PCA(Principal components analysis)

Lab10. 주성분 분석, 로지스틱 회귀 분석

학습 목표

```
01. 주성분 분석(PCA)에 대해 알아본다.
02. 로지스틱 회귀 분석에 대해 알아본다.
```

학습 내용

```
01. 주성분 분석(PCA)이란?
02. 로지스틱 회귀 분석

참고교재 : R in Action page363
```

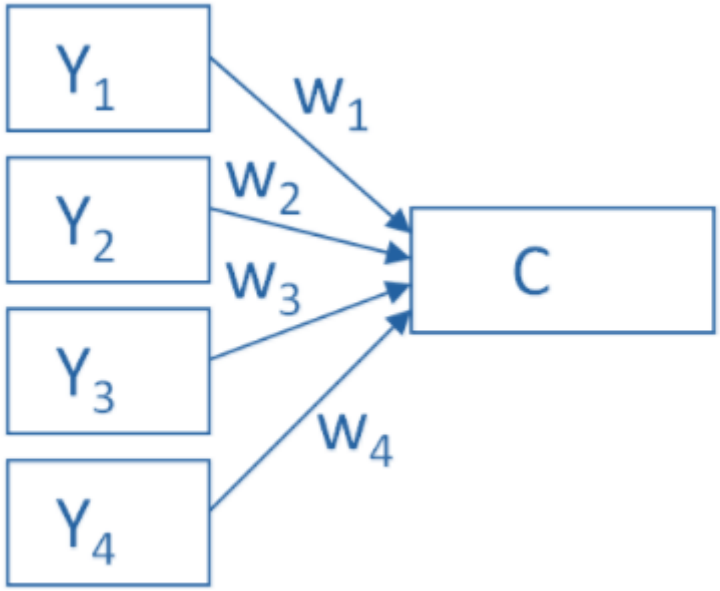
01. 주성분 분석(PCA)이란?

```
PCA은 상관성 있는 많은 변수들을 주성분(principal component)이라
부르는 상관성이 없는
보다 적은 수의 변수 집합으로 자료를 축소하는 방법.

[간단한 방정식의 표현]
C=w1(Y1) + w2(Y2) + w3(Y3) + w4(Y4)

[R에서의 주성분 분석 함수]
prcomp()

[사용]
prcomp(formula, data=NULL, subset, na.action...)
```



1-1 데이터 불러오기

- prcomp (df, scale=T)
- scale=T는 수치간 표준화를 지정하는 것이다.

```
data(iris)        # iris 데이터를 사용하기 위한 코드
df <- iris[,1:4]  # Species 를 제외한 데이터 선택
iris.pca <- prcomp(df, scale=T)  # 수치간 표준화를 지정
iris.pca
```

```
## Standard deviations:
## [1] 1.7083611 0.9560494 0.3830886 0.1439265
##
## Rotation:
##                     PC1         PC2        PC3        PC4
## Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
## Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
## Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
## Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

02. 주성분 분석(PCA)이란(2)?

```
[간단한 방정식의 표현]
C=w1(Y1) + w2(Y2) + w3(Y3) + w4(Y4)

가) 주성분(principal components)로 불리는 생성된 변수들은 관찰변수들의 선형결합.
나) 첫 주성분은 원 변수들 집합에서 가장 많은 분산을 설명하는 K개의 관찰변수 가중치의 결합이다.
PC1 = a1X1 + a2X2 + ... + akXk
다) 두번째 주성분은 첫 번째 주성분과 직각 관계(상관관계 없음)라는 조건 하에서
    원 변수의 분산을 가장 많이 설명하는 선형 결합이다.
라) 이론적으로 변수의 개수만큼 주성분 추출이 가능.
마) 실제적으로는 전체 변수의 수보다 훨씬 적은 성분으로 전체 분산을 설명하기를 원함.
```

03. pca 정보 보기

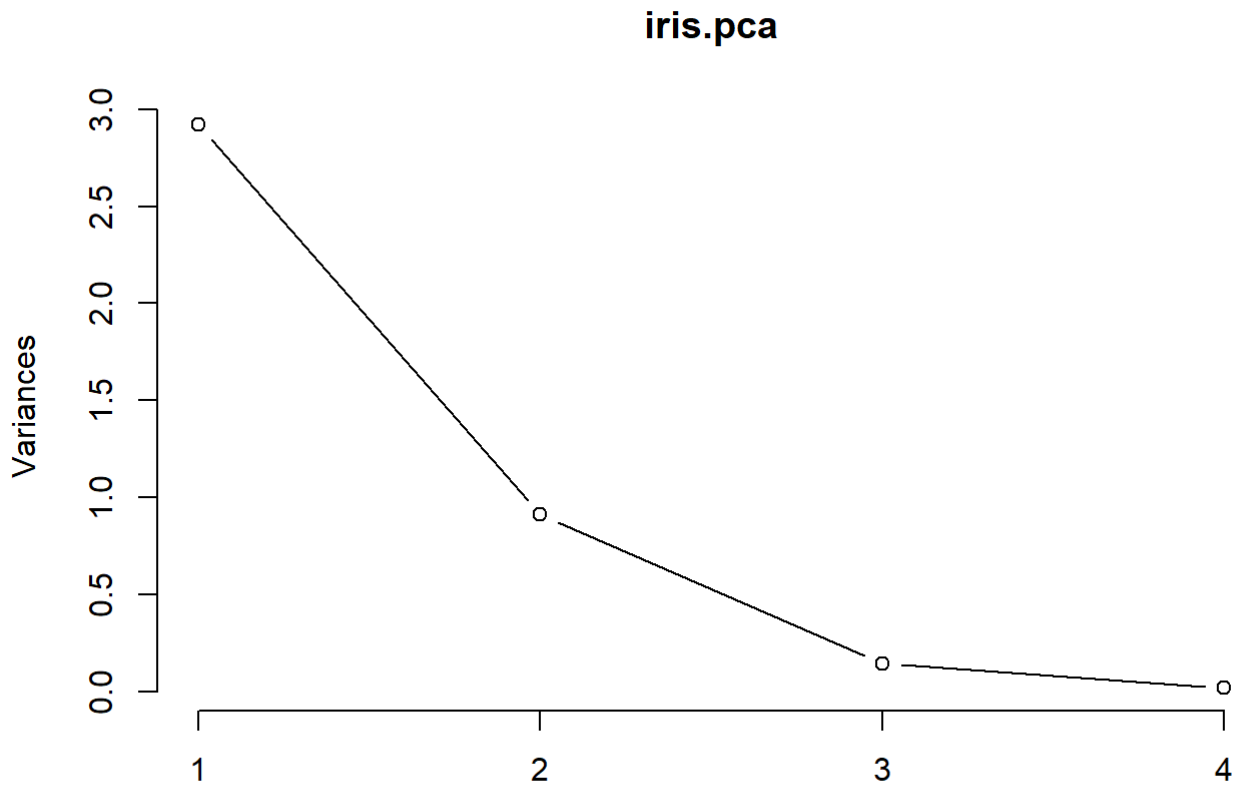
```
summary(iris.pca)  # PCA 결과 요약
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.7084 0.9560 0.38309 0.14393
## Proportion of Variance 0.7296 0.2285 0.03669 0.00518
## Cumulative Proportion  0.7296 0.9581 0.99482 1.00000
```

- 표준편차(standard deviation)은 각 변수가 얼마나 많은 부분을 차지하고 있는지
- Proportion of Variance(분산비율) : 각 주성분의 차치하는 비율. 클수록 영향력이 높다.

- Cumulative Proportion(분산의 누적합계)
- 보통 분산의 누적합계가 80%이상인 것 까지 하지만 통상적인 기준이다.

```
plot(iris.pca, type='l')
```



predict를 명령어를 이용하여 새로운 주성분으로 계산된 값을 구할 수 있다.

```
iris.predict <- predict(iris.pca)  # 주성분 점수 계산
iris.predict[, 1:3]  # 주성분 1~ 3의 점수 출력
```

```
##               PC1          PC2          PC3
##   [1,] -2.25714118 -0.478423832  0.127279624
##   [2,] -2.07401302  0.671882687  0.233825517
##   [3,] -2.35633511  0.340766425 -0.044053900
##   [4,] -2.29170679  0.595399863 -0.090985297
##   [5,] -2.38186270 -0.644675659 -0.015685647
##   [6,] -2.06870061 -1.484205297 -0.026878250
##   [7,] -2.43586845 -0.047485118 -0.334350297
##   [8,] -2.22539189 -0.222403002  0.088399352
##   [9,] -2.32684533  1.111603700 -0.144592465
##  [10,] -2.17703491  0.467447569  0.252918268
##  [11,] -2.15907699 -1.040205867  0.267784001
##  [12,] -2.31836413 -0.132633999 -0.093446191
##  [13,] -2.21104370  0.726243183  0.230140246
##  [14,] -2.62430902  0.958296347 -0.180192423
##  [15,] -2.19139921 -1.853846555  0.471322025
##  [16,] -2.25466121 -2.677315230 -0.030424684
##  [17,] -2.20021676 -1.478655729  0.005326251
##  [18,] -2.18303613 -0.487206131  0.044067686
##  [19,] -1.89223284 -1.400327567  0.373093377
##  [20,] -2.33554476 -1.124083597 -0.132187626
##  [21,] -1.90793125 -0.407490576  0.419885937
##  [22,] -2.19964383 -0.921035871 -0.159331502
##  [23,] -2.76508142 -0.456813301 -0.331069982
##  [24,] -1.81259716 -0.085272854 -0.034373442
##  [25,] -2.21972701 -0.136796175 -0.117599566
##  [26,] -1.94532930  0.623529705  0.304620475
##  [27,] -2.04430277 -0.241354991 -0.086075649
##  [28,] -2.16133650 -0.525389422  0.206125707
##  [29,] -2.13241965 -0.312172005  0.270244895
##  [30,] -2.25769799  0.336604248 -0.068207276
##  [31,] -2.13297647  0.502856075  0.074757996
##  [32,] -1.82547925 -0.422280389  0.269564311
##  [33,] -2.60621687 -1.787587272 -0.047070727
##  [34,] -2.43800983 -2.143546796  0.082392024
##  [35,] -2.10292986  0.458665270  0.169706329
##  [36,] -2.20043723  0.205419224  0.224688852
##  [37,] -2.03831765 -0.659349230  0.482919584
##  [38,] -2.51889339 -0.590315163 -0.019370918
```

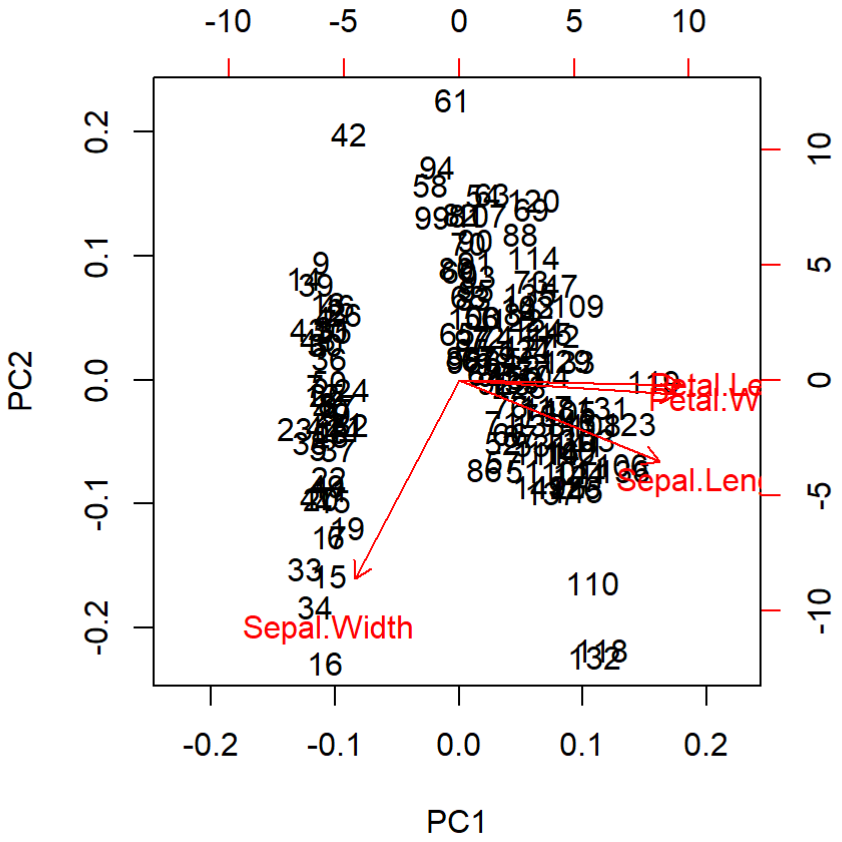
```
##  [39,] -2.42152026  0.901161067 -0.192609402
##  [40,] -2.16246625 -0.267981199  0.175296561
##  [41,] -2.27884081 -0.440240541 -0.034778398
##  [42,] -1.85191836  2.329610745  0.203552303
##  [43,] -2.54511203  0.477501017 -0.304745527
##  [44,] -1.95788857 -0.470749613 -0.308567588
##  [45,] -2.12992356 -1.138415464 -0.247604064
##  [46,] -2.06283361  0.708678586  0.063716370
##  [47,] -2.37677076 -1.116688691 -0.057026813
##  [48,] -2.38638171  0.384957230 -0.139002234
##  [49,] -2.22200263 -0.994627669  0.180886792
##  [50,] -2.19647504 -0.009185585  0.152518539
##  [51,]  1.09810244 -0.860091033  0.682300393
##  [52,]  0.72889556 -0.592629362  0.093807452
##  [53,]  1.23683580 -0.614239894  0.552157058
##  [54,]  0.40612251  1.748546197  0.023024633
##  [55,]  1.07188379  0.207725147  0.396925784
##  [56,]  0.38738955  0.591302717 -0.123776885
##  [57,]  0.74403715 -0.770438272 -0.148472007
##  [58,] -0.48569562  1.846243998 -0.248432992
##  [59,]  0.92480346 -0.032118478  0.594178807
##  [60,]  0.01138804  1.030565784 -0.537100055
##  [61,] -0.10982834  2.645211115  0.046634215
##  [62,]  0.43922201  0.063083852 -0.204389093
##  [63,]  0.56023148  1.758832129  0.763214554
##  [64,]  0.71715934  0.185602819  0.068429700
##  [65,] -0.03324333  0.437537419 -0.194282030
##  [66,]  0.87248429 -0.507364239  0.501830204
##  [67,]  0.34908221  0.195656268 -0.489234095
##  [68,]  0.15827980  0.789451008  0.301028700
##  [69,]  1.22100316  1.616827281  0.480693656
##  [70,]  0.16436725  1.298259939  0.172260719
##  [71,]  0.73521959 -0.395247446 -0.614467782
##  [72,]  0.47469691  0.415926887  0.264067576
##  [73,]  1.23005729  0.930209441  0.367182178
##  [74,]  0.63074514  0.414997441  0.290921638
##  [75,]  0.70031506  0.063200094  0.444537765
##  [76,]  0.87135454 -0.249956017  0.471001057
##  [77,]  1.25231375  0.076998069  0.724727099
```

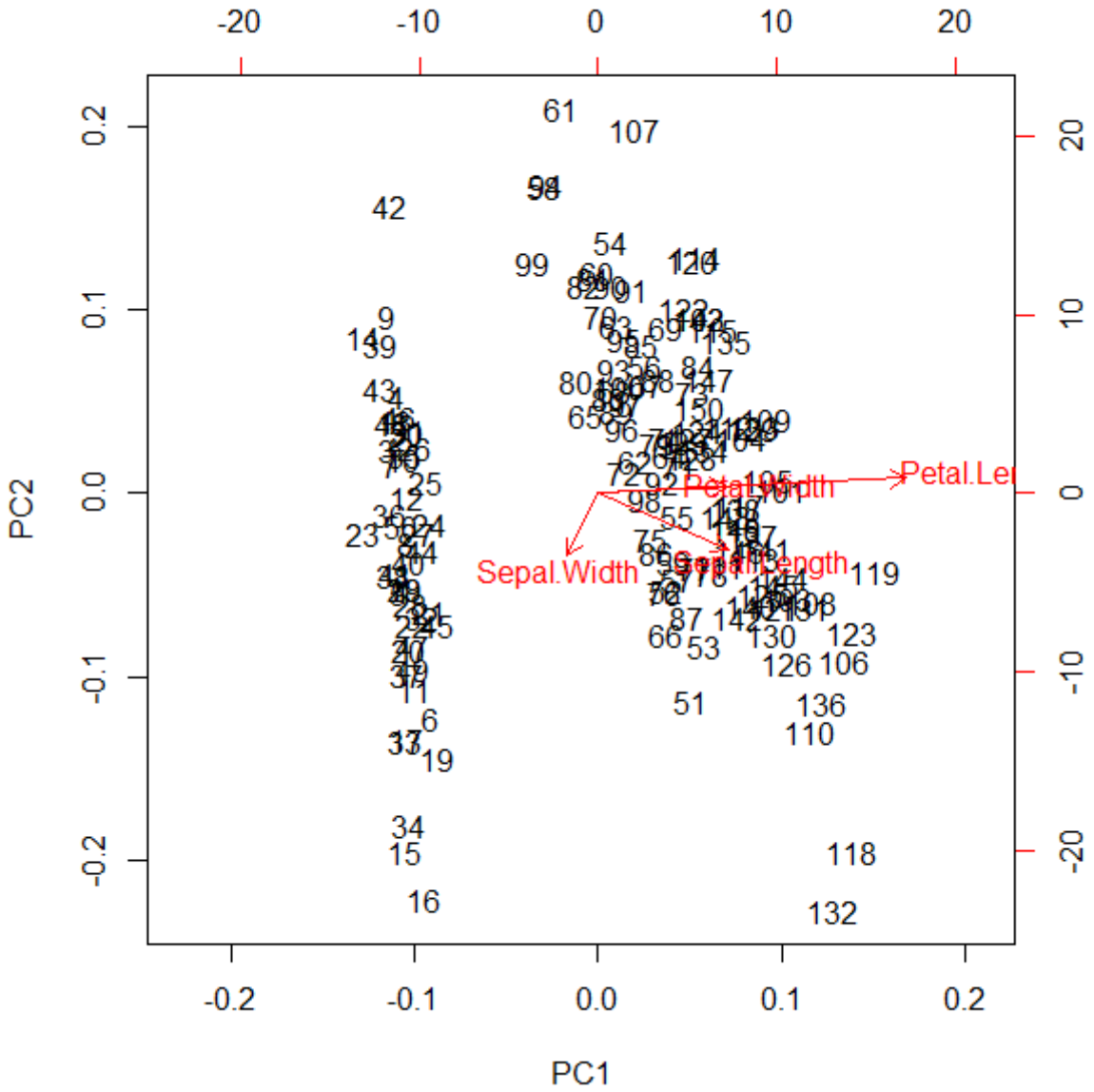
```
##  [78,]  1.35386953 -0.330205463  0.259955701
##  [79,]  0.66258066  0.225173502 -0.085577197
##  [80,] -0.04012419  1.055183583  0.318506304
##  [81,]  0.13035846  1.557055553  0.149482697
##  [82,]  0.02337438  1.567225244  0.240745761
##  [83,]  0.24073180  0.774661195  0.150707074
##  [84,]  1.05755171  0.631726901 -0.104959762
##  [85,]  0.22323093  0.286812663 -0.663028512
##  [86,]  0.42770626 -0.842758920 -0.449129446
##  [87,]  1.04522645 -0.520308714  0.394464890
##  [88,]  1.04104379  1.378371048  0.685997804
##  [89,]  0.06935597  0.218770433 -0.290605718
##  [90,]  0.28253073  1.324886147 -0.089111491
##  [91,]  0.27814596  1.116288852 -0.094172116
##  [92,]  0.62248441 -0.024839814  0.020412763
##  [93,]  0.33540673  0.985103828  0.198724011
##  [94,] -0.36097409  2.012495825 -0.105467721
##  [95,]  0.28762268  0.852873116 -0.130452657
##  [96,]  0.09105561  0.180587142 -0.128547696
##  [97,]  0.22695654  0.383634868 -0.155691572
##  [98,]  0.57446378  0.154356489  0.270743347
##  [99,] -0.44617230  1.538637456 -0.189765199
## [100,]  0.25587339  0.596852285 -0.091572385
## [101,]  1.83841002 -0.867515056 -1.002044077
## [102,]  1.15401555  0.696536401 -0.528389994
## [103,]  2.19790361 -0.560133976  0.202236658
## [104,]  1.43534213  0.046830701 -0.163083761
## [105,]  1.86157577 -0.294059697 -0.394307408
## [106,]  2.74268509 -0.797736709  0.580364827
## [107,]  0.36579225  1.556289178 -0.983598122
## [108,]  2.29475181 -0.418663020  0.649530452
## [109,]  1.99998633  0.709063226  0.392675073
## [110,]  2.25223216 -1.914596301 -0.396224508
## [111,]  1.35962064 -0.690443405 -0.283661780
## [112,]  1.59732747  0.420292431 -0.023108991
## [113,]  1.87761053 -0.417849815 -0.026250468
## [114,]  1.25590769  1.158379741 -0.578311891
## [115,]  1.46274487  0.440794883 -1.000517746
## [116,]  1.58476820 -0.673986887 -0.636297054
```

```
## [117,]  1.46651849 -0.254768327 -0.037306280
## [118,]  2.41822770 -2.548124795  0.127454475
## [119,]  3.29964148 -0.017721580  0.700957033
## [120,]  1.25954707  1.701046715  0.266643612
## [121,]  2.03091256 -0.907427443 -0.234015510
## [122,]  0.97471535  0.569855257 -0.825362161
## [123,]  2.88797650 -0.412259950  0.854558973
## [124,]  1.32878064  0.480202496  0.005410239
## [125,]  1.69505530 -1.010536476 -0.297454114
## [126,]  1.94780139 -1.004412720  0.418582432
## [127,]  1.17118007  0.315338060 -0.129503907
## [128,]  1.01754169 -0.064131184 -0.336588365
## [129,]  1.78237879  0.186735633 -0.269754304
## [130,]  1.85742501 -0.560413289  0.713244682
## [131,]  2.42782030 -0.258418706  0.725386035
## [132,]  2.29723178 -2.617554417  0.491826144
## [133,]  1.85648383  0.177953334 -0.352966242
## [134,]  1.11042770  0.291944582  0.182875741
## [135,]  1.19845835  0.808606364  0.164173760
## [136,]  2.78942561 -0.853942542  0.541093785
## [137,]  1.57099294 -1.065013214 -0.942695700
## [138,]  1.34179696 -0.421020154 -0.180271551
## [139,]  0.92173701 -0.017165594 -0.415434449
## [140,]  1.84586124 -0.673870645  0.012629804
## [141,]  2.00808316 -0.611835930 -0.426902678
## [142,]  1.89543421 -0.687273065 -0.129640697
## [143,]  1.15401555  0.696536401 -0.528389994
## [144,]  2.03374499 -0.864624030 -0.337014969
## [145,]  1.99147547 -1.045665670 -0.630301866
## [146,]  1.86425786 -0.385674038 -0.255418178
## [147,]  1.55935649  0.893692855  0.026283300
## [148,]  1.51609145 -0.268170747 -0.179576781
## [149,]  1.36820418 -1.007877934 -0.930278721
## [150,]  0.95744849  0.024250427 -0.526485033
```

주성분 1과 주성분 2를 이용한 산점도 출력

```
biplot(iris.pca)
```





02. 로지스틱 회귀 분석

로지스틱 회귀분석은 독립 변수들의 선형 결합을 통해 사건의 발생 여부 등의
종속 변수가 이항(0,1)과 같은 값을 분류하기 위한 목적으로 사용되는 통계기법이다.

로지스틱 회귀 분석은 선형 회귀 분석과는 다르게 종속변수가 범주형인 데이터인 경우에 사용.

```
head(mtcars)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

- mpg: 연비 (Miles/(US) gallon)
- cyl: 실린더 개수 (Number of cylinders)
- disp: 배기량 (Displacement (cu.in.))
- hp: 마력 (Gross horsepower)
- drat: 후방차축 비율 (Rear axle ratio)
- wt: 무게 (Weight (1,000 lbs))
- qsec: 1/4 마일에 도달하는데 소요되는 시간 (1/4 mile time)
- vs: 엔진 (0 = V engine, 1 = S engine)
- am: 변속기 (0 = 자동, 1 = 수동)
- gear: 기어 개수 (Number of forward gears)
- carb: 기화기 개수 (Number of carburetors)

2-1 일부 데이터 선택

- mpg: 연비 (Miles/(US) gallon)
- vs: 엔진 (0 = V engine, 1 = S engine)
- am: 변속기 (0 = 자동, 1 = 수동)

```
dat1 <- subset(mtcars, select=c(mpg, am, vs))
head(dat1)
```

```
##                   mpg am vs
## Mazda RX4         21.0  1  0
## Mazda RX4 Wag     21.0  1  0
## Datsun 710        22.8  1  1
## Hornet 4 Drive    21.4  0  1
## Hornet Sportabout 18.7  0  0
## Valiant           18.1  0  1
```

2-2 로지스틱 회귀분석 모델 생성

```
vs : 종속변수
mpg : 연속형 독립변수
am  : 범주형 독립변수
```

```
log_reg <- glm(vs ~ mpg + am, data=dat1, family="binomial") # 로지스틱 회귀분석 실행
log_reg
```

```
##
## Call:  glm(formula = vs ~ mpg + am, family = "binomial", data = dat1)
##
## Coefficients:
## (Intercept)          mpg           am
##    -12.7051       0.6809      -3.0073
##
## Degrees of Freedom: 31 Total (i.e. Null);  29 Residual
## Null Deviance:       43.86
## Residual Deviance: 20.65     AIC: 26.65
```

회귀분석 모델 요약 정보 확인

```
summary(log_reg)
```

```
##
## Call:
## glm(formula = vs ~ mpg + am, family = "binomial", data = dat1)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.05888  -0.44544  -0.08765   0.33335   1.68405
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.7051     4.6252  -2.747  0.00602 **
## mpg           0.6809     0.2524   2.698  0.00697 **
## am           -3.0073     1.5995  -1.880  0.06009 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 43.860  on 31  degrees of freedom
## Residual deviance: 20.646  on 29  degrees of freedom
## AIC: 26.646
##
## Number of Fisher Scoring iterations: 6
```