Data Preprocessing

Park Ju ho

2022 4 11

```
library(caret)
```

```
data(mdrr)
data.frame(table(mdrrDescr$nR11))
```

```
##   Var1 Freq
## 1    0  501
## 2    1    4
## 3    2   23
```

영분산 측정

freqRatio = 일 순위 빈발값의 빈도/차 순위 빈발값의 빈도 => 정상적일 수록 1에 가깝고 클수록 불균형

percentUnique = 유일한 값들의 수/전체 표본 수 => 0에 가까울 수록 영분산

nearZeroVar에선 유일 값 비율이 10%, 빈도비율이 19보다 큰 예측 변수를 영분산이로 간주

```
nzv = nearZeroVar(mdrrDescr,saveMetrics = TRUE)
#saveMetrics
str(nzv)
```

```
## 'data.frame':    342 obs. of  4 variables:
##  $ freqRatio    : num  1.25 1.12 1 1.25 1.25 ...
##  $ percentUnique: num  90 42.6 83 84.3 82.8 ...
##  $ zeroVar      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  $ nzv          : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

```
nzv[nzv$nzv,]#0
```

```
##            freqRatio percentUnique zeroVar  nzv
## nTB         23.00000     0.3787879   FALSE TRUE
## nBR        131.00000     0.3787879   FALSE TRUE
## nI         527.00000     0.3787879   FALSE TRUE
## nR03       527.00000     0.3787879   FALSE TRUE
## nR08       527.00000     0.3787879   FALSE TRUE
## nR11        21.78261     0.5681818   FALSE TRUE
## nR12        57.66667     0.3787879   FALSE TRUE
## D.Dr03     527.00000     0.3787879   FALSE TRUE
## D.Dr07    123.50000     5.8712121   FALSE TRUE
## D.Dr08    527.00000     0.3787879   FALSE TRUE
## D.Dr09    479.00000     9.4696970   FALSE TRUE
## D.Dr11    125.25000     4.5454545   FALSE TRUE
## D.Dr12    519.00000     1.8939394   FALSE TRUE
## T.N..S.    35.07692     5.4924242   FALSE TRUE
## T.N..F.    94.00000     6.0606061   FALSE TRUE
## T.N..Cl.   43.20000     7.1969697   FALSE TRUE
## T.N..Br.  262.00000     0.7575758   FALSE TRUE
## T.N..I.   527.00000     0.3787879   FALSE TRUE
## T.O..S.    80.50000     4.7348485   FALSE TRUE
## T.O..F.    68.00000     5.6818182   FALSE TRUE
## T.O..Cl.   50.22222     6.8181818   FALSE TRUE
## T.O..Br.  262.50000     0.5681818   FALSE TRUE
## T.O..I.   527.00000     0.3787879   FALSE TRUE
## T.S..S.    65.00000     0.3787879   FALSE TRUE
## T.S..F.   130.00000     0.9469697   FALSE TRUE
## T.S..Cl.   42.41667     1.5151515   FALSE TRUE
## T.F..F.    50.80000     2.0833333   FALSE TRUE
## T.F..Cl.  173.33333     1.3257576   FALSE TRUE
## T.Cl..Cl.  45.81818     2.4621212   FALSE TRUE
## T.Cl..Br. 527.00000     0.3787879   FALSE TRUE
## T.I..I.   527.00000     0.3787879   FALSE TRUE
## G.N..Br.  262.00000     0.7575758   FALSE TRUE
## G.N..I.   527.00000     0.3787879   FALSE TRUE
## G.O..S.   161.00000     7.1969697   FALSE TRUE
## G.O..F.   158.66667     8.7121212   FALSE TRUE
## G.O..Br.  262.50000     0.5681818   FALSE TRUE
## G.O..I.   527.00000     0.3787879   FALSE TRUE
## G.S..S.   260.00000     1.3257576   FALSE TRUE
## G.S..F.   260.00000     1.5151515   FALSE TRUE
## G.S..Cl.  169.66667     2.6515152   FALSE TRUE
## G.F..F.   101.60000     3.2196970   FALSE TRUE
## G.F..Cl.  520.00000     1.7045455   FALSE TRUE
## G.Cl..Cl. 168.00000     3.5984848   FALSE TRUE
## G.Cl..Br. 527.00000     0.3787879   FALSE TRUE
## G.I..I.   527.00000     0.3787879   FALSE TRUE
```

```r
dim(mdrrDescr)
```

```
## [1] 528 342
```

```r
nzv = nearZeroVar(mdrrDescr)#saveMetrics     index
nzv
```

```
##  [1]  22  31  32  34  38  41  42 259 262 263 264 266 267 270 271 272 273 274 276
## [20] 277 278 279 280 281 282 283 284 285 286 287 288 327 328 330 331 333 334 335
## [39] 336 337 338 339 340 341 342
```

```r
filteredDescr <- mdrrDescr[, -nzv]
dim(filteredDescr)
```

```
## [1] 528 297
```

중복 변수 제거

```
descrCor = cor(filteredDescr)
sum(abs(descrCor[upper.tri(descrCor)])>.999)
```

```
## [1] 65
```

```
#                  0.999

summary(descrCor[upper.tri(descrCor)])
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
## -0.99607 -0.05373  0.25006  0.26078  0.65527  1.00000
```

```
higlyCorDescr = findCorrelation(descrCor,cutoff=0.75)
higlyCorDescr
```

```
##   [1]   5  11  12  13  14  16  23  24  30  37  38  39  40  42  43  44  45  47
##  [19]  49  50  51  52  53  55  56  57  58  59  61  62  63  64  65  66  68  69
##  [37]  70  71  72  73  74  75  76  77  78  79  83  84  85  88  90  91  92  93
##  [55]  94  95  96  97  98  99 100 101 102 103 104 105 106 110 111 112 113 114
##  [73] 115 116 117 118 119 120 121 122 123 124 125 126 127 132 134 135 136 137
##  [91] 138 139 140 141 144 145 146 148 149 150 152 153 154 155 156 157 158 159
## [109] 160 161 162 164 165 167 169 170 172 174 175 176 177 178 179 180 181 182
## [127] 183 184 185 186 187 189 190 191 192 193 194 195 196 197 198 199 200 202
## [145] 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221
## [163] 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239
## [181] 240 246 249 250 251 252 253 254 258 259 261 262 263 265 266 267 274 277
## [199] 278 279 280 281 282 284 285 286 287 288 289 290 293 294 295 296   1   3
## [217]   4   7   8  17  19  15   6  20  41  80  81  18 108 109  54 163 166 168
## [235] 171 147 241 242 243 244 247  25  26  67 270 255 256
```

```
#    0.75
filteredDescr = filteredDescr[,-higlyCorDescr]
#
descrCor2 = cor(filteredDescr)
summary(descrCor2[upper.tri(descrCor2)])
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
## -0.70728 -0.05378  0.04418  0.06692  0.18858  0.74458
```

중심화와 척도화

```
set.seed(200)
inTrain = sample(seq(along = mdrrClass), length(mdrrClass)/2)
#seq(along = ) along     seq   , 1:1    Test,Train
```

```r
training = filteredDescr[inTrain,]
test = filteredDescr[-inTrain,]
trainMDRR = mdrrClass[inTrain]
testMDRR = mdrrClass[-inTrain]

preProcValues = preProcess(training,method = c("center","scale"))
#   ,
trainTransformed = predict(preProcValues,training)
testTransformed = predict(preProcValues,test)
head(training)
```

```
##              AMW   Mp   Ms nDB nAB nS nF nCL nR05 nR06 nR07 nR09 nR10 nBnz  HNar
## SKF-3301   5.99 0.64 1.89   0  12  0  0   0    0    2    0    0    0    2 1.878
## BERBERINE  7.82 0.68 2.06   1  16  0  0   0    1    4    0    1    3    2 2.113
## BEVANTOLOL 6.64 0.63 2.19   0  12  0  0   0    0    2    0    0    0    2 1.852
## ROPITOIN   7.01 0.66 2.14   2  18  0  0   0    1    4    0    0    0    3 2.028
## PINOXEPINE 7.11 0.67 2.02   0  15  0  0   1    0    3    1    0    0    2 2.049
## PROZAPINE  5.99 0.65 1.76   0  12  0  0   0    0    2    1    0    0    2 2.129
##            Xt     SPI Jhetm MAXDN MAXDP     TIE   X5v   BLI   PW2   PW3   PW4
## SKF-3301    0  75.319 2.447 1.316 2.630  90.635 2.134 1.065 0.548 0.317 0.188
## BERBERINE   0  34.322 1.945 1.201 2.096 100.021 2.805 0.858 0.588 0.374 0.217
## BEVANTOLOL  0 190.105 1.833 1.883 4.003 140.781 1.622 0.971 0.556 0.319 0.165
## ROPITOIN    0 146.755 1.325 2.487 6.873 171.757 3.953 0.955 0.576 0.355 0.198
## PINOXEPINE  0  36.318 1.650 1.180 3.132 148.285 3.361 1.032 0.567 0.333 0.189
## PROZAPINE   0   4.690 1.766 0.814 0.673  93.233 2.572 1.094 0.549 0.314 0.176
##            PJI2 BAC   Lop  IVDE  BIC2  BIC5  VEA1     VRA1   piPC10    PCR
## SKF-3301    1.0  21 1.783 1.781 0.595 0.739 3.803  303.441  385.172  5.695
## BERBERINE   1.0   9 0.769 1.791 0.733 0.872 4.565  182.523 8240.854 56.099
## BEVANTOLOL  1.0  21 0.900 1.939 0.739 0.877 3.512  728.459  136.688  6.979
## ROPITOIN    0.9  11 0.586 1.637 0.614 0.841 4.068 4623.585 1017.953  7.524
## PINOXEPINE  1.0   7 0.787 1.804 0.719 0.888 4.137  944.235 2643.902 23.147
## PROZAPINE   1.0   0 0.000 1.322 0.556 0.711 4.008  192.862  469.797  5.143
##            T.O..O.     H3D      G1  SPAM   SPH   FDI  PJI3 L.Bw DISPm   QXXm
## SKF-3301         0 226.670  68.614 0.314 0.920 0.677 0.953  3.1 6.458 47.096
## BERBERINE       43 130.876  73.173 0.365 0.952 0.757 0.936  6.8 4.371 35.326
## BEVANTOLOL      46 158.529  62.736 0.363 0.952 0.695 0.857  6.3 1.569 59.853
## ROPITOIN        19 304.815 115.141 0.350 0.972 0.714 0.912  9.6 3.812 73.573
## PINOXEPINE      13 206.001  83.168 0.342 0.939 0.710 0.890  3.4 9.612 95.132
## PROZAPINE        0 379.564  58.600 0.320 0.912 0.703 0.910  2.6 5.574 54.104
##            DISPe G.N..N. G.N..O. G.O..Cl.
## SKF-3301   0.030    0.00    2.69     0.00
## BERBERINE  0.104    0.00   16.51     0.00
## BEVANTOLOL 0.111    0.00   16.62     0.00
## ROPITOIN   0.145    8.28   27.74     0.00
## PINOXEPINE 0.090    0.00   10.37    12.58
## PROZAPINE  0.026    0.00    0.00     0.00
```

```r
head(trainTransformed)
```

```
##                     AMW         Mp         Ms        nDB        nAB         nS
## SKF-3301   -1.405904262 -0.4757759 -1.0994929 -0.8918723 -0.2587980 -0.4057484
## BERBERINE   1.125194926  0.9276301 -0.2700217  0.3473608  0.7571525 -0.4057484
## BEVANTOLOL -0.506879960 -0.8266274  0.3642798 -0.8918723 -0.2587980 -0.4057484
## ROPITOIN    0.004872335  0.2259271  0.1203177  1.5865938  1.2651278 -0.4057484
## PINOXEPINE  0.143183766  0.5767786 -0.4651914 -0.8918723  0.5031649 -0.4057484
## PROZAPINE  -1.405904262 -0.1249244 -1.7337943 -0.8918723 -0.2587980 -0.4057484
##                    nF        nCL       nR05        nR06       nR07       nR09
## SKF-3301   -0.2682069 -0.3992357 -0.5144837 -0.88966546 -0.2107378 -0.2623803
## BERBERINE  -0.2682069 -0.3992357  1.5127356  0.98182204 -0.2107378  2.7492888
## BEVANTOLOL -0.2682069 -0.3992357 -0.5144837 -0.88966546 -0.2107378 -0.2623803
## ROPITOIN   -0.2682069 -0.3992357  1.5127356  0.98182204 -0.2107378 -0.2623803
## PINOXEPINE -0.2682069  1.2737519 -0.5144837  0.04607829  4.0688599 -0.2623803
## PROZAPINE  -0.2682069 -0.3992357 -0.5144837 -0.88966546  4.0688599 -0.2623803
##                  nR10      nBnz       HNar         Xt        SPI       Jhetm
## SKF-3301   -0.6683224 0.3030348 -0.6113610 -0.2049629 -0.1420673  1.25480315
## BERBERINE   2.8370508 0.3030348  1.3095727 -0.2049629 -0.1438803  0.08713788
## BEVANTOLOL -0.6683224 0.3030348 -0.8238898 -0.2049629 -0.1369914 -0.17337708
## ROPITOIN   -0.6683224 1.6145297  0.6147669 -0.2049629 -0.1389084 -1.35499850
## PINOXEPINE -0.6683224 0.3030348  0.7864248 -0.2049629 -0.1437920 -0.59903991
## PROZAPINE  -0.6683224 0.3030348  1.4403597 -0.2049629 -0.1451906 -0.32922085
##                 MAXDN       MAXDP        TIE        X5v        BLI         PW2
## SKF-3301   -0.4070974 -0.69789671 -0.7486343 -0.5586002  0.8098268 -1.41298913
## BERBERINE  -0.5433005 -0.99527311 -0.6641345  0.1232677 -1.9514768  1.36977201
## BEVANTOLOL  0.2644431  0.06670592 -0.2971822 -1.0788929 -0.4440985 -0.85643690
## ROPITOIN    0.9798055  1.66496488 -0.0183128  1.2898615 -0.6575326  0.53494367
## PINOXEPINE -0.5681724 -0.41834061 -0.2296255  0.6882731  0.3696189 -0.09117759
## PROZAPINE  -1.0016536 -1.78771998 -0.7252452 -0.1135061  1.1966760 -1.34342010
##                    PW3        PW4       PJI2         BAC         Lop       IVDE
## SKF-3301   -0.86268283  0.4103976  1.0062838 -0.09181354  1.46243127  0.1670429
## BERBERINE   2.08702206  2.1608177  1.0062838 -0.55350447 -0.29254919  0.2156265
## BEVANTOLOL -0.75918441 -0.9778666  1.0062838 -0.09181354 -0.06582095  0.9346636
## ROPITOIN    1.10378710  1.0139907 -0.2728289 -0.47655598 -0.60927643 -0.5325608
## PINOXEPINE -0.03469549  0.4707569  1.0062838 -0.63045296 -0.26139569  0.2787851
## PROZAPINE  -1.01793045 -0.3139142  1.0062838 -0.89977267 -1.62349591 -2.0629438
##                  BIC2       BIC5        VEA1        VRA1     piPC10        PCR
## SKF-3301   -0.8188862 -1.1766428 -0.75237830 -0.06176831 -0.5380879 -0.4989373
## BERBERINE   1.3329936  1.1103366  0.83936860 -0.06180195  3.9126936  2.5869826
## BEVANTOLOL  1.4265536  1.1963132 -1.36025014 -0.06165008 -0.6788711 -0.4203261
## ROPITOIN   -0.5226129  0.5772812 -0.19881802 -0.06056652 -0.1795742 -0.3869591
## PINOXEPINE  1.1146870  1.3854619 -0.05468346 -0.06159006  0.7416372  0.5695389
## PROZAPINE  -1.4270261 -1.6581121 -0.32415243 -0.06179907 -0.4901421 -0.5327328
##               T.O..O.         H3D         G1       SPAM         SPH         FDI
## SKF-3301   -0.4397476 -0.22440005 -0.3377720 -1.2445246 -0.73295194 -1.32040065
## BERBERINE   0.2817461 -0.31961359 -0.2728433  0.7384670  0.37137800  2.09626949
## BEVANTOLOL  0.3320829 -0.29212815 -0.4214857  0.6607026  0.37137800 -0.55164987
## ROPITOIN   -0.1209481 -0.14672857  0.3248594  0.1552342  1.06158422  0.25980929
## PINOXEPINE -0.2216216 -0.24494381 -0.1304958 -0.1558233 -0.07725604  0.08897578
## PROZAPINE  -0.4397476 -0.07243251 -0.4803900 -1.0112315 -1.00903443 -0.20998285
##                   PJI3        L.Bw      DISPm       QXXm       DISPe    G.N..N.
## SKF-3301    0.88832219 -0.62073150  0.2349384 -0.5235121 -0.91480323 -0.5835681
## BERBERINE   0.64433259 -0.07463537 -0.2905734 -0.8059216 -0.13489505 -0.5835681
## BEVANTOLOL -0.48950141 -0.14843215 -0.9961241 -0.2174206 -0.06111995 -0.5835681
## ROPITOIN    0.29987669  0.33862656 -0.4313311  0.1117772  0.29721623 -0.1413708
## PINOXEPINE -0.01587455 -0.57645343  1.0291236  0.6290640 -0.28244525 -0.5835681
## PROZAPINE   0.27117204 -0.69452827  0.0123450 -0.3553621 -0.95696042 -0.5835681
##                G.N..O.   G.O..Cl.
## SKF-3301   -0.68364828 -0.2946877
## BERBERINE  -0.32740543 -0.2946877
## BEVANTOLOL -0.32456992 -0.2946877
## ROPITOIN   -0.03792589 -0.2946877
## PINOXEPINE -0.48567830  1.1312096
## PROZAPINE  -0.75298932 -0.2946877
```

box-cox

등분산 가정을 위하여

```
preProcValues2 = preProcess(training,method = "BoxCox")
trainBC = predict(preProcValues2,training)
testBC = predict(preProcValues2,test)
preProcValues2
```

```
## Created from 264 samples and 31 variables
##
## Pre-processing:
##   - Box-Cox transformation (31)
##   - ignored (0)
##
## Lambda estimates for Box-Cox transformation:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.0000 -0.2500  0.4000  0.4548  1.4500  2.0000
```

```
head(training)
```

```
##              AMW   Mp   Ms nDB nAB nS nF nCL nR05 nR06 nR07 nR09 nR10 nBnz  HNar
## SKF-3301    5.99 0.64 1.89   0  12  0  0   0    0    2    0    0    0    2 1.878
## BERBERINE   7.82 0.68 2.06   1  16  0  0   0    1    4    0    1    3    2 2.113
## BEVANTOLOL  6.64 0.63 2.19   0  12  0  0   0    0    2    0    0    0    2 1.852
## ROPITOIN    7.01 0.66 2.14   2  18  0  0   0    1    4    0    0    0    3 2.028
## PINOXEPINE  7.11 0.67 2.02   0  15  0  0   1    0    3    1    0    0    2 2.049
## PROZAPINE   5.99 0.65 1.76   0  12  0  0   0    0    2    1    0    0    2 2.129
##             Xt     SPI Jhetm MAXDN MAXDP     TIE   X5v   BLI   PW2   PW3   PW4
## SKF-3301     0  75.319 2.447 1.316 2.630  90.635 2.134 1.065 0.548 0.317 0.188
## BERBERINE    0  34.322 1.945 1.201 2.096 100.021 2.805 0.858 0.588 0.374 0.217
## BEVANTOLOL   0 190.105 1.833 1.883 4.003 140.781 1.622 0.971 0.556 0.319 0.165
## ROPITOIN     0 146.755 1.325 2.487 6.873 171.757 3.953 0.955 0.576 0.355 0.198
## PINOXEPINE   0  36.318 1.650 1.180 3.132 148.285 3.361 1.032 0.567 0.333 0.189
## PROZAPINE    0   4.690 1.766 0.814 0.673  93.233 2.572 1.094 0.549 0.314 0.176
##            PJI2 BAC   Lop  IVDE  BIC2  BIC5  VEA1     VRA1   piPC10    PCR
## SKF-3301    1.0  21 1.783 1.781 0.595 0.739 3.803  303.441  385.172  5.695
## BERBERINE   1.0   9 0.769 1.791 0.733 0.872 4.565  182.523 8240.854 56.099
## BEVANTOLOL  1.0  21 0.900 1.939 0.739 0.877 3.512  728.459  136.688  6.979
## ROPITOIN    0.9  11 0.586 1.637 0.614 0.841 4.068 4623.585 1017.953  7.524
## PINOXEPINE  1.0   7 0.787 1.804 0.719 0.888 4.137  944.235 2643.902 23.147
## PROZAPINE   1.0   0 0.000 1.322 0.556 0.711 4.008  192.862  469.797  5.143
##            T.O..O.     H3D     G1  SPAM   SPH   FDI  PJI3 L.Bw DISPm   QXXm
## SKF-3301         0 226.670  68.614 0.314 0.920 0.677 0.953  3.1 6.458 47.096
## BERBERINE       43 130.876  73.173 0.365 0.952 0.757 0.936   6.8 4.371 35.326
## BEVANTOLOL      46 158.529  62.736 0.363 0.952 0.695 0.857   6.3 1.569 59.853
## ROPITOIN        19 304.815 115.141 0.350 0.972 0.714 0.912   9.6 3.812 73.573
## PINOXEPINE      13 206.001  83.168 0.342 0.939 0.710 0.890   3.4 9.612 95.132
## PROZAPINE        0 379.564  58.600 0.320 0.912 0.703 0.910   2.6 5.574 54.104
##            DISPe G.N..N. G.N..O. G.O..Cl.
## SKF-3301   0.030    0.00    2.69     0.00
## BERBERINE  0.104    0.00   16.51     0.00
## BEVANTOLOL 0.111    0.00   16.62     0.00
## ROPITOIN   0.145    8.28   27.74     0.00
## PINOXEPINE 0.090    0.00   10.37    12.58
## PROZAPINE  0.026    0.00    0.00     0.00
```

```r
head(trainBC)
```

```
##                  AMW         Mp        Ms nDB nAB nS nF nCL nR05      nR06 nR07
## SKF-3301   0.4860647 -0.7207031 0.4100907   0  12  0  0   0    0 0.8595276    0
## BERBERINE  0.4918237 -0.5813149 0.4411867   1  16  0  0   0    1 2.1623278    0
## BEVANTOLOL 0.4886595 -0.7597632 0.4609627   0  12  0  0   0    0 0.8595276    0
## ROPITOIN   0.4898250 -0.6478421 0.4537115   2  18  0  0   0    1 2.1623278    0
## PINOXEPINE 0.4901092 -0.6138338 0.4344563   0  15  0  0   1    0 1.5553034    1
## PROZAPINE  0.4860647 -0.6834320 0.3811446   0  12  0  0   0    0 0.8595276    1
##            nR09 nR10 nBnz      HNar Xt      SPI     Jhetm      MAXDN MAXDP
## SKF-3301      0    0    2 1.0117341  0 4.321732 0.8948628  0.2862244 2.630
## BERBERINE     1    3    2 1.3214962  0 3.535787 0.6652620  0.1882798 2.096
## BEVANTOLOL    0    0    2 0.9783728  0 5.247577 0.6059540  0.6969339 4.003
## ROPITOIN      0    0    3 1.2077750  0 4.988765 0.2814125  1.0477497 6.873
## PINOXEPINE    0    0    2 1.2356968  0 3.592313 0.5007753  0.1696926 3.132
## PROZAPINE     0    0    2 1.3431103  0 1.545433 0.5687171 -0.1995709 0.673
##                 TIE       X5v         BLI        PW2        PW3        PW4
## SKF-3301   4.506840 0.9597607  0.06065413 -0.4065593 -0.5476802 -0.9851543
## BERBERINE  4.605380 1.4279615 -0.16812759 -0.3746603 -0.5141855 -0.9383145
## BEVANTOLOL 4.947205 0.5611490 -0.02995461 -0.4002517 -0.5465524 -1.0238644
## ROPITOIN   5.146081 2.1352697 -0.04733972 -0.3843241 -0.5256562 -0.9687729
## PINOXEPINE 4.999136 1.7826245  0.03091080 -0.3915194 -0.5385590 -0.9835046
## PROZAPINE  4.535102 1.2710596  0.08516733 -0.4057729 -0.5493653 -1.0051617
##              PJI2 BAC   Lop      IVDE  BIC2       BIC5      VEA1     VRA1
## SKF-3301    0.000  21 1.783 1.0859805 0.595 -0.2269395 0.6999359 2.733188
## BERBERINE   0.000   9 0.769 1.1038405 0.733 -0.1198080 0.7380018 2.634323
## BEVANTOLOL  0.000  21 0.900 1.3798605 0.739 -0.1154355 0.6807954 2.871849
## ROPITOIN   -0.095  11 0.586 0.8398845 0.614 -0.1463595 0.7148734 3.068248
## PINOXEPINE  0.000   7 0.787 1.1272080 0.719 -0.1057280 0.7184337 2.906405
## PROZAPINE   0.000   0 0.000 0.3738420 0.556 -0.2472395 0.7116729 2.645783
##              piPC10      PCR T.O..O.      H3D       G1       SPAM        SPH
## SKF-3301    385.172 1.355314       0 1.867159 4.228497 -0.4507020 -0.0768000
## BERBERINE  8240.854 2.337487      43 1.825176 4.292827 -0.4333875 -0.0468480
## BEVANTOLOL  136.688 1.472357      46 1.841154 4.138935 -0.4341155 -0.0468480
## ROPITOIN   1017.953 1.513867      19 1.885446 4.746157 -0.4387500 -0.0276080
## PINOXEPINE 2643.902 2.034569      13 1.860654 4.420863 -0.4415180 -0.0591395
## PROZAPINE   469.797 1.293880       0 1.897343 4.070735 -0.4488000 -0.0841280
##                  FDI       PJI3      L.Bw     DISPm     QXXm DISPe G.N..N.
## SKF-3301   -0.2708355 -0.0458955 1.1314021 2.7720464 7.253692 0.030    0.00
## BERBERINE  -0.2134755 -0.0619520 1.9169226 2.0099563 6.378631 0.104    0.00
## BEVANTOLOL -0.2584875 -0.1327755 1.8405496 0.4935684 8.043058 0.111    0.00
## ROPITOIN   -0.2451020 -0.0841280 2.2617631 1.7697374 8.769702 0.145    8.28
## PINOXEPINE -0.2479500 -0.1039500 1.2237754 3.6810964 9.739711 0.090    0.00
## PROZAPINE  -0.2528955 -0.0859500 0.9555114 2.4705769 7.703578 0.026    0.00
##            G.N..O. G.O..Cl.
## SKF-3301      2.69     0.00
## BERBERINE    16.51     0.00
## BEVANTOLOL   16.62     0.00
## ROPITOIN     27.74     0.00
## PINOXEPINE   10.37    12.58
## PROZAPINE     0.00     0.00
```

#더비변수 생성 범주형 변수를 원-핫 벡터로 바꾸는 것

```
library(earth)
data(etitanic)
str(etitanic)
```

```
## 'data.frame':    1046 obs. of  6 variables:
##  $ pclass  : Factor w/ 3 levels "1st","2nd","3rd": 1 1 1 1 1 1 1 1 1 1 ...
##  $ survived: int  1 1 0 0 0 1 1 0 1 0 ...
##  $ sex     : Factor w/ 2 levels "female","male": 1 2 1 2 1 2 1 2 1 2 ...
##  $ age     : num  29 0.917 2 30 25 ...
##  $ sibsp   : int  0 1 1 1 1 0 1 0 2 0 ...
##  $ parch   : int  0 2 2 2 2 0 0 0 0 0 ...
```

```
head(etitanic)
```

```
##   pclass survived    sex     age sibsp parch
## 1    1st        1 female 29.0000     0     0
## 2    1st        1   male  0.9167     1     2
## 3    1st        0 female  2.0000     1     2
## 4    1st        0   male 30.0000     1     2
## 5    1st        0 female 25.0000     1     2
## 6    1st        1   male 48.0000     0     0
```

```
head(model.matrix(survived~.,data=etitanic))
```

```
##   (Intercept) pclass2nd pclass3rd sexmale     age sibsp parch
## 1           1         0         0       0 29.0000     0     0
## 2           1         0         0       1  0.9167     1     2
## 3           1         0         0       0  2.0000     1     2
## 4           1         0         0       1 30.0000     1     2
## 5           1         0         0       0 25.0000     1     2
## 6           1         0         0       1 48.0000     0     0
```

```
#matrix     dummy           matrix
dummy.1 = dummyVars(survived~.,data=etitanic)
head(predict(dummy.1,newdata = etitanic))
```

```
##   pclass.1st pclass.2nd pclass.3rd sex.female sex.male     age sibsp parch
## 1          1          0          0          1        0 29.0000     0     0
## 2          1          0          0          0        1  0.9167     1      2
## 3          1          0          0          1        0  2.0000     1      2
## 4          1          0          0          0        1 30.0000     1      2
## 5          1          0          0          1        0 25.0000     1      2
## 6          1          0          0          0        1 48.0000     0      0
```

#선형 종속성 3,1,2와 6,1,4,5들끼리 선형 종속을 이루고 있음 이를 해결하기 위하여 3번과 6번 열을 제거하면 됨

```
ltfrDesign <- matrix(0, nrow = 6, ncol = 6)
ltfrDesign[, 1] <- c(1, 1, 1, 1, 1, 1)
ltfrDesign[, 2] <- c(1, 1, 1, 0, 0, 0)
ltfrDesign[, 3] <- c(0, 0, 0, 1, 1, 1)
ltfrDesign[, 4] <- c(1, 0, 0, 1, 0, 0)
ltfrDesign[, 5] <- c(0, 1, 0, 0, 1, 0)
ltfrDesign[, 6] <- c(0, 0, 1, 0, 0, 1)

comboinfo = findLinearCombos(ltfrDesign)
comboinfo
```

```
## $linearCombos
## $linearCombos[[1]]
## [1] 3 1 2
##
## $linearCombos[[2]]
## [1] 6 1 4 5
##
##
## $remove
## [1] 3 6
```

```
ltfrDesign[,-comboinfo$remove]
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    1    1    0
## [2,]    1    1    0    1
## [3,]    1    1    0    0
## [4,]    1    0    1    0
## [5,]    1    0    0    1
## [6,]    1    0    0    0
```

#결측값 대치

```
library(caret)
data("airquality")
summary(airquality)
```

```
##      Ozone           Solar.R           Wind             Temp
##  Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00
##  1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00
##  Median : 31.50   Median :205.0   Median : 9.700   Median :79.00
##  Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88
##  3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00
##  Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00
##  NA's   :37       NA's   :7
##      Month            Day
##  Min.   :5.000   Min.   : 1.0
##  1st Qu.:6.000   1st Qu.: 8.0
##  Median :7.000   Median :16.0
##  Mean   :6.993   Mean   :15.8
##  3rd Qu.:8.000   3rd Qu.:23.0
##  Max.   :9.000   Max.   :31.0
##
```

```
#
imp.1 = preProcess(airquality,method = c("knnImpute"))
#KNN
library(RANN)
imp.2 = predict(imp.1,airquality)
summary(imp.2)
```

```
##      Ozone              Solar.R              Wind               Temp
##  Min.   :-1.24680   Min.   :-1.98684   Min.   :-2.3439   Min.   :-2.3119
##  1st Qu.:-0.67083   1st Qu.:-0.75430   1st Qu.:-0.7259   1st Qu.:-0.6215
##  Median :-0.24643   Median : 0.13401   Median :-0.0731   Median : 0.1181
##  Mean   : 0.00666   Mean   :-0.00895   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.63268   3rd Qu.: 0.77803   3rd Qu.: 0.4378   3rd Qu.: 0.7520
##  Max.   : 3.81566   Max.   : 1.64414   Max.   : 3.0492   Max.   : 2.0198
##      Month                 Day
##  Min.   :-1.407294   Min.   :-1.67002
##  1st Qu.:-0.701340   1st Qu.:-0.88035
##  Median : 0.004614   Median : 0.02212
##  Mean   : 0.000000   Mean   : 0.00000
##  3rd Qu.: 0.710568   3rd Qu.: 0.81178
##  Max.   : 1.416522   Max.   : 1.71426
```

```
#
```

#군집거리 계산

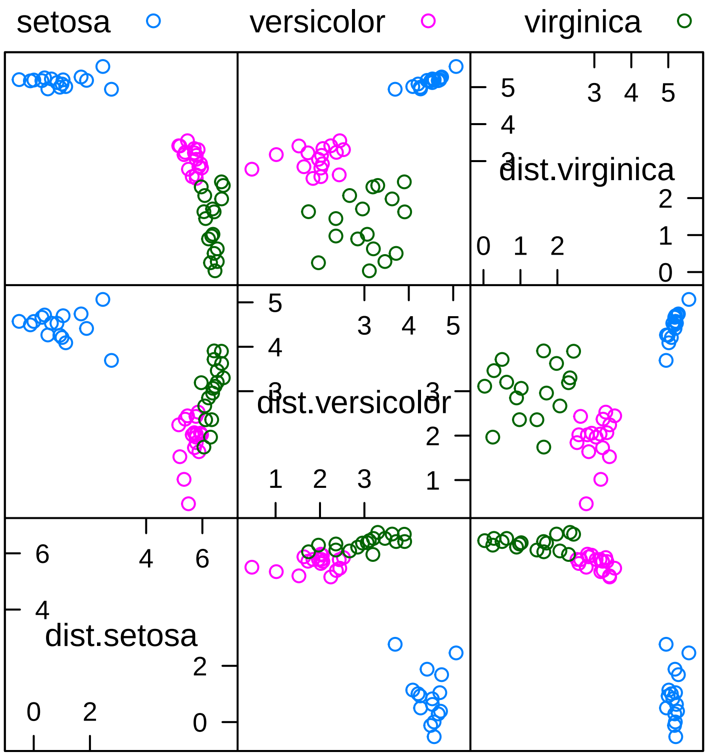
```
trainSet = sample(1:150,100)
#100:50  train,test
distData = classDist(iris[trainSet,1:4],iris$Species[trainSet])
#
distData$values
```

```
## $setosa
## $setosa$means
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##    4.9558824    3.4294118    1.4647059    0.2382353
##
## $setosa$A
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length   20.2608477  -13.658011    -7.699669  -0.7552542
## Sepal.Width   -13.6580115   17.586335     4.274466  -8.9586461
## Petal.Length   -7.6996694    4.274466    32.654508 -16.0113441
## Petal.Width    -0.7552542   -8.958646   -16.011344  99.3825581
##
##
## $versicolor
## $versicolor$means
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##     5.824242     2.784848     4.221212     1.354545
##
## $versicolor$A
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length    12.388139   -2.185631    -8.637163    3.528737
## Sepal.Width     -2.185631   20.195145     1.031926  -15.217328
## Petal.Length    -8.637163    1.031926    21.414366  -36.108266
## Petal.Width      3.528737  -15.217328   -36.108266  113.155926
##
##
## $virginica
## $virginica$means
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##     6.624242     2.984848     5.584848     2.042424
##
## $virginica$A
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length    14.009600   -4.911715   -13.283931    2.134639
## Sepal.Width     -4.911715   14.629276     3.871407   -7.476280
## Petal.Length   -13.283931    3.871407    15.829162   -3.040134
## Petal.Width      2.134639   -7.476280    -3.040134   19.850390
```

```
newDist = predict(distData, iris[-trainSet,1:4])
#test data
head(newDist)
```

```
##    dist.setosa dist.versicolor dist.virginica
## 1   -0.5199012        4.571742       5.203359
## 2    1.0071680        4.208166       5.083959
## 15   2.4599421        5.065173       5.558309
## 18  -0.1194796        4.493137       5.169857
## 19   1.6850069        4.738999       5.274501
## 20   0.2904837        4.662716       5.175954
```

```
splom(newDist, groups = iris$Species[-trainSet], auto.key=list(columns=3))
```



....(scatter plot matrix)