미국 총기 사고 예측

빅데이터 기술

201303009

이진웅

목차

교수님 피드백

발표 당시의 문제점

- 데이터 분석, 예측 시 제외하는 열 없이 분석, 예측 해보고 정확도 비교해보기
- 데이터 분할 시 비율이 너무 적다.

수정한 점

- 문자열 열도 분석에 참여시키기 위해 데이터 형 변환

열 이름	변환 전	변환 후
Intent(사망유형)	Suicide(자살)	0
	Accidental(우연사)	1
	Homicide(살인)	2
Police(경찰개입여부)	경찰 개입x	0
	경찰 개입o	1
Sex(성별)	남자	0
	여자	1
Race(인종)	White(백인)	0
	Asian(아시아인)	1
	Black(흑인)	2
	Native American(미국인)	3
	Hispanic(히스패닉)	4
Place(사망 장소)	Home(집)	0
	Street(거리)	1
	Other specified(기타 장소)	2
	Trade/service area(서비스지역)	3
	School/instiution(학교)	4
	Other unspecified(기타불특정)	5
Education(최종학력)	고졸미만	0
	고졸	1
	대재	2
	대졸	3
	확인불가	4

- 훈련데이터의 비율이 너무 적다.

기존 훈련데이터(10%), 검증데이터(90%) 에서
변경 훈련데이터(70%), 검증데이터(30%) 으로 수정

```
test_idx <- createDataPartition(gun$intent, p=0.7)$Resample1
# 훈련데이터(70%), 검증데이터(30%)
```

데이터 소개

- 2012 ~ 2014년 미국의 총기 사망사고에 대한 정보

컬럼명	내용
Year	사망자가 발생한 해
Month	사망자가 사망한 달
Intent	사망 유형(예측대상)
Police	사건 경찰 연루 여부
Sex	희생자의 성별
Age	희생자의 나이
Race	희생자의 인종
Histonic	희생자의 히스패닉 기원 코드
Place	사건 장소
education	희생자의 최종 학력

데이터 수정

```
gun <- read.csv("C:/Users/kslbs/Desktop/guns수치화.csv", header = T, stringsAsFactors = FALSE)
#read.csv로 csv파일 불러오기
```

```
gun <- gun[, !names(gun) %in% c("X","hispanic")]
```

```
#불필요 컬럼 제거
```

```
gun <- na.omit(gun)
#gun 데이터에 결측치 제거
```

```
gun$sex <- as.numeric(gun$sex)
gun$police <- as.numeric(gun$police)
gun$race <- as.numeric(gun$race)
gun$place <- as.numeric(gun$place)
gun$intent <- as.numeric(gun$intent)
gun$year <- as.numeric(gun$year)
gun$month <- as.numeric(gun$month)
gun$age <- as.numeric(gun$age)
gun$education <- as.numeric(gun$education)
#데이터 타입 지정
```

데이터 탐색

```
test_idx <- createDataPartition(gun$intent, p=0.7)$Resample1
#Y 값을 고려한 데이터의 분할(훈련데이터70%, 검증데이터30%)
```

```
gun.test <- gun[test_idx,]
gun.train <- gun[-test_idx,]
nrow(gun.test)
nrow(gun.train)
#test데이터와 train데이터로 분리
```

```
prop.table(table(gun.train$intent))
#gun데이터의 사망유형 비율
> prop.table(table(gun.train$intent))

          0           1           2
0.63466499 0.34909716 0.01623785
```

```
createFolds(gun.train$intent, k=10)
#데이터 분리
create_ten_fold_cv <- function() {
  set.seed(137)
  lapply(createFolds(gun.train$intent , k=10), function(idx) {
    return(list(train=gun.train[-idx, ],
```

```
                    validation=gun.train[idx, ]))
  })
}
#10겹 교차 검증 데이터를 만드는 함수
```

```
summary(intent ~ year + month + police + sex + race + education, data = data, method = "reverse")
# 각 변수 값에 따른 사망유형 종류
```

	0 (N=19191)	1 (N=10556)	2 (N=491)
year : 12	32% (6212)	35% (3662)	35% (173)
13	34% (6550)	33% (3530)	30% (149)
14	34% (6429)	32% (3364)	34% (169)
month	4/ 6/ 9	4/ 7/10	3/ 7/10
police	0% (0)	4% (419)	0% (0)
sex	14% (2711)	15% (1574)	14% (71)
race : 0	1% (231)	2% (177)	1% (4)
1	88% (16812)	26% (2727)	67% (330)
2	1% (145)	1% (111)	2% (8)
3	5% (1036)	56% (5862)	23% (112)
4	5% (967)	16% (1679)	8% (37)
education : 0	0% (2)	0% (9)	1% (3)
1	15% (2950)	33% (3530)	32% (155)
2	42% (8018)	44% (4697)	38% (185)
3	24% (4613)	16% (1699)	20% (98)
4	17% (3353)	5% (486)	10% (47)
5	1% (255)	1% (135)	1% (3)

```
xtabs( ~ intent + race , data=data)
# 사망사유, 인종별 분할표(xtabs)
> xtabs( ~ intent + race , data=data)
      race
intent     0     1     2     3     4
     0   231 16812   145  1036   967
     1   177  2727   111  5862  1679
     2     4   330     8   112    37
xtabs( ~ year + intent , data=data)
# 연도, 사망사유별 분할표(xtabs)
```

```
> xtabs( ~ year + intent , data=data)
    intent
year    0    1    2
  12 6212 3662  173
  13 6550 3530  149
  14 6429 3364  169
```

xtabs(~ sex + year , data=data)

연도, 성별 별 사망자 수 분할표(xtabs)

```
    year
sex   12   13   14
  0 8663 8729 8490
  1 1384 1500 1472
```

xtabs(~ year+ race , data=data)

연도, 인종별 사망자 수 분할표(xtabs)

```
     race
year    0    1    2    3    4
  12  130 6479   81 2463  894
  13  163 6802   82 2322  860
  14  119 6588  101 2225  929
```

xtabs(intent == "0" ~ year+ race , data=data)

연도, 인종별 사망자 수 분할표(사망 사유가 '자살')

```
     race
year    0    1    2    3    4
  12   66 5438   41  366  301
  13   96 5764   50  322  318
  14   69 5610   54  348  348
```

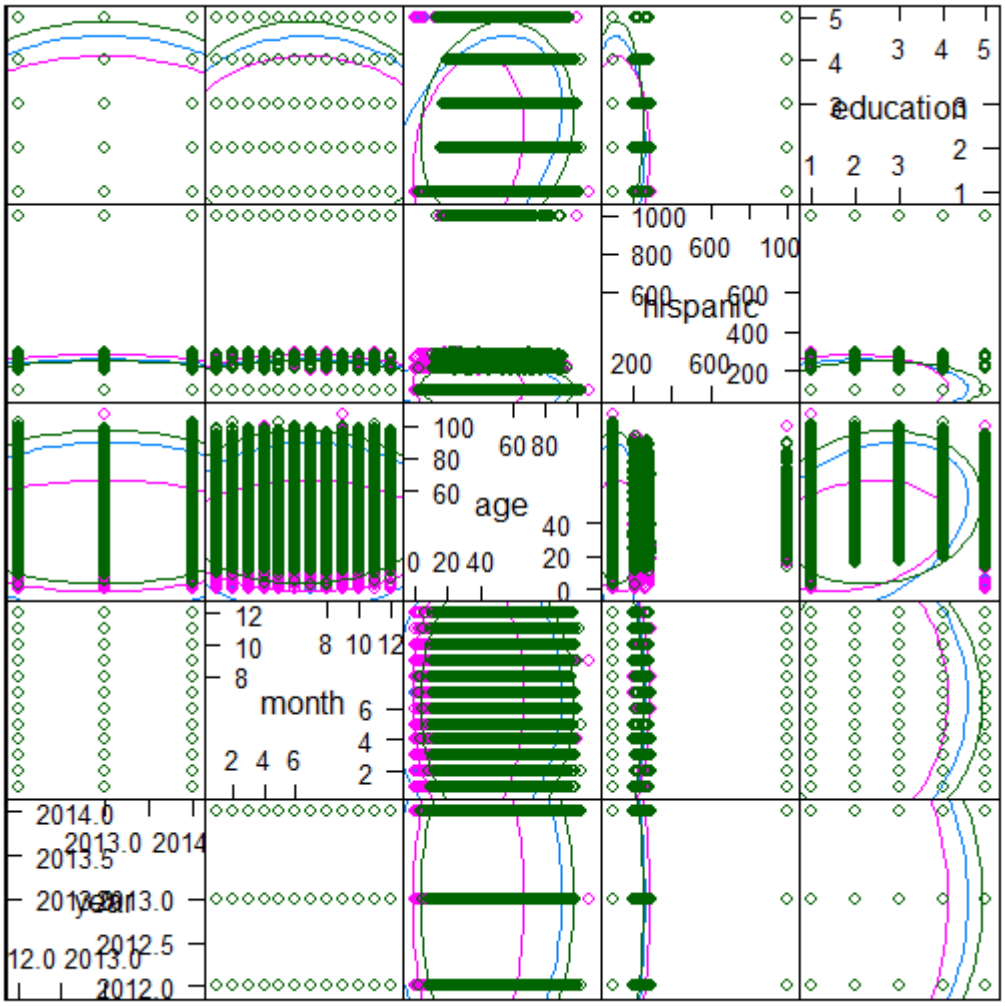
xtabs(race == "White" ~ sex + year , data=data) / xtabs(race == "Black" ~ sex + year , data=data)

연도, 성별 별 사망자 수 분할표(백인 사망 수/흑인 사망 수 >>> 남자는 4~5배, 여자는 2~3배 차이)

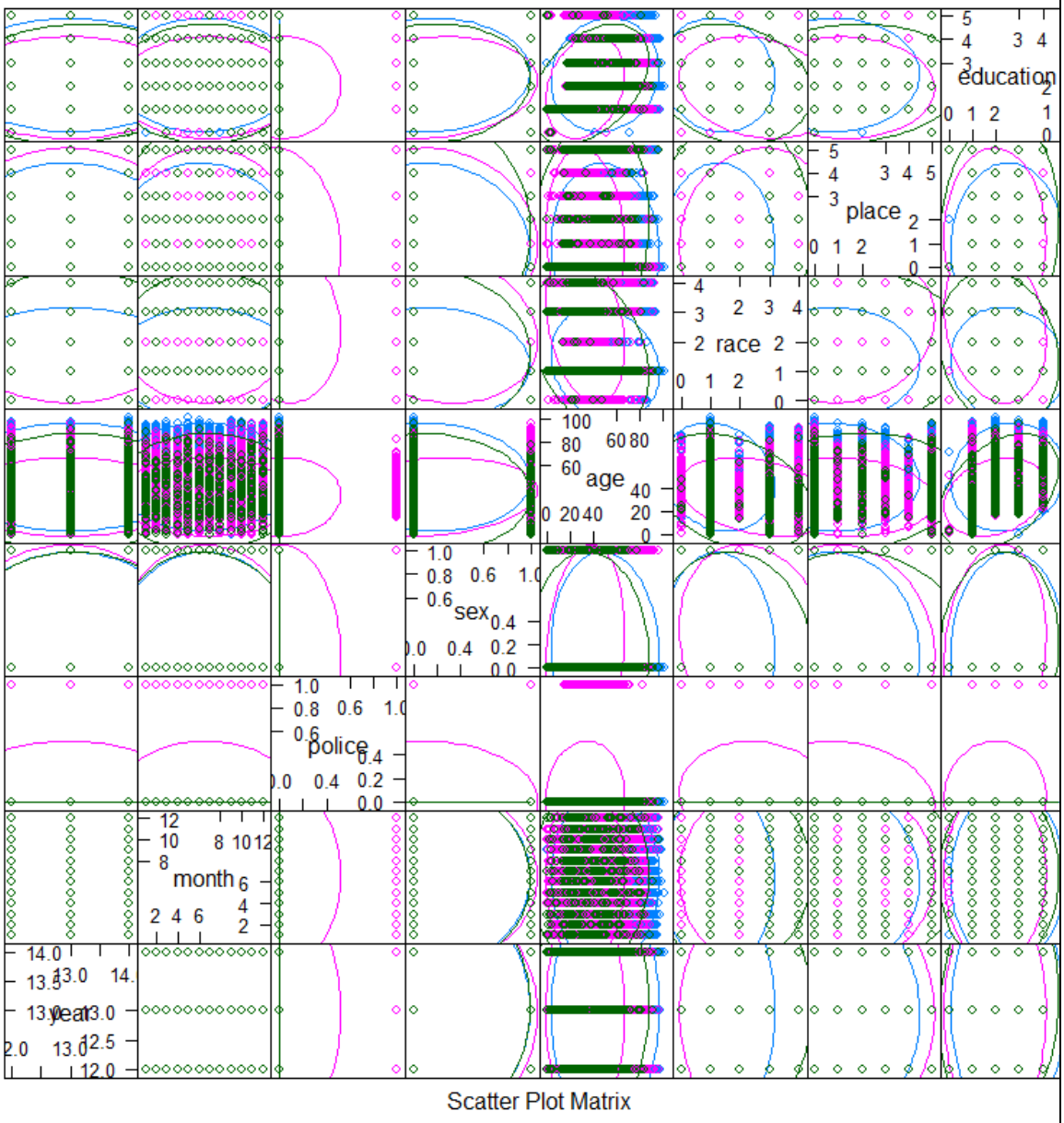
```
    year
sex        12        13        14
  0 1.5492958 1.8000000 1.2405063
  1 2.0000000 3.0833333 0.9545455
```

데이터 시각화

- featurePlot(
  data.complete[,sapply(names(data.complete),function(n) { is.numeric(data.complete [, n]) })],
  data.complete [, c("intent")], "ellipse")
  #featureplot을 이용한 데이터 시각화
  1. 데이터 수정 전

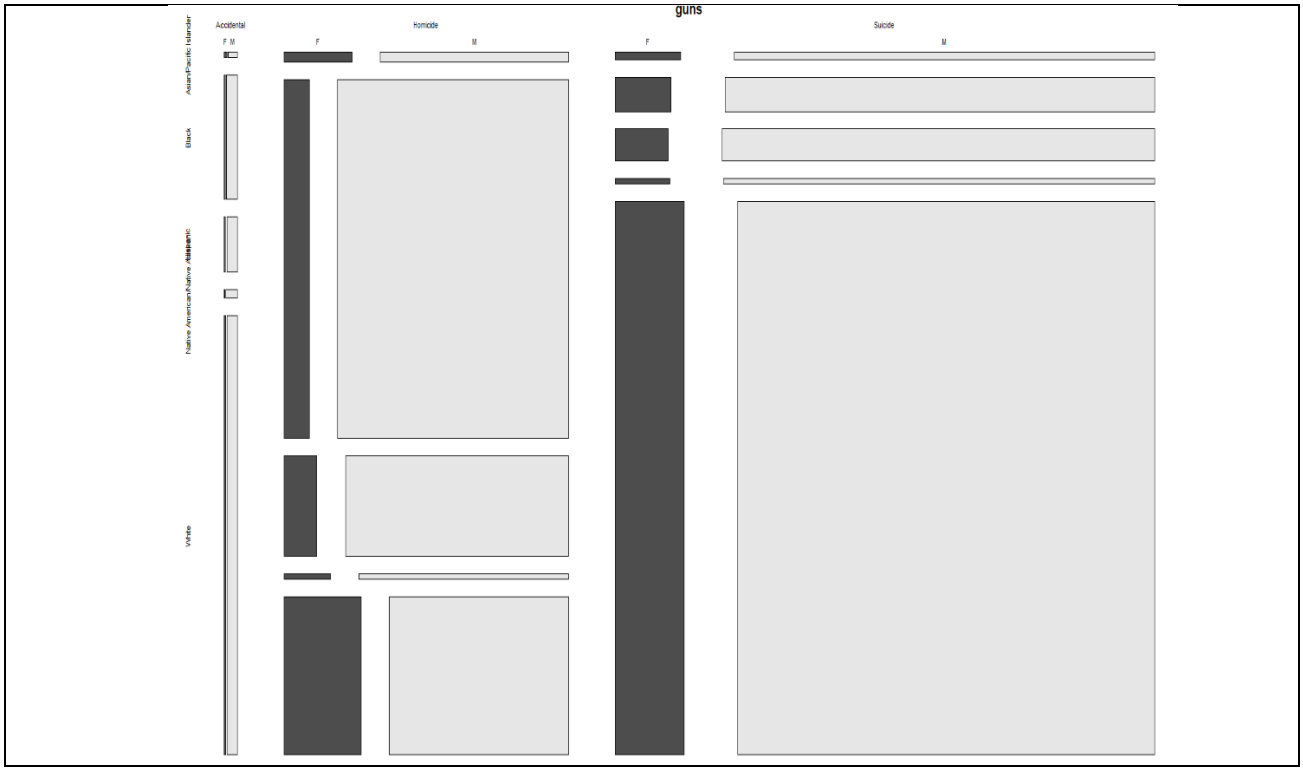


2. 데이터 수정 후(참여 열 추가 + 형 변환)



Scatter Plot Matrix

- mosaicplot(intent ~ race + sex, data = data, color=TRUE, main="guns")
  #인종과 성별로 구분한 사망유형(mosaicplot), 사망유형과 성별로 비교한 모자이크플롯



.

데이터 예측 및 정확도

Rpart 예측모델
m <- rpart(intent ~ year + month + police + sex + age + race + place + education, data=gun.train) p <- predict(m, newdata = gun.train, type = "class") head(p) # rpart 모델만듬
folds <- create_ten_fold_cv() rpart_result <- foreach(f=folds) %do% { model_rpart <- rpart(intent ~ year + month + police + sex + age + race + place + education, data=f$train) predicted <- predict(model_rpart, newdata=f$validation, type="class") return(list(actual=f$validation$intent, predicted=predicted))} # folds 전체에 대한 결과를 리스트로 묶어서 변수에 저장
evaluation <- function(lst) {

```
  accuracy <- sapply(lst, function(one_result) {
    return(sum(one_result$predicted == one_result$actual)
           / NROW(one_result$actual))
  })
  print(sprintf("MEAN +/- SD: %.3f +/- %.3f", mean(accuracy), sd(accuracy)))
  return(accuracy)
}
#평균과 표준편차를 계산한 뒤 Accuracy의 벡터를 결과로 반환
```

```
evaluation(rpart_result)
rpart_accuracy <- evaluation(rpart_result)
#rpart 모델의 성능 : 82.2%, 오차범위 : 0.010
> evaluation(rpart_result)
[1] "MEAN +/- SD: 0.822 +/- 0.010"
```

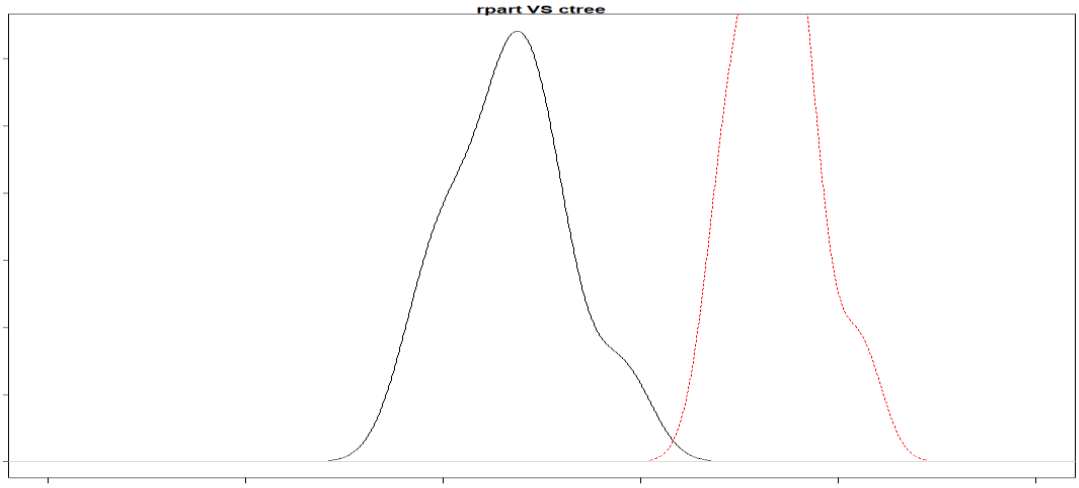
ctree 예측모델

```
ctree_result <- foreach(f=folds) %do% {
  model_ctree <- ctree(intent ~ year + month + police + sex + age + race + place
+ education,
                       data=f$train)
  predicted <- predict(model_ctree , newdata=f$validation, type="response")
  return(list(actual=f$validation$intent , predicted=predicted))
}
#ctree :type에 response를 지정해야 class가 반환
```

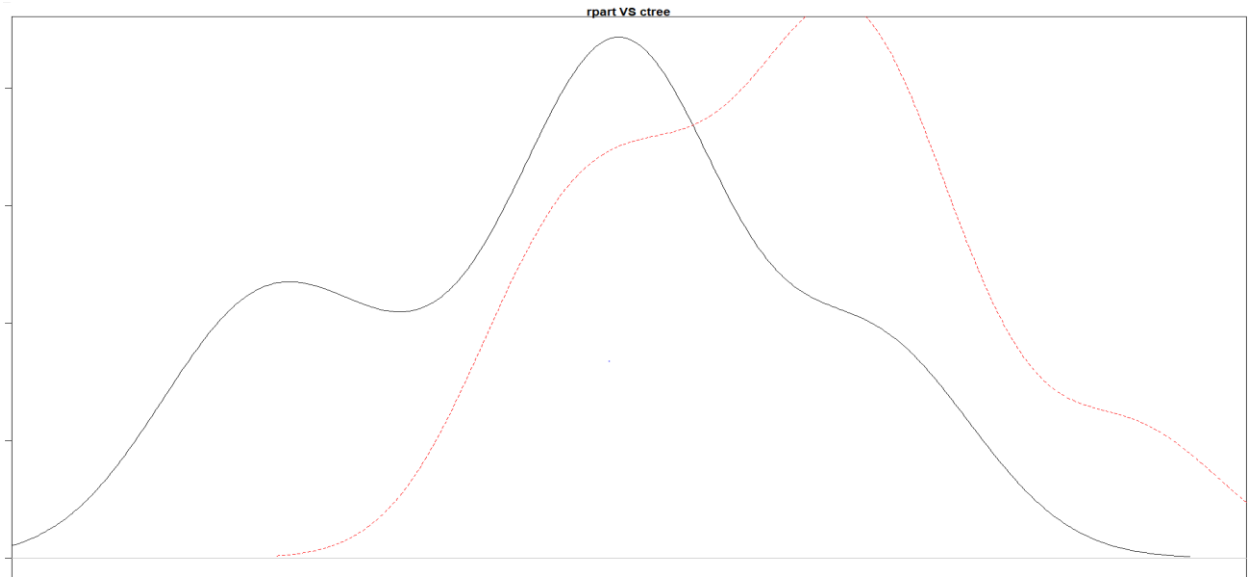
```
ctree_accuracy <- evaluation(ctree_result)
#cpart 모델의 성능: 83.2%, 오차범위 : 0.008(rpart와 1%나 차이가난다.)
 ctree_accuracy <- evaluation(ctree_result)
L] "MEAN +/- SD: 0.832 +/- 0.008"
```

Rpart 와 ctree 정확도 비교

1. 데이터 수정 전



2. **데이터 수정 후** (참여 열 추가 + 형 변환)



피어슨 상관계수

```
corrg <-cor (gun[,c("intent","police","sex","age","race","education")])
#상관계수 비교할 열만 추출하여 변수에 저장
```

```
corrgram(corrg,type = "corr",upper.panel = panel.conf)
#corr상관계수 그래프 그리기
```

1. 데이터 수정 전

2. 데이터 수정 후(참여 열 추가 + 형 변환)

피어슨 상관계수를 통한 분할표

인종과 사망유형 별 분할표

```
> xtabs( ~ intent + race , data=data)
      race
intent     0     1     2     3     4
     0   678 50320   519  3133  2906
     1   475  7597   272 17221  4826
     2    10  1013    18   289   132
```

백인과 흑인 자살율 1위 백인과 흑인 살해율 1위 백인과 흑인 돌발사 1위	Race(인종) 0 : 아시아인 1 : 백인 2 : 토종 미국인 3: 흑인 4: 라인계	intent(사망유형) 0 : 자살 1 : 타살 2 : 돌발사

```
> xtabs( ~ year + intent , data=data)
      intent
year      0     1     2
   2012 18881 10429   490
   2013 19296 10073   445
   2014 19379  9889   527
```

해를 거듭 할 수록 미국의 자살율이 늘고있다

```
> sum(predicted == actual) / NROW(predicted)
[1] 0.6104497
```

예측한 값 중 정확히 예측한 값의 비율(정확도 = 0.6104497)

결론

피드백을 받기 전까지 10만개가 넘는 데이터 중에 훈련데이터를 10%밖에 두지 않았다. 피드백을 받고 훈련데이터와 검증데이터의 비율을 7:3으로 바꾼 후 다시 분석 및 예측을 해보니 확실히 결과가 다르게 나왔다. Rpart와 ctree에서의 정확도도 살짝 다르게 나왔다. 또한 예측에 참여하는 열도 추가로 늘려서 분석을 해보니 결과가 바뀌는 것을 알 수 있었다.

느낀 점

한 학기 동안 배운 것을 기반으로 프로젝트를 진행하였는데 처음엔 예측 및 분석에 사용하기 적절한 데이터를 찾는 것에 시간이 오래 걸렸다. 검색 결과 한국 공공 데이터 말고도 해외 공공데이터를 구할 수 있는 사이트를 찾게 되었고, 그곳에서 미국의 총기 사고에 관한 데이터를 찾았다. 이 데이터로 프로젝트를 진행하기에 적합하다 생각되어 프로젝트를 진행하게 되었다. 데이터 탐색 부분까지는 순조롭게 진행을 했지만 데이터 예측(rpart & ctree) 쪽에서 많이 어려움을 겪었다. 교재와 인터넷으로 공부를 하며 잘못 된 부분을 찾을 수 있었고 예측 부분까지도 끝낼 수 있었다. 발표 후 데이터 열을 늘려보라는 것과 훈련 데이터 비율을 늘려보라는 교수님의 피드백을 받았고, 피드백을 기준으로 코드를 수정 해 나갔다. 코드 수정 후 결과를 보니 확실히 차이가 있었다. 10만줄을 빅데이터라고 부르기에는 터무니없지만 피드백을 통해 느낀 것은 데이터의 양이 많아 졌을 때 이런 사소한 설정으로도 차이가 크게 날 수 있다는 점이었다.