자동차 사고 Accidents.csv 파일에 새 레벨의 부상(NO INJURY, INJURY, FATALITY) 중의 하나와 연관된 2001년 미국의 실제 자동차 사고 42,183건에 대한 정보가 수록되어 있다. 각 사고에 대해서 요일, 기상조건, 도로 종류와 같은 추가적인 정보가 기록되어 있다. 어떤 회사가 초기 보고서와 이 시스템의 연관된 데이터(그중 일부는 GPS-지원 보고에 의존함)에 근거하여 사고의 심각성을 신속하게 분류하는 시스템을 개발하고자 한다.
여기서 우리의 목표는 보고된 사고가 부상이 동반될지(MAX_SEV_IR=1이나 2), 없을지(MAX_SEV_IR=0)를 예측하는 것이다. 이러한 목적을 위해서 MAX_SEV_IR=1이나 2면 "YES"를 취하고 그렇지 않으면 "NO"를 취하는 INJURY라는 가변수를 생성하시오.

a. 이 데이터세트의 정보를 사용하여 사고가 막 보고되는데 추가적인 정보가 없다면, 예측은 무엇이 되어야 하는가? (INJURY=YES 또는 NO?) 그 이유는?

```
setwd("c:/rdata")
accidents.df <- read.csv("c:/rdata/accidents.csv")

accidents.df$INJURY <- ifelse(accidents.df$MAX_SEV_IR>0, "yes", "no")

for(i in c(1:dim(accidents.df)[2])){
  accidents.df[,i] <-as.factor(accidents.df[,i])
}

prop.table(table(accidents.df$INJURY))
> prop.table(table(accidents.df$INJURY))

       no       yes
0.4912168 0.5087832
```

=> INJURY='YES'로 예측한다.
(∵추가적인 정보가 없다면 조건부 확률이 아닌 각 사건의 단순 확률을 보고 예측한다.)

b. 이 데이터세트의 처음 12개 레코드를 선택하여 응답(INJURY)과 두 개의 예측변수 WEARTHER_R과 TRAF_CON_R만을 고려하시오.

I. 이 12개의 레코드들에 대해서 두 예측변수들의 함수로서 INJURY를 검토하는 피벗 테이블을 작성하시오. 피벗 테이블의 세 변수들 모두 행/열로 사용하시오.

```
> table(accidents.df[1:12, c("INJURY", "WEATHER_R", "TRAF_CON_R")])
, , TRAF_CON_R = 0

      WEATHER_R
INJURY 1 2
   no  1 5
   yes 2 1

, , TRAF_CON_R = 1

      WEATHER_R
INJURY 1 2
   no  1 1
   yes 0 0

, , TRAF_CON_R = 2

      WEATHER_R
INJURY 1 2
   no  1 0
   yes 0 0
```

=>신호기가 없을 때 INJURY가 'yes' 값을 갖는다.
(신호기가 있느냐 없느냐가 INJURY에 영향을 미친다.)

II. 예측변수들의 여섯 개 가능한 조합이 주어졌을 때 부상이 있을(INJURY=yes) 정확한 베이즈 조건부 확률을 계산하시오.

```
> head(accidents.df[, c("INJURY", "WEATHER_R", "TRAF_CON_R")], 12)
   INJURY WEATHER_R TRAF_CON_R
1     yes         1          0
2      no         2          0
3      no         2          1
4      no         1          1
5      no         1          0
6     yes         2          0
7      no         2          0
8     yes         1          0
9      no         2          0
10     no         2          0
11     no         2          0
12     no         1          2
```

$$P(INJURY = yes | WEATHER\_R = 1, TRAF\_CON\_R = 0) = \frac{2}{3}$$

$$P(INJURY = yes | WEATHER\_R = 1, TRAF\_CON\_R = 1) = \frac{0}{1} = 0$$

$$P(INJURY = yes | WEATHER\_R = 1, TRAF\_CON\_R = 2) = \frac{0}{1} = 0$$

$$P(INJURY = yes | WEATHER\_R = 2, TRAF\_CON\_R = 0) = \frac{1}{6}$$

$$P(INJURY = yes | WEATHER\_R = 2, TRAF\_CON\_R = 1) = \frac{0}{1} = 0$$

$$P(INJURY = yes | WEATHER\_R = 2, TRAF\_CON\_R = 2) = 0$$

III) 이 확률값들과 컷오프 값 0.5를 사용하여 3개의 사고를 분류하시오.

첫 번째 사고의 확률 $\frac{2}{3}$ => 부상이 동반된 사고로 분류

두 번째 사고의 확률 $\frac{0}{1} = 0$ => 부상이 동반되지 않는 사고로 분류

세 번째 사고의 확률 $\frac{0}{1} = 0$ => 부상이 동반되지 않는 사고로 분류

IV) WEATHER_R=1과 TRAF_CON_R=1이 주어졌을 때 부상이 있을 나이브 베이즈 조건부 확률을 수작업으로 계산하시오.

```
> head(accidents.df[, c("INJURY", "WEATHER_R", "TRAF_CON_R")], 12)
   INJURY WEATHER_R TRAF_CON_R
1     yes         1          0
2      no         2          0
3      no         2          1
4      no         1          1
5      no         1          0
6     yes         2          0
7      no         2          0
8     yes         1          0
9      no         2          0
10     no         2          0
11     no         2          0
12     no         1          2
```

$$P(WEATHER\_R=1,\ TRAF\_CON\_R=1|INJURY=yes)P(INJURY=yes)$$
$$=P(WEATHER\_R=1|INJURY=yes)\times P(TRAF\_CON\_R=1|INJURY=yes)\times P(INJURY=yes)$$
$$=\frac{2}{3}\times 0\times \frac{1}{4}=0$$

$$P(WEATHER\_R=1,\ TRAF\_CON\_R=1|INJURY=no)P(INJURY=no)$$
$$=P(WEATHER\_R=1|INJURY=no)\times P(TRAF\_CON\_R=1|INJURY=no)\times P(INJURY=no)$$
$$=\frac{1}{3}\times \frac{2}{9}\times \frac{3}{4}=\frac{1}{18}$$

$$P_{NB}(INJURY=yes|WEATHER\_R=1,\ TRAF\_CON\_R=1)$$
$$=\frac{P(WEATHER\_R=1,\ TRAF\_CON\_R=1|INJURY=yes)P(INJURY=yes)}{P(WEATHER\_R=1,\ TRAF\_CON\_R=1|INJURY=yes)P(INJURY=yes)+P(WEATHER\_R=1,\ TRAF\_CON\_R=1|INJURY=no)P(INJURY=no)}$$
$$=\frac{0}{0+\frac{1}{18}}=0$$

V) 12개의 레코드에 대해서 나이브베이즈 분류기를 돌려서 결과에 대한 정오분류표를 작성하시오.

```
library(e1071)
accidents.nb <- naiveBayes(INJURY ~ WEATHER_R+TRAF_CON_R , data=accidents.df[1:12,c('INJURY', 'WEATHER_R', 'TRAF_CON_R')])
accidents.nb

pred.class <- predict(accidents.nb, newdata=accidents.df[1:12,c('INJURY', 'WEATHER_R', 'TRAF_CON_R')])
pred.class

library(caret)
confusionMatrix(pred.class, accidents.df[1:12, 'INJURY'], positive = 'yes')
```

```
> confusionMatrix(pred.class, accidents.df[1:12, 'INJURY'], positive = 'yes')
Confusion Matrix and Statistics

          Reference
Prediction no yes
       no   9   3
       yes  0   0

               Accuracy : 0.75
                 95% CI : (0.4281, 0.9451)
    No Information Rate : 0.75
    P-Value [Acc > NIR] : 0.6488

                  Kappa : 0

 Mcnemar's Test P-Value : 0.2482

            Sensitivity : 0.00
            Specificity : 1.00
         Pos Pred Value :  NaN
         Neg Pred Value : 0.75
             Prevalence : 0.25
         Detection Rate : 0.00
   Detection Prevalence : 0.00
      Balanced Accuracy : 0.50

       'Positive' Class : yes
```

c. 이제는 전체 데이터세트로 돌아가 보자. 데이터를 학습(60%)과 검증(40%)세트로 분할하시오.

```
selected.var  <- c(25,1,2,8,15,16,17,19)
train.index <- sample(c(1:dim(accidents.df)[1]), dim(accidents.df)[1]*0.6)
train.df <-accidents.df[ train.index, selected.var]
valid.df <-accidents.df[ - train.index, selected.var]
```

ii) 관련된 예측변수들(과 응답으로서 INJURY)을 갖는 완전한 학습세트를 이용하여 나이브 베이즈 분류기를 실행하시오. 모든 예측변수들은 범주형임에 주목하고, 정오행렬을 보이시오.

```
accidents.nb <- naiveBayes(INJURY ~ ., data=train.df)
pred.class <- predict(accidents.nb, newdata = valid.df)
pred.class

confusionMatrix(pred.class, valid.df$INJURY, positive = 'yes')
```

```
> confusionMatrix(pred.class, valid.df$INJURY, positive = 'yes')
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  2387 1905
       yes 5930 6652

               Accuracy : 0.5357
                 95% CI : (0.5281, 0.5432)
    No Information Rate : 0.5071
    P-Value [Acc > NIR] : 5.941e-14

                  Kappa : 0.0648

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.7774
            Specificity : 0.2870
         Pos Pred Value : 0.5287
         Neg Pred Value : 0.5562
             Prevalence : 0.5071
         Detection Rate : 0.3942
   Detection Prevalence : 0.7456
      Balanced Accuracy : 0.5322

       'Positive' Class : yes
```

iii) 검증 데이터세트에 대한 전체 오차는 얼마인가?

1-0.5357 = 0.4643