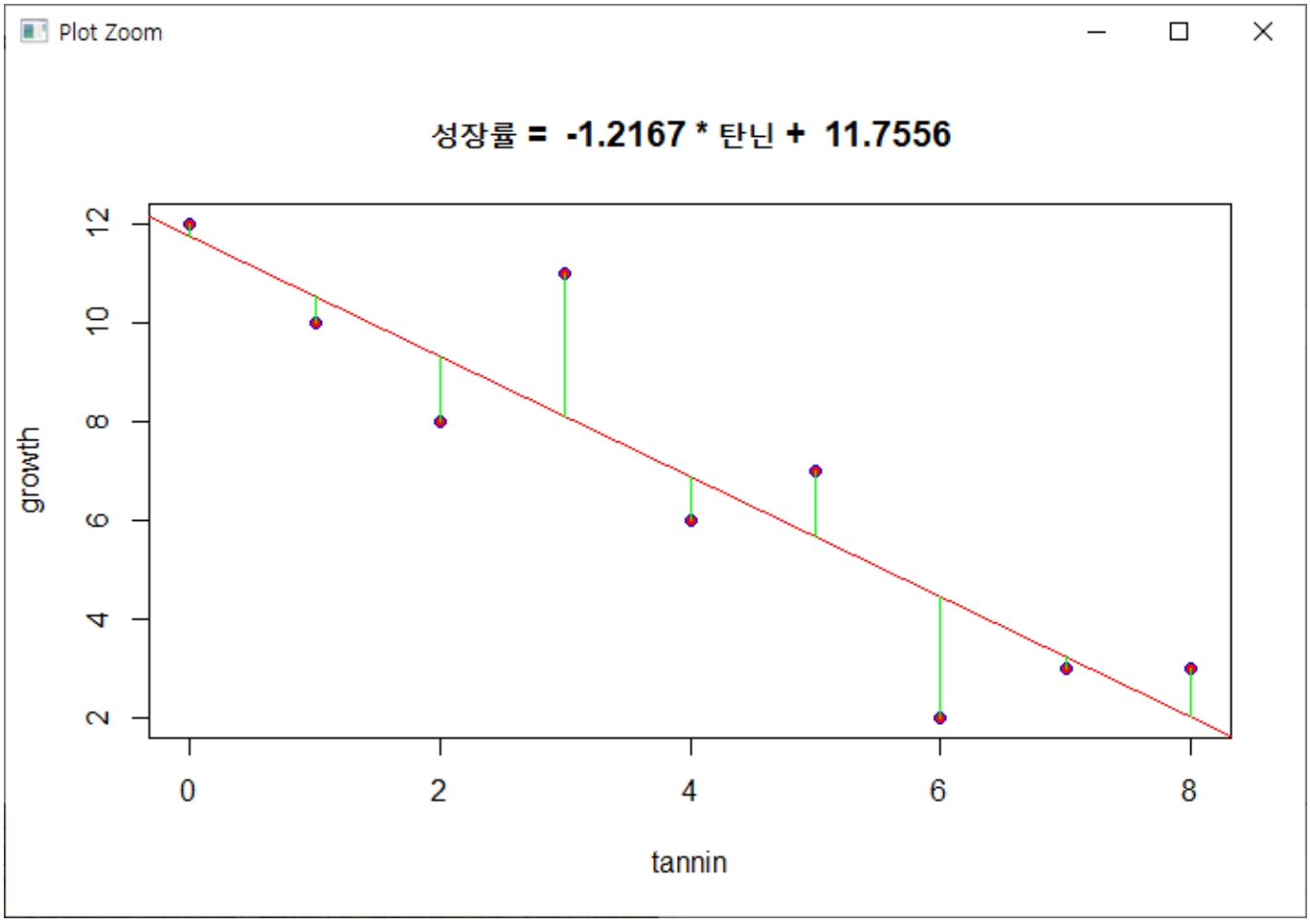
5. 그래프 제목을 회귀 직선의 방정식으로 출력되게 한다.

```
title(paste( '성장률 = ', round(m$coefficients[2], 4), "* 탄닌 + ", round(m$coefficients[1], 4)))
```



6. 위의 그래프에 잔차를 그린다.

```
y_hat <- predict(m, tannin=tannin)
y_hat
> y_hat
       1        2        3        4        5        6        7        8        9
11.755556 10.538889  9.322222  8.105556  6.888889  5.672222  4.455556  3.238889  2.022222

join <- function(i)
lines( c(tannin[i], tannin[i]), c( growth[i],y_hat[i]), col="green")
sapply(1:9, join)
```



From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SZTZ&datanum=2086>

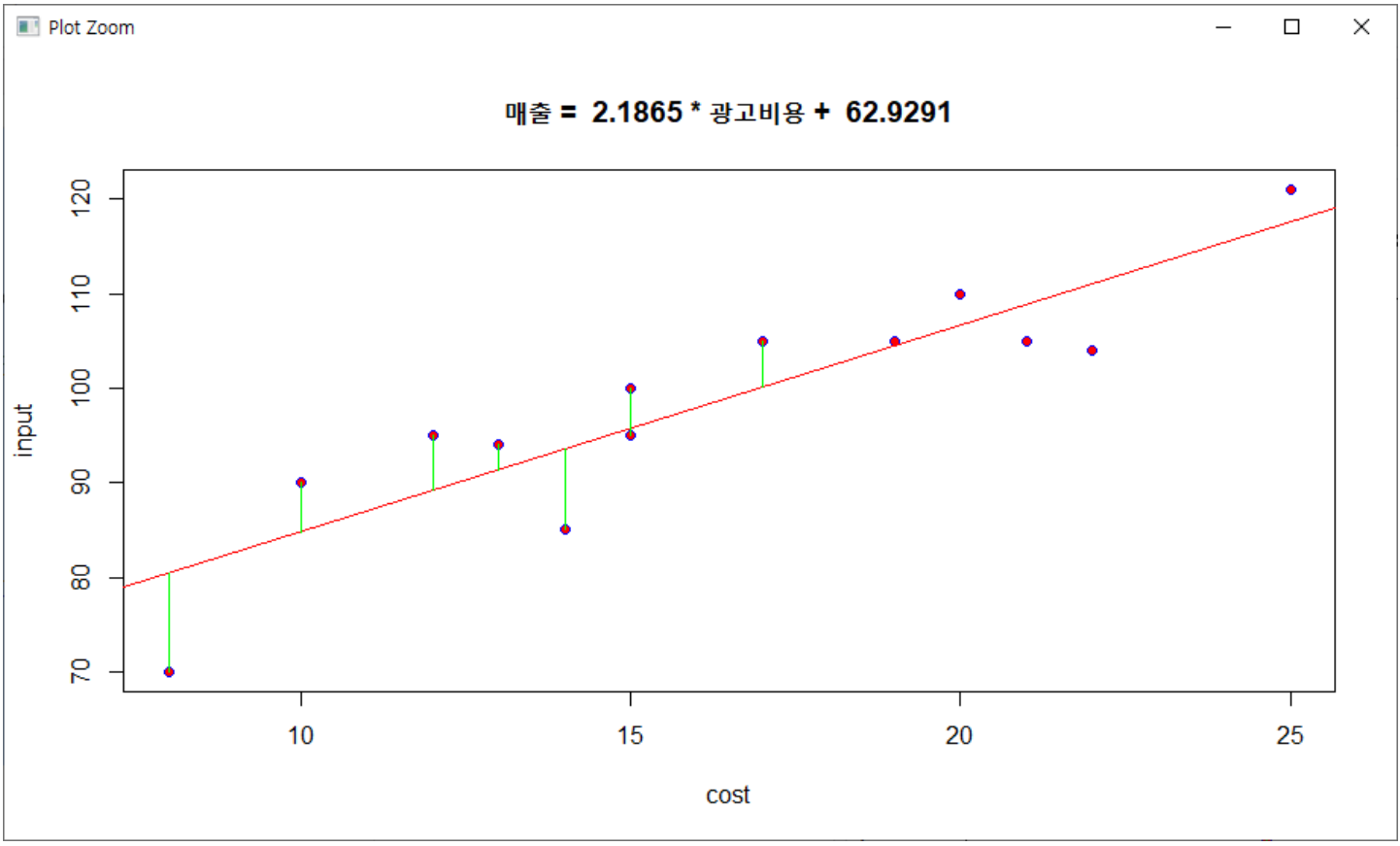
회귀분석 실습 2

2020년 6월 26일 금요일 오후 3:04

문제 250. 광고비가 매출에 미치는 영향에 대해 회귀식과 그래프를 시각화하시오.

데이터 : simple_hg.csv

```
# 1. 데이터를 로드한다.
reg <- read.csv("simple_hg.csv", header=T)
# 데이터설명 : 연도별 광고 비용과 매출

# 2. 데이터를 시각화 한다.
attach(reg)
reg
plot(input~cost, data = reg, pch=21, col='blue', bg='red')    # 그래프 생성
      # y ~ x

# 3. 회귀분석을 해서 회귀 계수인 기울기와 절편을 구하시오.
m <- lm (  input ~  cost, data=reg)
   # ↑        ↑         ↑
 # 회귀함수 종속변수 독립변수
# m    # intercept = 절편, tannin = 기울기    # y = -1.217x + 11.756 이라는 방정식을 구
할 수 있다.

# 4. 위의 산포도 그래프에 회귀 직선을 그린다.
abline(m, col='red')

# 5. 그래프 제목을 회귀 직선의 방정식으로 출력되게 한다.
title(paste( '매출 = ', round(m$coefficients[2], 4), "* 광고비용 + ", round(m$coefficients[1],
4)))

# 6. 위의 그래프에 잔차를 그린다.
y_hat <- predict(m, cost=cost)
y_hat
join <- function(i)
lines( c(cost[i], cost[i]), c( input[i],y_hat[i]), col="green")
sapply(1:9, join)
```



답 :

```
launch2 <- read.csv("simple_hg.csv")

attach(launch2)

plot(input ~ cost, data=launch2, pch=21, col='red', bg='red')

m <- lm(input  ~  cost, launch2)

abline(m, col='red')

title(expression(italic(input== 2.18649 %*%cost + 62.92913)))

yhat <- predict(m, cost=cost)

join <- function(i)

  lines(c(cost[i],cost[i]),c(input[i],yhat[i]), col="green")

sapply(1:19,join)
```

From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SeFi&contentval=0004kzzzzzzzzzzzzzzzzzzzzzzzzz&datanum=294&page=1&prev_page=0&firstbbsdepth=&lastbbsdepth=zzzzzzzzzzzzzzzzzzzzzzzzzzzzz&listnum=20>

회귀분석 실습 3

2020년 6월 26일 금요일 오후 3:16

■ 우주왕복선 챌린져호의 폭파원인 분석

4기 학생 -> 삼성갤럭시 노트 7 폭파원인 분석 (마찬가지로 회귀)

■ 챌린져호의 폭파원인을 분석하기 위한 회귀직선의 기울기를 R 로 알아내시오 !

"발사 온도에 대한 o 형링의 파손이 원인 "

y = ax + b

y 가 o형링의 파손수, x 가 발사 온도

여기서 회귀모수인 a,b 를 기계가 구하도록 해야한다.

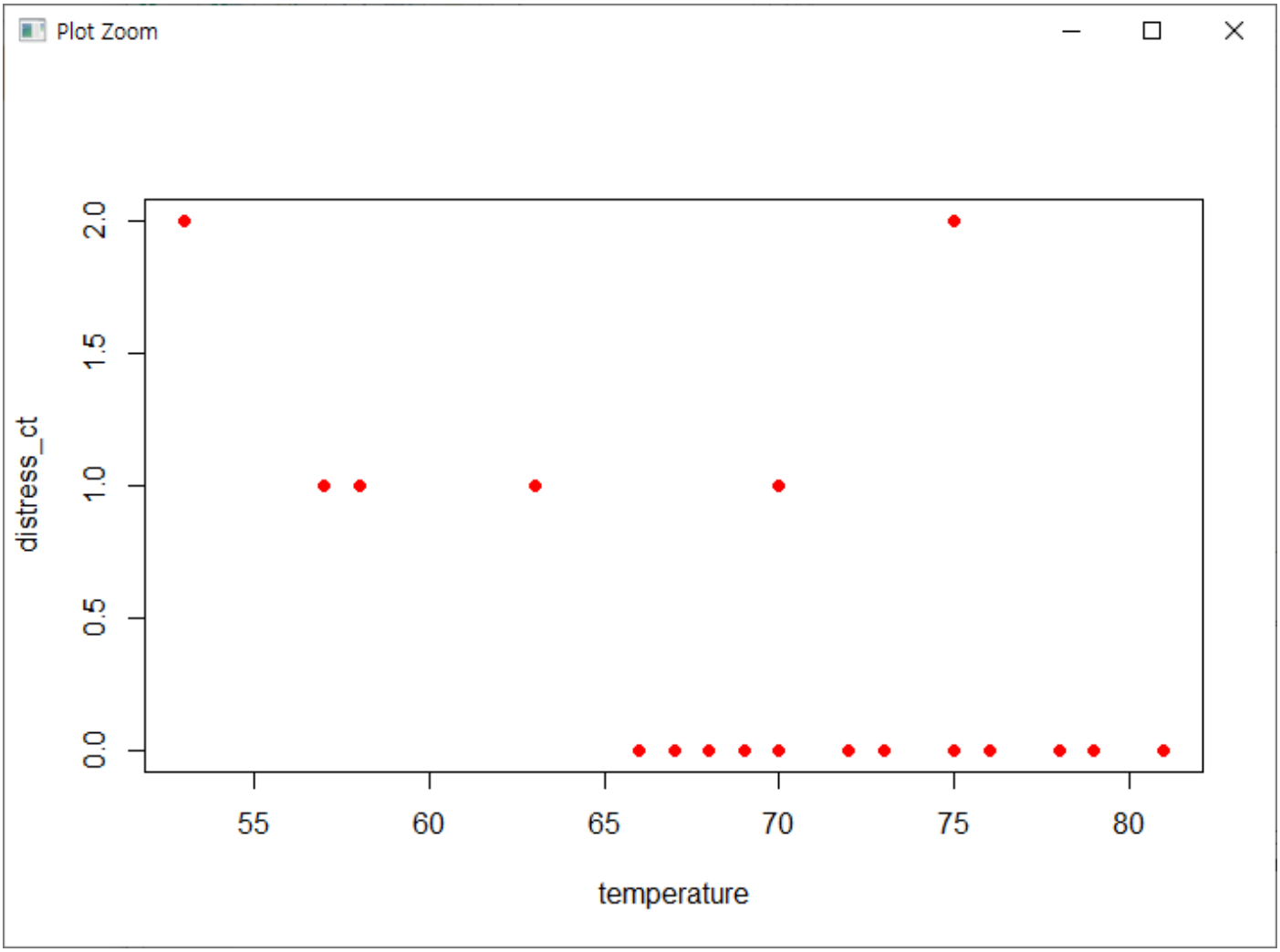
```
setwd("d:₩₩data")

# 1. 데이터를 로드한다.
launch <- read.csv("challenger.csv")
    # launch    # 데이터에 대한 설명명
    # distress_ct :  O형링 파손수
    # temperature :  온도
    # field_check_pressure : 압력
    # flight_num  : 비행기 번호    # 오래된 비행기일수록 고장이 잘 난다.

# 2. lm 회귀함수로 기울기와 절편을 구한다.
attach(launch)
lm( distress_ct ~ temperature,  launch )
       # ↑              ↑
      # y축           x축
     # intercept = 절편, tannin = 기울기    # y = -0.04754x + 3.69841 이라는 방정식을 구할 수 있다.

> lm( distress_ct ~ temperature,  launch )
Call:
lm(formula = distress_ct ~ temperature, data = launch)
Coefficients:
(Intercept)  temperature
    3.69841     -0.04754
```

3. plot 그래프를 생성한다.

plot(distress_ct ~ temperature, data=launch, col="red", bg="red", pch=21)

 # x축 : 온도 # y축 : o형링 파손수

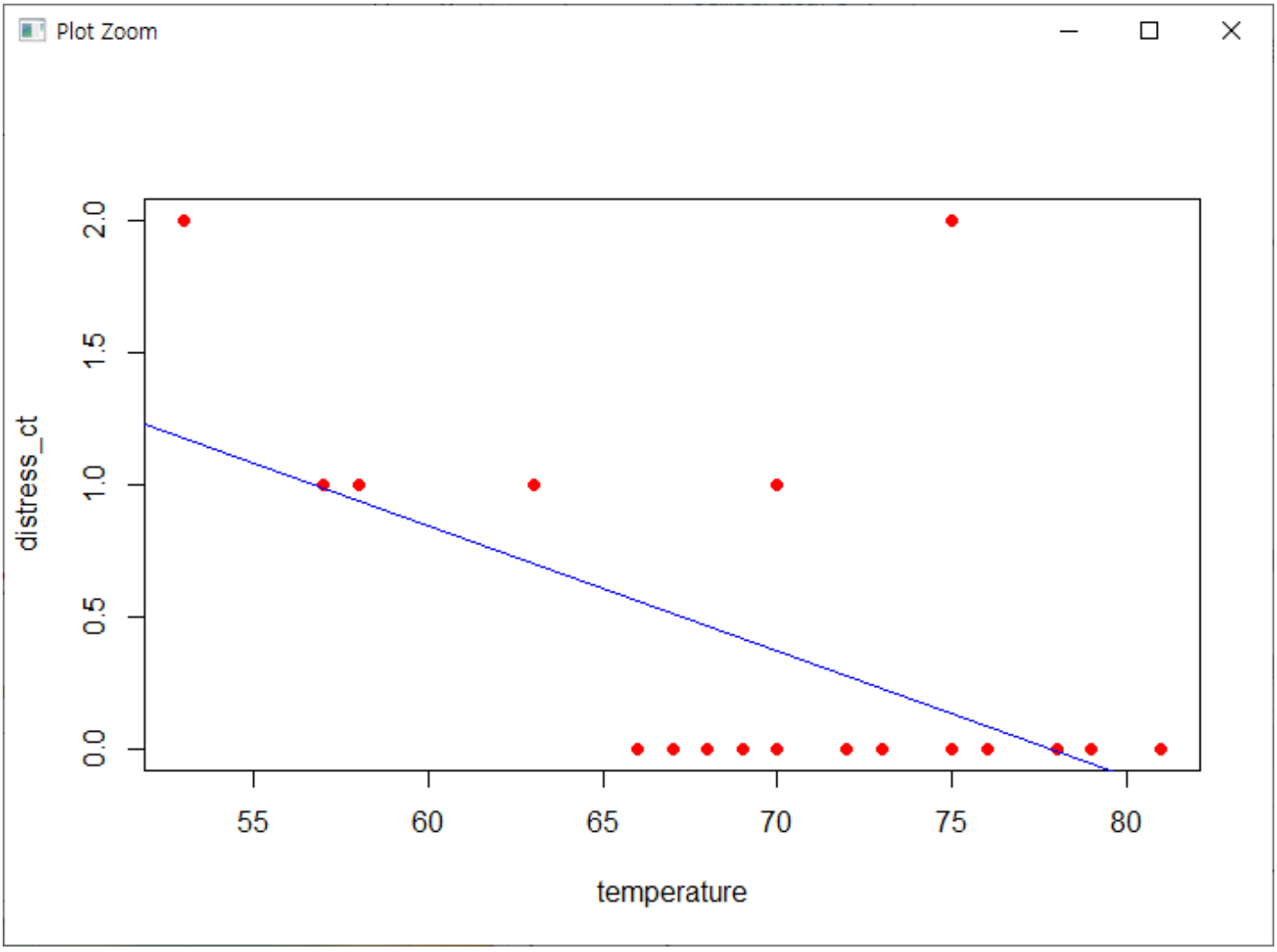
공식 : plot(y축~x축, data=데이터셋 이름)



4. plot 그래프의 data 에 맞는 회귀직선을 그린다.

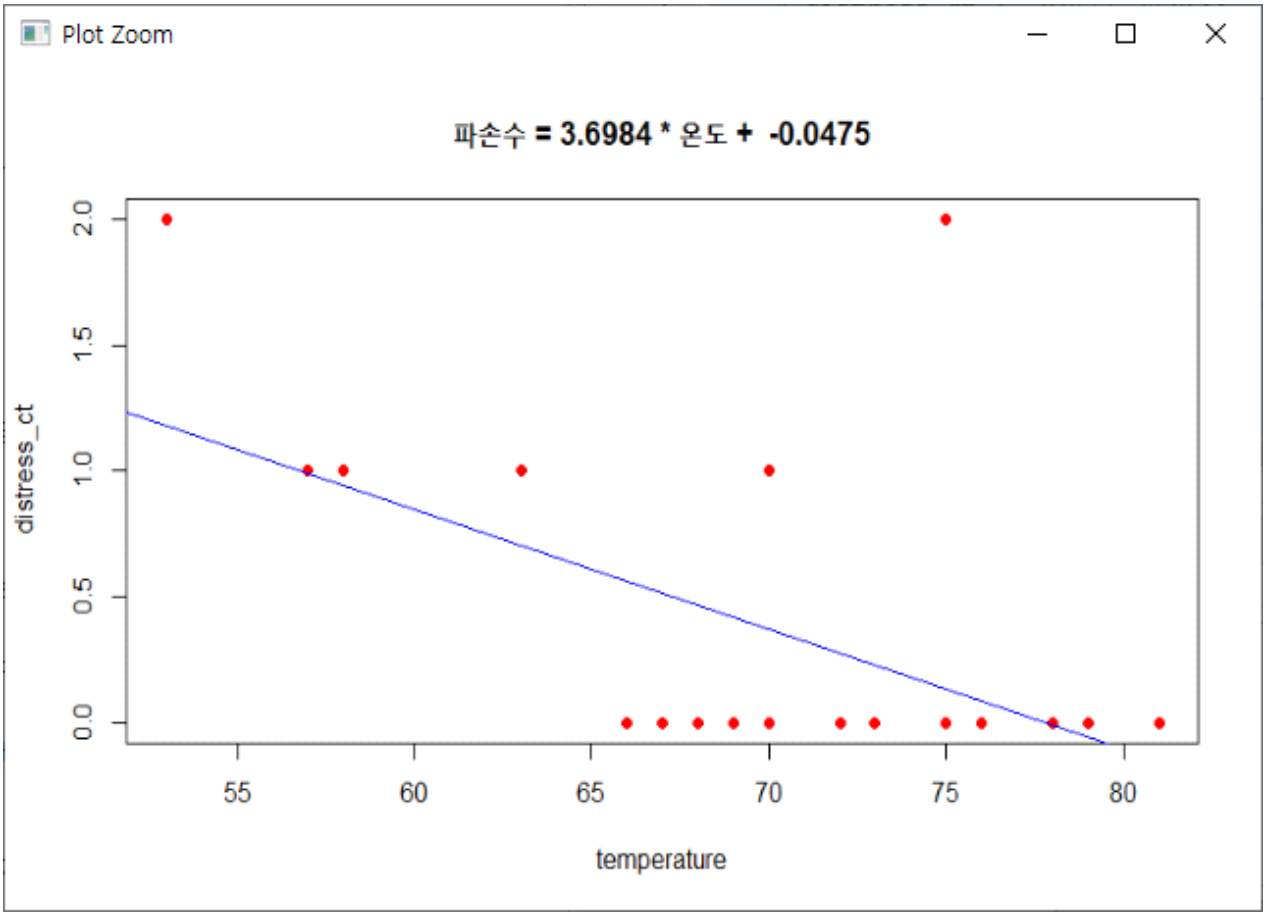
m <- lm(distress_ct ~ temperature, launch)

abline(m , col="blue")



5. 회귀직선의 방정식을 제목으로 출력한다. # y = -0.04754 x + 3.69841

title(paste('파손수 =', round(m$coefficients[1], 4), "* 온도 + ", round(m$coefficients[2], 4)))


Plot Zoom
파손수 = 3.6984 * 온도 + -0.0475
distress_ct
0.0
0.5
1.0
1.5
2.0
55
60
65
70
75
80
temperature

다중공선성

2020년 6월 26일 금요일 오후 3:47

단순회귀분석	------>	상관관계	------>	다중회귀분석
		↓		
		다중공선성		

■ 다중공선성 (variance inflation factor)

회귀분석에서 사용된 모형의 일부 설명변수(독립변수)가 다른 독립변수와의 상관정도가 높아 데이터 분석 시 부정적인 영향을 미치는 현상을 말한다.

두 독립변수들끼리 서로에게 영향을 주고 있다면
둘 중 하나의 영향력을 검증할 때 다른 하나의 영향력을 완전히 통제할 수 없게 된다.

예시 : 학업성취도, 일평균음주량, 혈중 알코올 농도가 있을 때
↑ ↑
종속변수 나머지가 독립변수

음주가 학업 성취도에 미치는 영향을 알아보려고 회귀분석을 하려고 한다.
일평균 음주량과 혈중 알코올 농도는 **서로 아주 강한 상관정도를 보인다**.

실제로 x1과 x2의 값이 증가 또는 감소할수록 y 값이 증가 또는 감소할 것인데,
이 중 하나는 굉장히 불안정한 계수값을 보이게 된다.

그래서 다중회귀분석을 할 때 꼭 다중공선성을 확인하고 가야한다.

공선성은 두개의 독립변수들 간의 관계를 의미하는데,

예를 들어 두 개의 독립변수들 간의 상관관계 계수가 1이면
완전한 공선성을 보인다고 하고, 계수가 0이면 전혀 공선성이 없음을 의미한다.

특히 3개 이상의 변수들 간의 관계를 다중 공선성이라 한다.

한 독립변수가 종속변수에 대한 설명력이 높더라도 (다중) 공선성이 높으면 설명력이 낮은 것처럼 나타난다.

우리 반에 선생님이 2명 있다고 생각
-> 한 선생님의 영향력이 크면 다른 선생님의 영향력이 약해지는 원리

다중공선성을 알아보기 위한 가장 간단한 방법은 독립변수들 간의 상관관계를 조사하는 것이다. **독립변수들 간의 높은 상관관계**는 **공선성을 판단하는 지표**이다.
(일반적으로 0.9 이상)

공선성을 보다 엄격하게 점검하려면 공차한계와 분산 팽창요인을 본다.

팽창계수(VIF)

현업기준 : 팽창계수(VIF)가 보통 10보다 큰 것으로 골라내고, 엄격하게 하려면 5보다 큰 것을 골라낸다.
느슨하게 하려면 15 또는 20으로 주로 골라낸다.

예제 :

```
# install.packages("car")
library(car)
data(Boston, package='MASS')
head(Boston)     # 보스턴 지역의 집값 데이터
```

```
> data(Boston, package='MASS')
> head(Boston)
     crim zn indus chas   nox     rm  age     dis rad tax ptratio  black lstat medv
1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98 24.0
2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14 21.6
3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03 34.7
4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94 33.4
5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33 36.2
6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12  5.21 28.7
```

boston.csv

데이터명 : Boston Housing Price (보스턴 주택 가격 데이터)
레코드수 : 506 개
필드개수 : 14 개

데이터설명 : 보스턴 시의 주택 가격에 대한 데이터이다.
주택의 여러가지 요건들과 주택의 가격 정보가 포함되어 있다. 주택의 가격에 영향을 미치는 요소를 분석하고자 하는 목적으로 사용될 수 있다. 회귀분석 등의 분석에 활용될 수 있다.

보스턴 주택 데이터는 여러 개의 측정지표들 (예를 들어, 범죄율, 학생/교사 비율 등)을 포함한, 보스턴 인근의 주택 가격의 중앙값(median value)이다. 이 데이터 집합은 14개의 변수를 포함하고 있다.

필드의 이해 :
데이터의 이해를 돕기 위해 포함된 14개의 변수에 대하여 간략하게 설명한다.
위 14개의 필드는 입력 변수로 사용되고, 맨 아래의 Play 속성이 목표(종속) 변수로 사용된다.

[01] CRIM	자치시(town) 별 1인당 범죄율
[02] ZN	25,000 평방피트를 초과하는 거주지역의 비율
[03] INDUS	비소매상업지역이 점유하고 있는 토지의 비율
[04] CHAS	찰스강에 대한 더미변수(강의 경계에 위치한 경우는 1, 아니면 0)
[05] NOX	10ppm 당 농축 일산화질소
[06] RM	주택 1가구당 평균 방의 개수
[07] AGE	1940년 이전에 건축된 소유주택의 비율
[08] DIS	5개의 보스턴 직업센터까지의 접근성 지수
[09] RAD	방사형 도로까지의 접근성 지수
[10] TAX	10,000 달러 당 재산세율
[11] PTRATIO	자치시(town)별 학생/교사 비율
[12] B	1000(Bk-0.63)^2, 여기서 Bk는 자치시별 흑인의 비율을 말함.
[13] LSTAT	모집단의 하위계층의 비율(%)
[14] MEDV	본인 소유의 주택가격(중앙값) (단위: $1,000)

-> 마지막 MEDV 가 종속변수

앞의 1~13번까지의 독립변수들과 14번 MEDV와의 관계를 예측하는 것이 다중회귀분석
이때 먼저 독립변수들 간에 강한 상관관계를 가지는 것이 있는지 확인해야 한다.

데이터파일

출처: http://ai-times.tistory.com/431 [ai-times]

From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SeFi&contentval=0004lzzzzzzzzzzzzzzzzzzzzzzzz&datanum=295&page=1&prev_page=0&firstbbsdepth=&lastbbsdepth=zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz&listnum=20>

상관관계

2020년 6월 26일 금요일 오후 4:14

단순회귀분석	------>	상관관계	------>	다중회귀분석
		↓		
		다중공선성		

■ 상관관계

두 변수 간의 상관관계는 변수들의 관계가 직선에 가깝게 따르는 정도를 나타내는 숫자이다. 상관관계는 -1에서 +1 사이의 범위에 있다. 극값은 완벽한 선형관계를 나타내는 반면, 0에 가까운 상관관계는 선형관계가 없음을 나타낸다.

Corr(x, y) =

r에서는 cor이 상관관계를 구하는 함수이다.

문제 251. 온도와 o형링 파손수 간의 상관계수를 구하시오.

```
launch <- read.csv("challenger.csv", header=T)
cor(launch$temperature, launch$distress_ct)
```

```
> cor(launch$temperature, launch$distress_ct)
[1] -0.5111264
```

-> 적당히 음의 상관관계가 있음을 의미한다.

문제 252. 삼성전자와 현대자동차 둘 중에 코스피 등락율과 더 상관관계가 높은 주식이 어떤 것인지 알아내시오.

상관관계 구해보기

2020년 6월 26일 금요일 오후 4:22

■ 코스피 지수 수익율과 삼성전자와 현대자동차 수익율의 상관관계

K_index.csv

S_stock.csv

H_stock.csv

```
k_index <- read.csv("K_index.csv", header=T,  stringsAsFactors=F)

s_stock <- read.csv("S_stock.csv", header=T, stringsAsFactors=F)

h_stock <- read.csv("H_stock.csv", header=T,  stringsAsFactors=F)

all_data <- merge(merge(k_index,s_stock), h_stock)

head(all_data)

attach(all_data)
```

```
          y축(삼성 수익율 등락 비율)
            ↓
plot(k_rate, s_rate, col="blue")
     ↑
  x축(코스피 등락 비율)
```

```
plot(k_rate, h_rate, col="blue")
```

문제. 코스피 등락 비율과 삼성 수익율 등락 비율을 plot 그래프로 그리고 그 그래프에 회귀직선을 그으시오 !

```
plot(k_rate, s_rate, col="blue")

model_s <- lm( s_rate ~ k_rate, data=all_data)

abline( model_s,  col="red")
```

문제223. 현대자동차도 마찬가지로 회귀 그래프를 만드시오 !

```
plot(k_rate, h_rate, col="blue")

model_h <- lm( h_rate ~ k_rate, data=all_data)

abline( model_h,  col="red")
```

문제. 현대 자동차와 삼성전자 그래프를 하나의 화면으로 출력되게 하시오 !

```
graphics.off()
par(mfrow=c(1,2), new=T)
par(mar=c(2,2,2,2) )

plot(k_rate, s_rate, col="blue")
model_s <- lm( s_rate ~ k_rate, data=all_data)
abline( model_s,  col="red")

plot(k_rate, h_rate, col="blue")
model_h <- lm( h_rate ~ k_rate, data=all_data)
abline( model_h,  col="red")
```

From <http://cafe.daum.net/_c21_/bbs_read?grpid=zchT&fldid=SZTZ&datanum=2088>

마지막 문제

2020년 6월 26일 금요일 오후 4:47

문제 254. (오늘의 마지막 문제) 자동화 스크립트에 1번 산포도 그래프에 회귀 직선이 같이 출력되게 하시오.

S_stock.csv를 로드하면 회귀직선도 같이 출력되도록 만들기