다변량 팀프로젝트
판별 분석

2조
김남철
권태양
곽수진

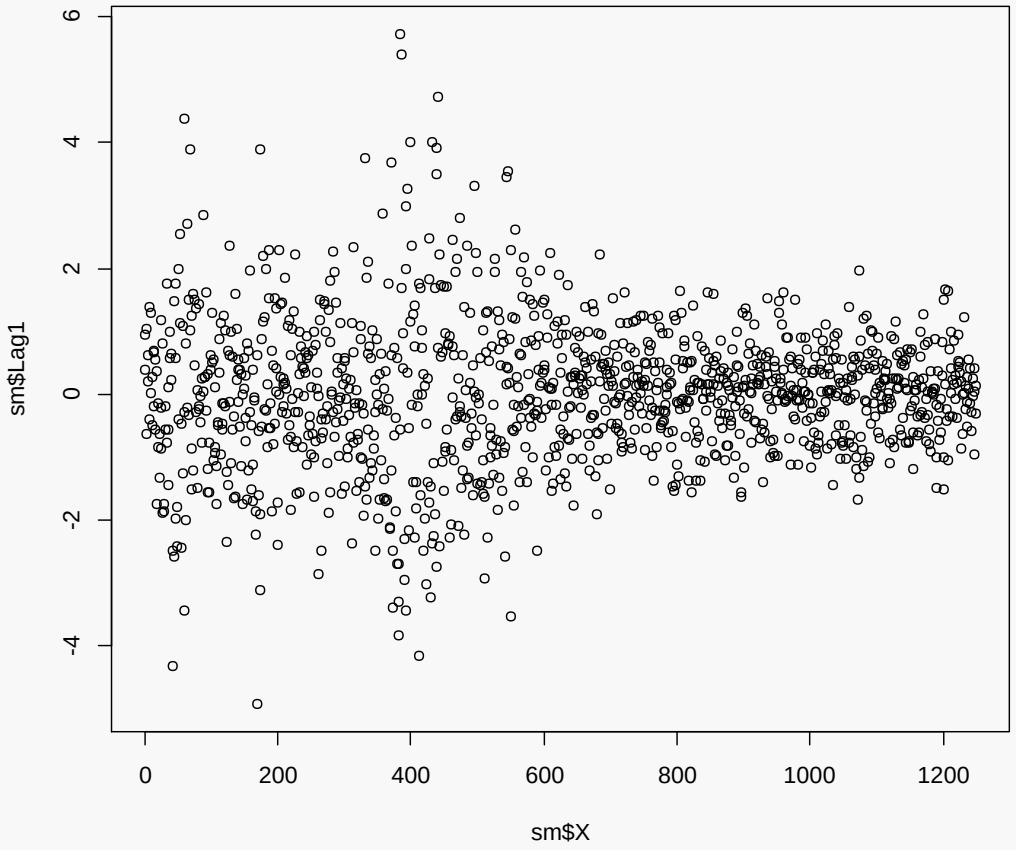
```
> sm <-
read.csv("C:/Users/sunni/OneDrive/바탕 화면/
태양/`19년 1학기/다변량통계분석/smarket_
판별분석.csv",header = TRUE)
```

```
> head(sm)
  X  Lag1   Lag2 Direction
1 1 0.381 -0.192        Up
2 2 0.959  0.381        Up
3 3 1.032  0.959      Down
4 4 -0.623 1.032        Up
5 5 0.614 -0.623        Up
6 6 0.213  0.614        Up
```

⇒ 데이터 확인

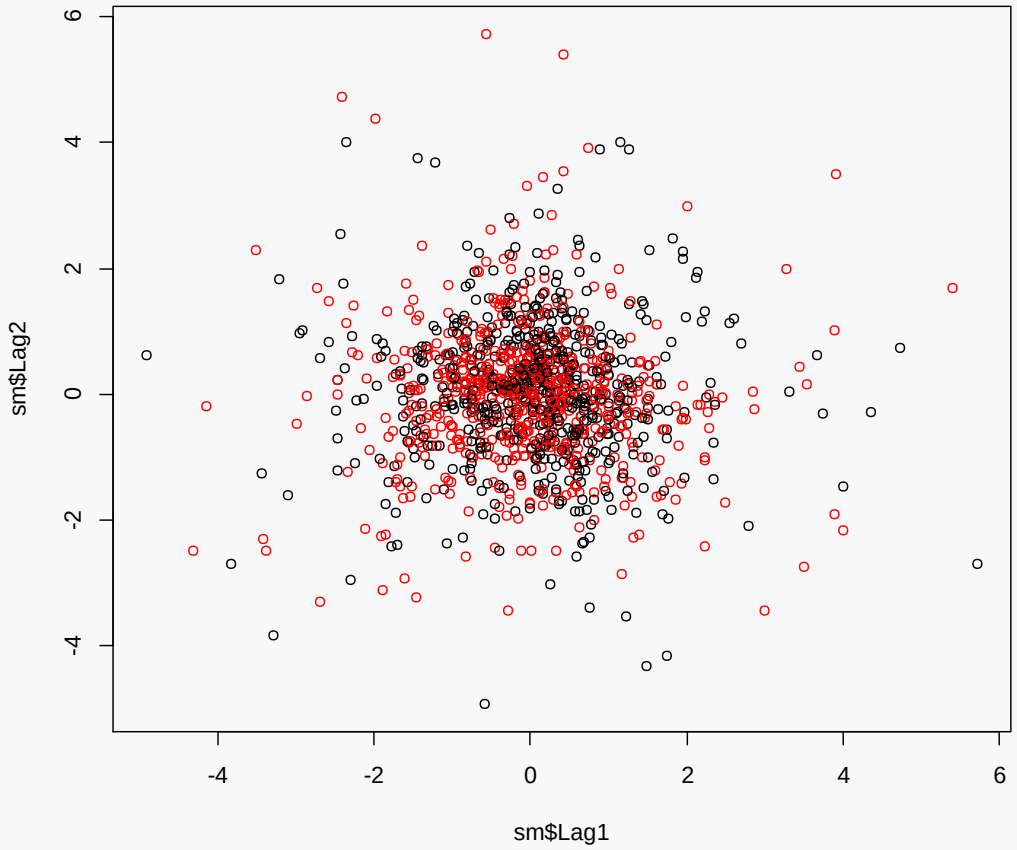
X : 날짜 (단위 : 하루)
Lag1 : 1일 전 주식 수익률
Lag2 : 2일 전 주식 수익률
Direction : 현 시점의 수익률 상승, 감소

> plot(smX,smLag1)



=> 시간별 데이터 분포도

> plot(sm$Lag1,sm$Lag2,
col=as.numeric(sm$Direction))



=> 하루 전날, 이틀 전날에 따른 Up, Down값 비교

<선형 판별분석>

```
> rs <- lda(sm$Direction ~ sm$Lag1 + sm$Lag2, data = sm)

> rs
Call:
lda(sm$Direction ~ sm$Lag1 + sm$Lag2, data = sm)

Prior probabilities of groups:
  Down     Up
0.4816 0.5184

Group means:
            sm$Lag1       sm$Lag2
Down  0.05068605  0.03229734
Up     -0.03969136 -0.02244444

Coefficients of linear discriminants:
             LD1
sm$Lag1 -0.7567605
sm$Lag2 -0.4707872
```

=> 선형 판별분석을 수행해 선형 판별식을 얻었다.

판별식 : Lag1 * -0.7567605 + Lag2 * -0.4707872

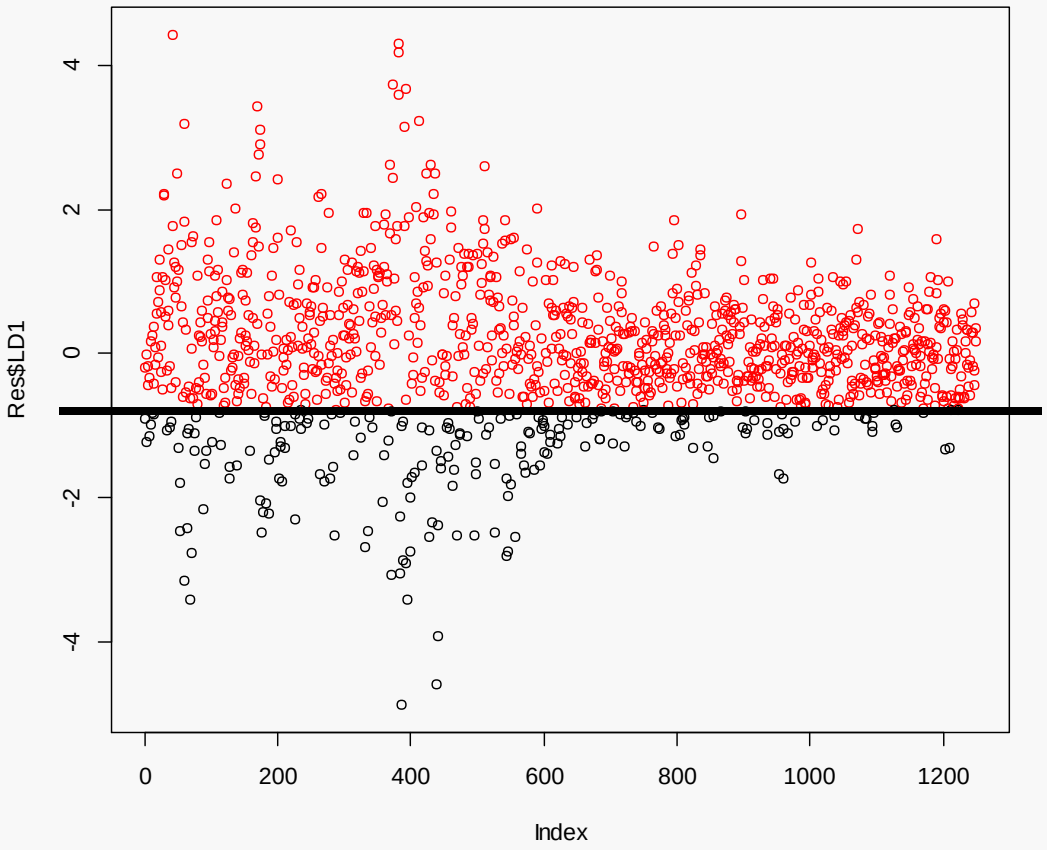
```
> calc <- with(x, Lag1 * (-0.7567605) + Lag2 *
(-0.4707872))

> head(calc)
[1] -0.1979346 -0.9051032 -1.2324618 -
0.0143906 -0.1713505 -0.4502533

> p <- predict(rs)

> X <- cbind(sm, p$x)

> head(X)
   X   Lag1   Lag2   Direction        LD1
1 1  0.381 -0.192       Up     -0.193187790
2 2  0.959  0.381       Up     -0.900356413
3 3  1.032  0.959     Down     -1.227714911
4 4 -0.623  1.032       Up     -0.009643717
5 5  0.614 -0.623       Up     -0.166603724
6 6  0.213  0.614       Up     -0.445506476
```

=> Lag1, Lag2의 값을 판별함수에 대입해 LD1이라는 값이 나왔다.

```
> pc = predict(rs, sm)$class

> head(pc)
[1] Up   Down Down Up   Up   Up
Levels: Down Up

> pc=as.numeric(pc)

> res = cbind(sm$Direction, pc)

> Res <- cbind(X, res)

> head(Res)
   X   Lag1   Lag2   Direction        LD1        V1 pc
1 1  0.381 -0.192       Up     -0.193187790  2   2
2 2  0.959  0.381       Up     -0.900356413  2  1
3 3  1.032  0.959     Down   -1.227714911  1   1
4 4 -0.623  1.032       Up     -0.009643717  2   2
5 5  0.614 -0.623       Up     -0.166603724  2   2
6 6  0.213  0.614       Up     -0.445506476  2   2
```

=> Direction을 수치형으로 변환하고,
예측값과 함께 Res 열에 추가했다.

> plot(Res$LD1, col=Res$pc)



> min(Res[Res$pc == 2,]$LD1)
[1] -0.7792976

> max(Res[Res$pc == 1,]$LD1)
[1] -0.7863482

=> 판별함수로 나온 값이 대략 -0.78을 기준으로 나눠지는 것을 알 수 있음.

-0.78보다 클 경우 Up
-0.78보다 작을 경우 Down

판별함수에 데이터를 넣어 나온 값의 예측값 확인.

```
> correct = res[(sm$Direction == pc),]

> correct.rate = dim(correct)[[1]]/n

> correct.rate
[1]   0.528 165.000

> table(res[,1],res[,2])

     1   2
  1 114 488
  2 102 546
```

=> 오분류율 : 1 – 0.528 = 0.472

=> 전체 데이터를 가지고 모델링 했을 때, 나오는 오분류율

<이 차 판별분석>

```
> x = sm[,2:3]
> qd = qda(x,sm$Direction)
> qc = predict(qd)$class
> head(qc)
[1] Up Up Down Up Up Up
Levels: Down Up
> qc = as.numeric(qc)
> head(qc)
[1] 2 2 1 2 2 2
> resq=cbind(sm$Direction,qc)
> correctq = resq[(resq[,1]==resq[,2]),]
> correctq.rate=dim(correctq)[[1]]/n
> correctq.rate
[1] 0.5304 165.7500
```

=> 오분류율 : 1 – 0.5304 = 0.4696

– 선형 판별분석 보다는 조금 낮게 나왔기 때문에 오분류율만 봤을때는 이 차 판별분석이 더 나은 분석방법이라고 볼 수 있다.

<선형 판별분석 교차타당성>

```
> ldc = lda(sm$Direction ~ sm$Lag1 + sm$Lag2,
data = sm, CV = TRUE, prior = c(0.4816,0.5184))

> results = data.frame(sm$Direction, ldc$class,
ldc$posterior)

> head(results)
  Direction ldc.class      Down        Up
1        Up        Up 0.4861599 0.5138401
2        Up      Down 0.5030997 0.4969003
3      Down      Down 0.5098580 0.4901420
4        Up        Up 0.4822116 0.5177884
5        Up        Up 0.4857059 0.5142941
6        Up        Up 0.4921773 0.5078227
```

```
> class.table = table(sm$Direction,
ldc$class)

> class.table

Direction Down  Up
     Down  109 493
       Up  109 539
```

정분류율 : (109+539) /1250 = 0.5184
오분류율 : 1 – 0.5184 = 0.4816

감사합니다