|  |
| --- |
| > setwd('C:/Rdirectory/data\_mining/data\_mining\_covid');  > covid\_train = read.csv("covid\_train.csv", header=T);  > covid\_train$sex = as.factor(covid\_train$sex);  > covid\_train$patient\_type = as.factor(covid\_train$patient\_type);  > covid\_train$pneumonia = as.factor(covid\_train$pneumonia);  > covid\_train$age = as.numeric(covid\_train$age);  > covid\_train$diabetes = as.factor(covid\_train$diabetes);  > covid\_train$copd = as.factor(covid\_train$copd);  > covid\_train$asthma = as.factor(covid\_train$asthma);  > covid\_train$inmsupr = as.factor(covid\_train$inmsupr);  > covid\_train$hypertension = as.factor(covid\_train$hypertension);  > covid\_train$other\_disease = as.factor(covid\_train$other\_disease);  > covid\_train$cardiovascular = as.factor(covid\_train$cardiovascular);  > covid\_train$obesity = as.factor(covid\_train$obesity);  > covid\_train$renal\_chronic = as.factor(covid\_train$renal\_chronic);  > covid\_train$tobacco = as.factor(covid\_train$tobacco);  > covid\_train$contact\_other\_covid = as.factor(covid\_train$contact\_other\_covid);  > covid\_train$is\_dead = as.factor(covid\_train$is\_dead);  > str(covid\_train);  'data.frame': 32356 obs. of 16 variables:  $ sex : Factor w/ 1 level "1": 1 1 1 1 1 1 1 1 1 1 ...  $ patient\_type : Factor w/ 1 level "2": 1 1 1 1 1 1 1 1 1 1 ...  $ pneumonia : Factor w/ 2 levels "1","2": 2 2 1 1 1 2 1 1 1 2 ...  $ age : num 25 52 51 67 59 52 54 78 80 40 ...  $ diabetes : Factor w/ 2 levels "1","2": 2 2 2 1 1 1 2 2 1 2 ...  $ copd : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ asthma : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ inmsupr : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ hypertension : Factor w/ 2 levels "1","2": 2 2 2 1 2 1 2 1 1 2 ...  $ other\_disease : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ cardiovascular : Factor w/ 2 levels "1","2": 2 2 2 2 2 1 2 2 2 2 ...  $ obesity : Factor w/ 2 levels "1","2": 2 1 2 1 2 2 2 1 1 2 ...  $ renal\_chronic : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ tobacco : Factor w/ 2 levels "1","2": 2 1 2 2 2 2 2 2 2 2 ...  $ contact\_other\_covid: Factor w/ 2 levels "1","2": 1 1 1 1 1 2 2 1 1 2 ...  $ is\_dead : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 1 2 2 ...  > head(covid\_train);  sex patient\_type pneumonia age diabetes copd asthma inmsupr hypertension other\_disease cardiovascular obesity renal\_chronic  1 1 2 2 25 2 2 2 2 2 2 2 2 2  2 1 2 2 52 2 2 2 2 2 2 2 1 2  3 1 2 1 51 2 2 2 2 2 2 2 2 2  4 1 2 1 67 1 2 2 2 1 2 2 1 2  5 1 2 1 59 1 2 2 2 2 2 2 2 2  6 1 2 2 52 1 2 2 2 1 2 1 2 2  tobacco contact\_other\_covid is\_dead  1 2 1 2  2 1 1 2  3 2 1 2  4 2 1 2  5 2 1 2  6 2 2 2  > covid\_test = read.csv("covid\_test.csv", header=T);  > covid\_test$sex = as.factor(covid\_test$sex);  > covid\_test$patient\_type = as.factor(covid\_test$patient\_type);  > covid\_test$pneumonia = as.factor(covid\_test$pneumonia);  > covid\_test$age = as.numeric(covid\_test$age);  > covid\_test$diabetes = as.factor(covid\_test$diabetes);  > covid\_test$copd = as.factor(covid\_test$copd);  > covid\_test$asthma = as.factor(covid\_test$asthma);  > covid\_test$inmsupr = as.factor(covid\_test$inmsupr);  > covid\_test$hypertension = as.factor(covid\_test$hypertension);  > covid\_test$other\_disease = as.factor(covid\_test$other\_disease);  > covid\_test$cardiovascular = as.factor(covid\_test$cardiovascular);  > covid\_test$obesity = as.factor(covid\_test$obesity);  > covid\_test$renal\_chronic = as.factor(covid\_test$renal\_chronic);  > covid\_test$tobacco = as.factor(covid\_test$tobacco);  > covid\_test$contact\_other\_covid = as.factor(covid\_test$contact\_other\_covid);  > covid\_test$is\_dead = as.factor(covid\_test$is\_dead);  > str(covid\_test);  'data.frame': 5920 obs. of 16 variables:  $ sex : Factor w/ 1 level "1": 1 1 1 1 1 1 1 1 1 1 ...  $ patient\_type : Factor w/ 1 level "2": 1 1 1 1 1 1 1 1 1 1 ...  $ pneumonia : Factor w/ 2 levels "1","2": 2 2 1 1 1 1 2 2 2 1 ...  $ age : num 52 36 0 85 75 19 46 60 62 39 ...  $ diabetes : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 1 2 ...  $ copd : Factor w/ 2 levels "1","2": 2 2 2 1 2 2 2 2 2 2 ...  $ asthma : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ inmsupr : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ hypertension : Factor w/ 2 levels "1","2": 2 2 2 1 1 2 2 2 2 1 ...  $ other\_disease : Factor w/ 2 levels "1","2": 1 2 2 2 2 2 2 2 2 2 ...  $ cardiovascular : Factor w/ 2 levels "1","2": 2 2 2 1 2 2 2 2 2 2 ...  $ obesity : Factor w/ 2 levels "1","2": 2 2 2 1 2 2 2 2 2 2 ...  $ renal\_chronic : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ tobacco : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ contact\_other\_covid: Factor w/ 2 levels "1","2": 2 2 2 2 2 2 1 2 2 2 ...  $ is\_dead : Factor w/ 2 levels "1","2": 2 2 1 2 2 2 2 2 2 2 ...  > head(covid\_test);  sex patient\_type pneumonia age diabetes copd asthma inmsupr hypertension other\_disease cardiovascular obesity renal\_chronic  1 1 2 2 52 2 2 2 2 2 1 2 2 2  2 1 2 2 36 2 2 2 2 2 2 2 2 2  3 1 2 1 0 2 2 2 2 2 2 2 2 2  4 1 2 1 85 2 1 2 2 1 2 1 1 2  5 1 2 1 75 2 2 2 2 1 2 2 2 2  6 1 2 1 19 2 2 2 2 2 2 2 2 2  tobacco contact\_other\_covid is\_dead  1 2 2 2  2 2 2 2  3 2 2 1  4 2 2 2  5 2 2 2  6 2 2 2  > covid\_dead\_train = read.csv("covid\_dead\_train.csv", header=T);  > covid\_dead\_train$sex = as.factor(covid\_dead\_train$sex);  > covid\_dead\_train$patient\_type = as.factor(covid\_dead\_train$patient\_type);  > covid\_dead\_train$pneumonia = as.factor(covid\_dead\_train$pneumonia);  > covid\_dead\_train$age = as.numeric(covid\_dead\_train$age);  > covid\_dead\_train$diabetes = as.factor(covid\_dead\_train$diabetes);  > covid\_dead\_train$copd = as.factor(covid\_dead\_train$copd);  > covid\_dead\_train$asthma = as.factor(covid\_dead\_train$asthma);  > covid\_dead\_train$inmsupr = as.factor(covid\_dead\_train$inmsupr);  > covid\_dead\_train$hypertension = as.factor(covid\_dead\_train$hypertension);  > covid\_dead\_train$other\_disease = as.factor(covid\_dead\_train$other\_disease);  > covid\_dead\_train$cardiovascular = as.factor(covid\_dead\_train$cardiovascular);  > covid\_dead\_train$obesity = as.factor(covid\_dead\_train$obesity);  > covid\_dead\_train$renal\_chronic = as.factor(covid\_dead\_train$renal\_chronic);  > covid\_dead\_train$tobacco = as.factor(covid\_dead\_train$tobacco);  > covid\_dead\_train$contact\_other\_covid = as.factor(covid\_dead\_train$contact\_other\_covid);  > covid\_dead\_train$day\_cnt = as.numeric(covid\_dead\_train$day\_cnt);  > str(covid\_dead\_train);  'data.frame': 3600 obs. of 16 variables:  $ sex : Factor w/ 1 level "1": 1 1 1 1 1 1 1 1 1 1 ...  $ patient\_type : Factor w/ 1 level "2": 1 1 1 1 1 1 1 1 1 1 ...  $ pneumonia : Factor w/ 2 levels "1","2": 1 1 2 1 1 1 1 1 1 2 ...  $ age : num 78 65 58 78 92 70 71 65 56 72 ...  $ diabetes : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 1 1 2 2 ...  $ copd : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 1 2 2 2 ...  $ asthma : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ inmsupr : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ hypertension : Factor w/ 2 levels "1","2": 1 2 2 2 1 2 1 1 2 1 ...  $ other\_disease : Factor w/ 2 levels "1","2": 2 2 1 2 2 2 2 2 2 2 ...  $ cardiovascular : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ obesity : Factor w/ 2 levels "1","2": 1 1 2 1 2 1 1 1 2 2 ...  $ renal\_chronic : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 1 ...  $ tobacco : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 1 2 2 2 ...  $ contact\_other\_covid: Factor w/ 2 levels "1","2": 1 2 1 1 2 2 1 2 2 2 ...  $ day\_cnt : num 10 7 1 6 3 16 21 14 30 20 ...  > head(covid\_dead\_train);  sex patient\_type pneumonia age diabetes copd asthma inmsupr hypertension other\_disease cardiovascular obesity renal\_chronic  1 1 2 1 78 2 2 2 2 1 2 2 1 2  2 1 2 1 65 2 2 2 2 2 2 2 1 2  3 1 2 2 58 2 2 2 2 2 1 2 2 2  4 1 2 1 78 2 2 2 2 2 2 2 1 2  5 1 2 1 92 2 2 2 2 1 2 2 2 2  6 1 2 1 70 2 2 2 2 2 2 2 1 2  tobacco contact\_other\_covid day\_cnt  1 2 1 10  2 2 2 7  3 2 1 1  4 2 1 6  5 2 2 3  6 2 2 16  > covid\_dead\_test = read.csv("covid\_dead\_test.csv", header=T);  > covid\_dead\_test$sex = as.factor(covid\_dead\_test$sex);  > covid\_dead\_test$patient\_type = as.factor(covid\_dead\_test$patient\_type);  > covid\_dead\_test$pneumonia = as.factor(covid\_dead\_test$pneumonia);  > covid\_dead\_test$age = as.numeric(covid\_dead\_test$age);  > covid\_dead\_test$diabetes = as.factor(covid\_dead\_test$diabetes);  > covid\_dead\_test$copd = as.factor(covid\_dead\_test$copd);  > covid\_dead\_test$asthma = as.factor(covid\_dead\_test$asthma);  > covid\_dead\_test$inmsupr = as.factor(covid\_dead\_test$inmsupr);  > covid\_dead\_test$hypertension = as.factor(covid\_dead\_test$hypertension);  > covid\_dead\_test$other\_disease = as.factor(covid\_dead\_test$other\_disease);  > covid\_dead\_test$cardiovascular = as.factor(covid\_dead\_test$cardiovascular);  > covid\_dead\_test$obesity = as.factor(covid\_dead\_test$obesity);  > covid\_dead\_test$renal\_chronic = as.factor(covid\_dead\_test$renal\_chronic);  > covid\_dead\_test$tobacco = as.factor(covid\_dead\_test$tobacco);  > covid\_dead\_test$contact\_other\_covid = as.factor(covid\_dead\_test$contact\_other\_covid);  > covid\_dead\_test$day\_cnt = as.numeric(covid\_dead\_test$day\_cnt);  > str(covid\_dead\_test);  'data.frame': 420 obs. of 16 variables:  $ sex : Factor w/ 1 level "1": 1 1 1 1 1 1 1 1 1 1 ...  $ patient\_type : Factor w/ 1 level "2": 1 1 1 1 1 1 1 1 1 1 ...  $ pneumonia : Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...  $ age : num 66 68 49 61 47 56 68 72 78 64 ...  $ diabetes : Factor w/ 2 levels "1","2": 1 2 2 1 1 2 1 1 1 1 ...  $ copd : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 1 2 2 ...  $ asthma : Factor w/ 2 levels "1","2": 2 2 2 2 1 2 2 2 2 2 ...  $ inmsupr : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ hypertension : Factor w/ 2 levels "1","2": 1 2 2 1 2 2 1 1 2 2 ...  $ other\_disease : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ cardiovascular : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ obesity : Factor w/ 2 levels "1","2": 1 2 1 2 2 2 1 1 1 2 ...  $ renal\_chronic : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ tobacco : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2 ...  $ contact\_other\_covid: Factor w/ 2 levels "1","2": 1 2 2 2 2 2 2 2 2 2 ...  $ day\_cnt : num 3 5 16 4 18 10 9 6 6 8 ...  > head(covid\_dead\_test);  sex patient\_type pneumonia age diabetes copd asthma inmsupr hypertension other\_disease cardiovascular obesity renal\_chronic  1 1 2 1 66 1 2 2 2 1 2 2 1 2  2 1 2 1 68 2 2 2 2 2 2 2 2 2  3 1 2 1 49 2 2 2 2 2 2 2 1 2  4 1 2 1 61 1 2 2 2 1 2 2 2 2  5 1 2 1 47 1 2 1 2 2 2 2 2 2  6 1 2 1 56 2 2 2 2 2 2 2 2 2  tobacco contact\_other\_covid day\_cnt  1 2 1 3  2 2 2 5  3 2 2 16  4 2 2 4  5 2 2 18  6 2 2 10  > # install.packages("gbm");  > require(gbm);  필요한 패키지를 로딩중입니다: gbm  Error: package or namespace load failed for ‘gbm’:  package ‘gbm’ was installed before R 4.0.0: please re-install it  > install.packages("gbm");  WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:  https://cran.rstudio.com/bin/windows/Rtools/  ‘C:/Users/JNP/Documents/R/win-library/4.1’의 위치에 패키지(들)을 설치합니다.  (왜냐하면 ‘lib’가 지정되지 않았기 때문입니다)  trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/gbm\_2.1.8.1.zip'  Content type 'application/zip' length 888085 bytes (867 KB)  downloaded 867 KB  패키지 ‘gbm’를 성공적으로 압축해제하였고 MD5 sums 이 확인되었습니다  다운로드된 바이너리 패키지들은 다음의 위치에 있습니다  C:\Users\JNP\AppData\Local\Temp\RtmpEXqUUA\downloaded\_packages  > #install.packages("gbm");  > require(gbm);  필요한 패키지를 로딩중입니다: gbm  Loaded gbm 2.1.8.1  > covid\_train = covid\_train[,!names(covid\_train) %in% c("sex", "patient\_type")];  > covid\_test = covid\_test[,!names(covid\_test) %in% c("sex", "patient\_type")];  > gbmFit = gbm(is\_dead~., data=covid\_train,distribution = "multinomial", n.trees = 1000, shrinkage = 0.01, interaction.depth = 4);  Warning message:  Setting `distribution = "multinomial"` is ill-advised as it is currently broken. It exists only for backwards compatibility. Use at your own risk.  > plot(gbmFit)  스크린샷, 텍스트, 도표, 라인이(가) 표시된 사진  자동 생성된 설명  > print(gbmFit$fit);  1 2  [1,] 3.926130e-01 -3.926130e-01  [2,] -1.457685e+00 1.457685e+00  [3,] -9.240206e-01 9.240206e-01  [4,] -4.031684e-01 4.031684e-01  [5,] 2.101038e-01 -2.101038e-01  [6,] -5.687782e-02 5.687782e-02  [7,] -4.621170e-01 4.621170e-01  [8,] -4.404805e-02 4.404805e-02  [9,] 3.713149e-01 -3.713149e-01  [10,] -1.014084e+00 1.014084e+00  [11,] 1.282463e-01 -1.282463e-01  [12,] 1.437771e-01 -1.437771e-01  [13,] 1.328032e-01 -1.328032e-01  [14,] -9.056060e-01 9.056060e-01  [15,] -3.711374e-01 3.711374e-01  [16,] 3.233290e-01 -3.233290e-01  [17,] 4.256083e-01 -4.256083e-01  [18,] -1.086464e+00 1.086464e+00  [19,] -1.220228e+00 1.220228e+00  [20,] 2.114187e-01 -2.114187e-01  [21,] 2.002544e-01 -2.002544e-01  [22,] 1.767010e-01 -1.767010e-01  [23,] 2.225970e-02 -2.225970e-02  [24,] 7.412953e-02 -7.412953e-02  [25,] 5.070211e-02 -5.070211e-02  [26,] -5.437430e-01 5.437430e-01  [27,] -6.588110e-01 6.588110e-01  [28,] 8.562625e-02 -8.562625e-02  [29,] -1.158477e+00 1.158477e+00  [30,] 4.041064e-01 -4.041064e-01  [31,] -3.565016e-01 3.565016e-01  [32,] 8.776178e-02 -8.776178e-02  [33,] -5.805901e-01 5.805901e-01  [34,] -1.211786e+00 1.211786e+00  [35,] -4.689288e-02 4.689288e-02  [36,] 6.346903e-02 -6.346903e-02  [37,] 1.854542e-01 -1.854542e-01  [38,] -7.410726e-01 7.410726e-01  [39,] 5.919893e-02 -5.919893e-02  [40,] -1.485848e+00 1.485848e+00  [41,] -1.327511e+00 1.327511e+00  [42,] -7.320375e-01 7.320375e-01  [43,] -7.679306e-01 7.679306e-01  [44,] -6.012828e-01 6.012828e-01  [45,] 1.373675e-01 -1.373675e-01  [46,] -3.245501e-01 3.245501e-01  [47,] -6.225142e-01 6.225142e-01  [48,] -2.194662e-01 2.194662e-01  [49,] -2.019050e-01 2.019050e-01  [50,] -3.358055e-01 3.358055e-01  [51,] -1.457887e+00 1.457887e+00  [52,] -5.969775e-03 5.969775e-03  [53,] 4.269561e-01 -4.269561e-01  [54,] 1.877657e-01 -1.877657e-01  [55,] -3.826232e-02 3.826232e-02  [56,] -5.472846e-01 5.472846e-01  [57,] -1.260860e+00 1.260860e+00  [58,] -9.969836e-01 9.969836e-01  [59,] -2.618602e-01 2.618602e-01  [60,] -3.940575e-01 3.940575e-01  [61,] -4.211553e-01 4.211553e-01  [62,] 7.091700e-02 -7.091700e-02  [63,] 3.153318e-01 -3.153318e-01  [64,] 2.090126e-01 -2.090126e-01  [65,] 4.019599e-01 -4.019599e-01  [66,] 1.894829e-01 -1.894829e-01  [67,] 1.376706e-01 -1.376706e-01  [68,] 1.418822e-02 -1.418822e-02  [69,] 2.879574e-01 -2.879574e-01  [70,] -2.222085e-01 2.222085e-01  [71,] -4.493031e-01 4.493031e-01  [72,] 3.115997e-01 -3.115997e-01  [73,] -7.849718e-02 7.849718e-02  [74,] -7.758892e-01 7.758892e-01  [75,] -2.701979e-01 2.701979e-01  [76,] -1.524201e-01 1.524201e-01  [77,] -3.476280e-01 3.476280e-01  [78,] -8.944246e-01 8.944246e-01  [79,] 2.181049e-01 -2.181049e-01  [80,] 3.673171e-01 -3.673171e-01  [81,] 6.759909e-02 -6.759909e-02  [82,] -8.531832e-01 8.531832e-01  [83,] 1.574034e-01 -1.574034e-01  [84,] -1.512125e-01 1.512125e-01  [85,] -4.832418e-01 4.832418e-01  [86,] -5.384473e-01 5.384473e-01  [87,] 2.701861e-01 -2.701861e-01  [88,] -7.832571e-02 7.832571e-02  [89,] -2.498222e-01 2.498222e-01  [90,] 2.837022e-01 -2.837022e-01  [91,] 3.356149e-01 -3.356149e-01  [92,] -7.539363e-02 7.539363e-02  [93,] -4.526907e-01 4.526907e-01  [94,] 2.090126e-01 -2.090126e-01  [95,] -3.997912e-01 3.997912e-01  [96,] 8.012430e-02 -8.012430e-02  [97,] -4.345287e-01 4.345287e-01  [98,] -6.512482e-01 6.512482e-01  [99,] -2.327432e-01 2.327432e-01  [100,] 4.383629e-01 -4.383629e-01  [101,] -3.308044e-01 3.308044e-01  [102,] -4.977609e-02 4.977609e-02  [103,] -9.435646e-03 9.435646e-03  [104,] -1.457685e+00 1.457685e+00  [105,] -9.969836e-01 9.969836e-01  [106,] -8.159164e-01 8.159164e-01  [107,] -8.591339e-01 8.591339e-01  [108,] 8.620173e-02 -8.620173e-02  [109,] -4.125114e-01 4.125114e-01  [110,] -6.204048e-01 6.204048e-01  [111,] -1.227581e+00 1.227581e+00  [112,] 4.183558e-01 -4.183558e-01  [113,] 3.229231e-01 -3.229231e-01  [114,] 7.145724e-02 -7.145724e-02  [115,] -4.345287e-01 4.345287e-01  [116,] 2.752777e-01 -2.752777e-01  [117,] -3.565016e-01 3.565016e-01  [118,] -3.603579e-01 3.603579e-01  [119,] -9.144725e-01 9.144725e-01  [120,] 2.336226e-01 -2.336226e-01  [121,] -2.019050e-01 2.019050e-01  [122,] -6.523047e-01 6.523047e-01  [123,] -7.679306e-01 7.679306e-01  [124,] 3.409781e-01 -3.409781e-01  [125,] -2.451642e-01 2.451642e-01  [126,] -1.086304e+00 1.086304e+00  [127,] -6.512482e-01 6.512482e-01  [128,] -3.798729e-01 3.798729e-01  [129,] 2.731588e-01 -2.731588e-01  [130,] -2.838730e-03 2.838730e-03  [131,] -6.248080e-02 6.248080e-02  [132,] -3.862138e-01 3.862138e-01  [133,] -4.064677e-01 4.064677e-01  [134,] -4.618389e-01 4.618389e-01  [135,] -7.838453e-01 7.838453e-01  [136,] 2.409571e-01 -2.409571e-01  [137,] 1.917171e-01 -1.917171e-01  [138,] 1.156092e-01 -1.156092e-01  [139,] -1.433845e+00 1.433845e+00  [140,] -2.085558e-01 2.085558e-01  [141,] 1.514089e-01 -1.514089e-01  [142,] -3.649675e-01 3.649675e-01  [143,] 2.916603e-01 -2.916603e-01  [144,] 6.259305e-02 -6.259305e-02  [145,] 4.832237e-02 -4.832237e-02  [146,] -5.925473e-01 5.925473e-01  [147,] -4.657187e-01 4.657187e-01  [148,] 1.303206e-02 -1.303206e-02  [149,] -4.696005e-01 4.696005e-01  [150,] 4.269561e-01 -4.269561e-01  [151,] -1.413486e+00 1.413486e+00  [152,] -4.967330e-01 4.967330e-01  [153,] 2.784118e-01 -2.784118e-01  [154,] 1.797100e-01 -1.797100e-01  [155,] -1.910247e-01 1.910247e-01  [156,] -7.849718e-02 7.849718e-02  [157,] -5.202065e-02 5.202065e-02  [158,] -5.805901e-01 5.805901e-01  [159,] -1.226119e+00 1.226119e+00  [160,] 4.793010e-01 -4.793010e-01  [161,] -7.849718e-02 7.849718e-02  [162,] -4.967377e-01 4.967377e-01  [163,] -2.587045e-01 2.587045e-01  [164,] 9.150264e-02 -9.150264e-02  [165,] -3.826232e-02 3.826232e-02  [166,] -1.113845e+00 1.113845e+00  [167,] -5.884465e-01 5.884465e-01  [168,] -2.433528e-01 2.433528e-01  [169,] -6.560740e-01 6.560740e-01  [170,] 8.493618e-02 -8.493618e-02  [171,] -1.629831e-01 1.629831e-01  [172,] -1.440407e+00 1.440407e+00  [173,] -2.019050e-01 2.019050e-01  [174,] -3.579141e-01 3.579141e-01  [175,] -3.950491e-02 3.950491e-02  [176,] -1.159287e+00 1.159287e+00  [177,] 1.435029e-01 -1.435029e-01  [178,] 1.429627e-01 -1.429627e-01  [179,] -9.331947e-01 9.331947e-01  [180,] -3.743998e-02 3.743998e-02  [181,] -1.374382e+00 1.374382e+00  [182,] -1.111776e+00 1.111776e+00  [183,] -4.500362e-01 4.500362e-01  [184,] -1.314572e+00 1.314572e+00  [185,] -6.720919e-01 6.720919e-01  [186,] 3.422236e-01 -3.422236e-01  [187,] -4.937398e-01 4.937398e-01  [188,] 1.466543e-01 -1.466543e-01  [189,] 1.315509e-01 -1.315509e-01  [190,] 1.999132e-01 -1.999132e-01  [191,] -1.413486e+00 1.413486e+00  [192,] -1.406677e-01 1.406677e-01  [193,] -4.979963e-01 4.979963e-01  [194,] -8.111197e-01 8.111197e-01  [195,] 1.367103e-01 -1.367103e-01  [196,] -2.625736e-02 2.625736e-02  [197,] 1.978218e-01 -1.978218e-01  [198,] -4.455729e-02 4.455729e-02  [199,] 1.764193e-01 -1.764193e-01  [200,] 2.181831e-01 -2.181831e-01  [201,] -4.031684e-01 4.031684e-01  [202,] 7.040799e-02 -7.040799e-02  [203,] 3.920985e-01 -3.920985e-01  [204,] 1.742819e-01 -1.742819e-01  [205,] -1.113845e+00 1.113845e+00  [206,] 2.079548e-01 -2.079548e-01  [207,] 2.615032e-01 -2.615032e-01  [208,] -1.132704e+00 1.132704e+00  [209,] -4.025374e-01 4.025374e-01  [210,] -6.612318e-01 6.612318e-01  [211,] -4.455715e-01 4.455715e-01  [212,] 3.294234e-01 -3.294234e-01  [213,] -4.332426e-01 4.332426e-01  [214,] -9.615975e-01 9.615975e-01  [215,] -5.925473e-01 5.925473e-01  [216,] -4.750762e-01 4.750762e-01  [217,] -8.768073e-01 8.768073e-01  [218,] -1.149205e+00 1.149205e+00  [219,] 1.923256e-01 -1.923256e-01  [220,] -3.826232e-02 3.826232e-02  [221,] 1.018391e-01 -1.018391e-01  [222,] -4.925926e-01 4.925926e-01  [223,] -1.392923e-01 1.392923e-01  [224,] -6.366547e-01 6.366547e-01  [225,] -1.374382e+00 1.374382e+00  [226,] -1.186091e-01 1.186091e-01  [227,] 1.563302e-01 -1.563302e-01  [228,] -2.362000e-01 2.362000e-01  [229,] 1.942739e-01 -1.942739e-01  [230,] -1.422611e+00 1.422611e+00  [231,] 4.843839e-01 -4.843839e-01  [232,] -1.109992e+00 1.109992e+00  [233,] 8.397283e-03 -8.397283e-03  [234,] -4.750762e-01 4.750762e-01  [235,] -4.523271e-01 4.523271e-01  [236,] 1.563302e-01 -1.563302e-01  [237,] 2.615032e-01 -2.615032e-01  [238,] 1.242050e-01 -1.242050e-01  [239,] 3.648823e-01 -3.648823e-01  [240,] 9.544306e-02 -9.544306e-02  [241,] -1.472083e-01 1.472083e-01  [242,] -4.618389e-01 4.618389e-01  [243,] -3.459042e-02 3.459042e-02  [244,] -6.424755e-01 6.424755e-01  [245,] 1.268739e-02 -1.268739e-02  [246,] 4.038133e-01 -4.038133e-01  [247,] -7.320375e-01 7.320375e-01  [248,] 1.233727e-01 -1.233727e-01  [249,] -5.282156e-02 5.282156e-02  [250,] -1.277030e-01 1.277030e-01  [251,] 4.269561e-01 -4.269561e-01  [252,] -1.911792e-01 1.911792e-01  [253,] -3.997912e-01 3.997912e-01  [254,] 2.356649e-01 -2.356649e-01  [255,] -4.500362e-01 4.500362e-01  [256,] 2.090126e-01 -2.090126e-01  [257,] 2.015359e-01 -2.015359e-01  [258,] -5.521990e-01 5.521990e-01  [259,] -7.255840e-01 7.255840e-01  [260,] -2.019050e-01 2.019050e-01  [261,] -1.844109e-01 1.844109e-01  [262,] 1.324174e-01 -1.324174e-01  [263,] -8.350945e-01 8.350945e-01  [264,] 1.762148e-01 -1.762148e-01  [265,] -6.097299e-01 6.097299e-01  [266,] 2.035557e-01 -2.035557e-01  [267,] -2.062964e-01 2.062964e-01  [268,] 2.701049e-01 -2.701049e-01  [269,] 9.119817e-02 -9.119817e-02  [270,] -8.293354e-02 8.293354e-02  [271,] 4.969037e-02 -4.969037e-02  [272,] -1.288708e+00 1.288708e+00  [273,] 5.620067e-02 -5.620067e-02  [274,] 2.895238e-01 -2.895238e-01  [275,] -8.547909e-02 8.547909e-02  [276,] 1.298595e-02 -1.298595e-02  [277,] 1.694560e-03 -1.694560e-03  [278,] 2.167403e-01 -2.167403e-01  [279,] -1.457887e+00 1.457887e+00  [280,] 3.493686e-01 -3.493686e-01  [281,] 1.504871e-01 -1.504871e-01  [282,] 1.033031e-01 -1.033031e-01  [283,] -3.826232e-02 3.826232e-02  [284,] 2.079548e-01 -2.079548e-01  [285,] 7.569971e-02 -7.569971e-02  [286,] 3.196410e-01 -3.196410e-01  [287,] -1.117930e+00 1.117930e+00  [288,] -6.097299e-01 6.097299e-01  [289,] 1.373675e-01 -1.373675e-01  [290,] -1.108920e+00 1.108920e+00  [291,] 1.851650e-01 -1.851650e-01  [292,] 2.236987e-01 -2.236987e-01  [293,] 1.304959e-01 -1.304959e-01  [294,] 1.897555e-01 -1.897555e-01  [295,] -7.518120e-01 7.518120e-01  [296,] -2.159791e-01 2.159791e-01  [297,] 1.842838e-01 -1.842838e-01  [298,] 1.153421e-01 -1.153421e-01  [299,] 1.414149e-02 -1.414149e-02  [300,] 2.775690e-01 -2.775690e-01  [301,] -8.483649e-01 8.483649e-01  [302,] 2.583303e-01 -2.583303e-01  [303,] 1.563302e-01 -1.563302e-01  [304,] 1.169071e-01 -1.169071e-01  [305,] -3.139522e-01 3.139522e-01  [306,] -1.159287e+00 1.159287e+00  [307,] -2.628698e-01 2.628698e-01  [308,] -5.469705e-01 5.469705e-01  [309,] -1.143693e+00 1.143693e+00  [310,] 7.091700e-02 -7.091700e-02  [311,] 5.331282e-02 -5.331282e-02  [312,] -2.062964e-01 2.062964e-01  [313,] -9.896062e-01 9.896062e-01  [314,] -1.885120e-01 1.885120e-01  [315,] -1.142691e+00 1.142691e+00  [316,] 2.904647e-01 -2.904647e-01  [317,] -1.028398e+00 1.028398e+00  [318,] -1.166029e-01 1.166029e-01  [319,] 7.036105e-02 -7.036105e-02  [320,] 7.748759e-02 -7.748759e-02  [321,] -5.450616e-02 5.450616e-02  [322,] -1.472290e+00 1.472290e+00  [323,] -3.137039e-01 3.137039e-01  [324,] -1.243380e+00 1.243380e+00  [325,] -6.886694e-01 6.886694e-01  [326,] 4.969037e-02 -4.969037e-02  [327,] -1.059007e-01 1.059007e-01  [328,] -7.838453e-01 7.838453e-01  [329,] -1.310411e+00 1.310411e+00  [330,] -1.079295e-01 1.079295e-01  [331,] -4.363834e-01 4.363834e-01  [332,] -4.432306e-01 4.432306e-01  [333,] -1.142691e+00 1.142691e+00  [334,] 1.379707e-01 -1.379707e-01  [335,] 2.057567e-01 -2.057567e-01  [336,] -7.492387e-01 7.492387e-01  [337,] -8.111197e-01 8.111197e-01  [338,] 6.618908e-03 -6.618908e-03  [339,] -3.798729e-01 3.798729e-01  [340,] 3.435179e-01 -3.435179e-01  [341,] 1.718557e-01 -1.718557e-01  [342,] 1.699971e-01 -1.699971e-01  [343,] 1.563302e-01 -1.563302e-01  [344,] 1.155378e-01 -1.155378e-01  [345,] -2.951855e-01 2.951855e-01  [346,] -3.139522e-01 3.139522e-01  [347,] -1.118870e-01 1.118870e-01  [348,] 1.842838e-01 -1.842838e-01  [349,] -2.493524e-02 2.493524e-02  [350,] -2.799060e-01 2.799060e-01  [351,] 3.167465e-01 -3.167465e-01  [352,] -8.947865e-02 8.947865e-02  [353,] 6.419401e-02 -6.419401e-02  [354,] -3.010891e-01 3.010891e-01  [355,] -6.245951e-02 6.245951e-02  [356,] 4.183558e-01 -4.183558e-01  [357,] -6.660936e-01 6.660936e-01  [358,] -5.771065e-01 5.771065e-01  [359,] 1.425709e-01 -1.425709e-01  [360,] -8.176230e-01 8.176230e-01  [361,] 2.261163e-01 -2.261163e-01  [362,] -6.886694e-01 6.886694e-01  [363,] -1.220228e+00 1.220228e+00  [364,] -2.620838e-01 2.620838e-01  [365,] -5.426124e-01 5.426124e-01  [366,] -1.415615e+00 1.415615e+00  [367,] -1.239897e+00 1.239897e+00  [368,] -4.867236e-01 4.867236e-01  [369,] 1.466543e-01 -1.466543e-01  [370,] 2.821978e-01 -2.821978e-01  [371,] 1.642746e-02 -1.642746e-02  [372,] -8.399693e-02 8.399693e-02  [373,] -1.393468e-01 1.393468e-01  [374,] -9.389628e-01 9.389628e-01  [375,] 1.909019e-01 -1.909019e-01  [376,] -2.393134e-01 2.393134e-01  [377,] -4.681547e-01 4.681547e-01  [378,] -6.770822e-01 6.770822e-01  [379,] -9.216586e-01 9.216586e-01  [380,] 8.848965e-02 -8.848965e-02  [381,] -1.458408e+00 1.458408e+00  [382,] -1.777178e-01 1.777178e-01  [383,] 8.238516e-02 -8.238516e-02  [384,] 7.854321e-02 -7.854321e-02  [385,] -9.946121e-02 9.946121e-02  [386,] 2.073934e-01 -2.073934e-01  [387,] -1.421986e+00 1.421986e+00  [388,] 1.797100e-01 -1.797100e-01  [389,] 1.800108e-01 -1.800108e-01  [390,] -9.226253e-01 9.226253e-01  [391,] -6.838696e-01 6.838696e-01  [392,] -3.498088e-02 3.498088e-02  [393,] 5.630138e-02 -5.630138e-02  [394,] 1.630350e-01 -1.630350e-01  [395,] -3.598612e-01 3.598612e-01  [396,] -3.859997e-01 3.859997e-01  [397,] 2.406957e-01 -2.406957e-01  [398,] 1.490241e-01 -1.490241e-01  [399,] -4.011043e-01 4.011043e-01  [400,] -9.896062e-01 9.896062e-01  [401,] 1.719835e-01 -1.719835e-01  [402,] -1.175756e+00 1.175756e+00  [403,] -8.380685e-01 8.380685e-01  [404,] -1.238415e+00 1.238415e+00  [405,] -1.718120e+00 1.718120e+00  [406,] 1.470824e-02 -1.470824e-02  [407,] 1.382115e-01 -1.382115e-01  [408,] -5.241246e-01 5.241246e-01  [409,] 2.775977e-01 -2.775977e-01  [410,] 3.866801e-01 -3.866801e-01  [411,] -1.117765e+00 1.117765e+00  [412,] -4.725752e-01 4.725752e-01  [413,] 3.409781e-01 -3.409781e-01  [414,] 8.842120e-02 -8.842120e-02  [415,] -1.212412e+00 1.212412e+00  [416,] -3.957330e-01 3.957330e-01  [417,] -1.632824e-01 1.632824e-01  [418,] -6.988858e-01 6.988858e-01  [419,] -8.961687e-01 8.961687e-01  [420,] -1.298593e-01 1.298593e-01  [421,] -9.896062e-01 9.896062e-01  [422,] 1.540470e-01 -1.540470e-01  [423,] 1.382115e-01 -1.382115e-01  [424,] -9.615975e-01 9.615975e-01  [425,] -1.449313e+00 1.449313e+00  [426,] 7.487691e-02 -7.487691e-02  [427,] 8.706236e-02 -8.706236e-02  [428,] -3.108379e-01 3.108379e-01  [429,] -1.060476e+00 1.060476e+00  [430,] -9.204233e-02 9.204233e-02  [431,] 1.390407e-01 -1.390407e-01  [432,] -1.440407e+00 1.440407e+00  [433,] 1.923256e-01 -1.923256e-01  [434,] -3.658384e-01 3.658384e-01  [435,] -2.201047e-02 2.201047e-02  [436,] 1.699971e-01 -1.699971e-01  [437,] -1.575019e-01 1.575019e-01  [438,] -9.161486e-02 9.161486e-02  [439,] 1.231858e-01 -1.231858e-01  [440,] 1.005489e-01 -1.005489e-01  [441,] -1.446747e+00 1.446747e+00  [442,] 1.582474e-01 -1.582474e-01  [443,] -3.997912e-01 3.997912e-01  [444,] -1.606368e-01 1.606368e-01  [445,] -2.551273e-01 2.551273e-01  [446,] -1.718120e+00 1.718120e+00  [447,] 1.764193e-01 -1.764193e-01  [448,] -1.001144e+00 1.001144e+00  [449,] -8.228381e-02 8.228381e-02  [450,] -6.507564e-01 6.507564e-01  [451,] -4.345287e-01 4.345287e-01  [452,] -9.405612e-02 9.405612e-02  [453,] -5.222718e-01 5.222718e-01  [454,] -1.524201e-01 1.524201e-01  [455,] 1.205208e-01 -1.205208e-01  [456,] 1.080116e-01 -1.080116e-01  [457,] 1.198130e-01 -1.198130e-01  [458,] 2.667954e-01 -2.667954e-01  [459,] -9.405612e-02 9.405612e-02  [460,] -1.954525e-01 1.954525e-01  [461,] -1.800117e-03 1.800117e-03  [462,] -1.360412e-01 1.360412e-01  [463,] -5.029046e-02 5.029046e-02  [464,] 3.150107e-01 -3.150107e-01  [465,] -4.141992e-01 4.141992e-01  [466,] 9.780350e-02 -9.780350e-02  [467,] 4.969037e-02 -4.969037e-02  [468,] -1.598361e-01 1.598361e-01  [469,] 5.620067e-02 -5.620067e-02  [470,] -1.604927e-01 1.604927e-01  [471,] -5.615832e-01 5.615832e-01  [472,] 1.469267e-01 -1.469267e-01  [473,] -3.459042e-02 3.459042e-02  [474,] 3.460925e-01 -3.460925e-01  [475,] 1.080116e-01 -1.080116e-01  [476,] -1.153831e+00 1.153831e+00  [477,] 7.145724e-02 -7.145724e-02  [478,] -5.925473e-01 5.925473e-01  [479,] -1.773513e-01 1.773513e-01  [480,] -3.466748e-01 3.466748e-01  [481,] -1.844109e-01 1.844109e-01  [482,] -7.325908e-01 7.325908e-01  [483,] 1.198130e-01 -1.198130e-01  [484,] -3.459042e-02 3.459042e-02  [485,] 8.651736e-02 -8.651736e-02  [486,] -8.176230e-01 8.176230e-01  [487,] -1.197473e+00 1.197473e+00  [488,] -8.548047e-01 8.548047e-01  [489,] -1.145240e+00 1.145240e+00  [490,] -2.809718e-01 2.809718e-01  [491,] 8.620173e-02 -8.620173e-02  [492,] 1.923256e-01 -1.923256e-01  [493,] -2.906451e-01 2.906451e-01  [494,] 1.480420e-01 -1.480420e-01  [495,] -1.472083e-01 1.472083e-01  [496,] -1.325377e+00 1.325377e+00  [497,] -3.281204e-02 3.281204e-02  [498,] 2.842335e-01 -2.842335e-01  [499,] -1.132704e+00 1.132704e+00  [500,] -6.886694e-01 6.886694e-01  [ reached getOption("max.print") -- omitted 31856 rows ]  > summary(gbmFit)  var rel.inf  age age 68.36827198  pneumonia pneumonia 26.34785409  diabetes diabetes 1.80866654  contact\_other\_covid contact\_other\_covid 1.21728108  hypertension hypertension 0.70147827  obesity obesity 0.49533507  renal\_chronic renal\_chronic 0.47465054  other\_disease other\_disease 0.14639863  tobacco tobacco 0.12027083  cardiovascular cardiovascular 0.10905133  copd copd 0.07474158  asthma asthma 0.07424997  inmsupr inmsupr 0.06175009  > prediction = predict.gbm(gbmFit, newdata=covid\_test[], type="response");  Using 1000 trees...  텍스트, 스크린샷, 도표, 라인이(가) 표시된 사진  자동 생성된 설명  > summary(prediction);  Min. 1st Qu. Median Mean 3rd Qu. Max.  0.02966 0.28037 0.50000 0.50000 0.71963 0.97034  > value = apply(prediction,1,which.max);  > comparison=cbind(covid\_test,value);  > comparison=as.data.frame(comparison);  > print(paste("test 건수 : ",nrow(covid\_test)));  [1] "test 건수 : 5920"  > predictCorrect = comparison[comparison$is\_dead == comparison$value,];  > print(paste("사망여부 예측성공 건수 : ", nrow(predictCorrect)));  [1] "사망여부 예측성공 건수 : 4240"  > print(paste("사망여부 예측 정확도 : " ,nrow(predictCorrect)/nrow(covid\_test))); # 71.4%  [1] "사망여부 예측 정확도 : 0.716216216216216"  > covid\_dead\_train = covid\_dead\_train[,!names(covid\_dead\_train) %in% c("sex", "patient\_type")];  > covid\_dead\_test = covid\_dead\_test[,!names(covid\_dead\_test) %in% c("sex", "patient\_type")];  > gbmFit\_dead = gbm(day\_cnt~., data=covid\_dead\_train,distribution = "gaussian", n.trees = 1000, shrinkage = 0.01, interaction.depth = 4);  > plot(gbmFit\_dead)  텍스트, 스크린샷, 도표, 직사각형이(가) 표시된 사진  자동 생성된 설명  > print(gbmFit\_dead$fit);  [1] 10.200693 12.542804 11.608025 10.555908 7.929324 11.384838 10.505397 11.206875 12.202362 9.333658 12.338519 12.471259  [13] 12.499913 10.802470 11.704425 12.592462 9.595793 10.861817 9.709770 8.367209 10.237996 8.285036 10.392454 9.661724  [25] 10.157287 11.316243 11.704162 10.071253 10.946730 12.086343 9.890082 11.917080 11.417961 7.676059 8.098453 12.827042  [37] 11.329353 9.468903 8.841331 9.944871 10.950584 9.594285 7.285735 9.925839 10.821925 9.491220 11.411792 13.006744  [49] 12.860287 10.824921 10.624312 12.894464 10.572882 9.890082 10.871086 12.471259 12.358919 11.561567 10.863758 13.289982  [61] 11.319087 7.368077 9.351902 12.202362 8.343320 12.323452 11.442978 11.559852 10.670809 9.411366 11.319087 10.868688  [73] 10.985057 9.912038 9.713317 11.784993 11.559852 12.237056 11.132399 7.671625 12.310893 10.977456 9.701921 11.417961  [85] 11.564784 13.117474 11.497717 12.249933 12.135165 11.118144 10.857131 8.628413 11.335445 12.135165 10.710776 9.111135  [97] 12.074009 11.620469 12.059014 12.456895 10.736874 12.301296 12.358919 10.008096 12.761367 11.541522 9.388610 12.108226  [109] 10.116340 12.372545 10.569899 11.258655 10.736778 10.627081 9.799663 12.275930 10.821925 12.092924 12.137143 10.319863  [121] 11.090359 12.148449 10.363983 10.302992 9.438552 11.154514 10.237996 11.789077 10.746307 11.923825 10.923944 12.847512  [133] 10.737209 9.340717 11.716326 8.863924 12.531170 8.891643 10.861817 8.414120 11.620469 10.962488 11.538453 10.840653  [145] 11.038216 11.460223 11.019017 11.668100 9.037366 11.755941 10.843308 11.258210 9.101561 10.757247 10.597653 11.051895  [157] 10.627081 10.663375 10.853856 11.946578 10.641828 11.180195 12.970907 10.392454 11.099879 10.087034 11.665876 12.531170  [169] 9.373763 10.819401 11.444981 12.901969 11.758390 11.154514 10.033889 10.928228 9.454175 11.263282 9.725580 10.905135  [181] 8.929461 12.805477 8.869082 11.610064 10.455240 10.994314 12.249933 12.225409 12.275930 9.989269 12.367476 10.757247  [193] 9.761007 12.761367 9.864622 9.514927 11.839496 11.697716 11.037480 11.929300 11.693384 9.167555 12.636674 12.102863  [205] 10.935067 10.614061 11.319087 9.224818 10.757247 10.002121 11.436276 11.099205 11.309340 11.094889 10.584312 11.221552  [217] 12.202362 6.968670 13.274240 12.297478 12.202362 12.847512 11.898236 9.685171 10.385990 11.029102 11.824506 12.557493  [229] 10.558914 11.215192 10.420536 11.405073 9.425742 12.329652 10.343258 11.575580 12.074009 8.999640 12.236728 10.924477  [241] 10.682842 12.289579 11.090359 11.567047 9.702366 9.362696 11.214640 11.288270 9.886532 11.154514 12.367476 12.095600  [253] 12.456895 11.318605 10.149620 12.202362 12.521233 11.846110 12.367476 11.851888 10.731252 10.802470 12.275334 10.512784  [265] 12.102863 13.112815 12.372545 10.756839 11.378552 9.266228 11.270841 11.668100 12.743584 9.782316 10.651212 11.240171  [277] 11.810406 12.225409 10.910912 10.497690 10.731597 10.522483 11.123487 9.123845 9.246001 10.211169 10.664456 9.440385  [289] 10.655395 8.402365 9.005635 12.203217 11.883849 11.417961 12.367615 9.795650 9.798938 11.128477 13.006744 11.549328  [301] 11.342099 9.389287 12.929764 11.214640 11.285070 10.985057 10.349575 10.622029 11.935149 11.946578 10.757247 11.697716  [313] 9.033085 12.512163 11.895647 12.203217 13.238247 6.883631 10.249549 10.519094 10.715826 11.685780 10.887404 9.246602  [325] 10.135388 12.249933 9.896597 11.814012 11.361774 9.993957 9.390811 10.185604 11.669542 11.814012 11.662270 9.360096  [337] 10.025840 10.776208 11.322529 12.053573 10.624312 8.899393 10.887404 9.955155 12.959985 11.907965 11.500011 11.755941  [349] 13.094371 11.249331 10.575786 10.049451 12.805477 13.128592 8.886015 13.001859 12.062564 11.319087 11.270686 10.444127  [361] 11.758957 10.037140 10.861817 12.304372 9.772161 11.197998 12.914119 11.405073 10.682842 12.775200 11.015616 12.103267  [373] 11.779295 12.531170 13.112815 10.142772 13.206017 12.671727 11.606941 10.726507 10.686503 10.802470 9.160429 10.935373  [385] 10.742263 9.965060 10.016012 11.004827 11.285070 8.916065 11.318605 10.135388 12.367476 9.170624 9.514927 10.742263  [397] 11.517626 12.060507 8.808516 7.927306 11.057098 11.221552 12.380816 11.917967 11.030325 12.854196 9.036435 12.453618  [409] 11.789077 10.090626 11.029174 8.040192 10.580632 12.059014 11.051895 12.456895 10.737209 11.090359 12.445335 11.561567  [421] 11.928473 12.542804 11.316243 10.156622 10.842798 11.269097 8.766913 10.349575 9.912038 10.742253 9.072401 8.543487  [433] 10.777370 12.413435 9.225226 8.431266 10.486373 11.613030 7.602163 10.568397 11.758957 12.666797 11.208385 12.105333  [445] 12.137386 8.981633 12.847512 11.575407 10.007052 10.519094 11.653799 12.367476 13.174332 12.202362 11.854550 10.737209  [457] 10.519094 11.620469 12.557493 10.392454 9.772161 12.445335 8.672210 10.808461 12.864526 12.090214 11.814012 12.761367  [469] 11.378552 10.696499 11.221552 10.641908 11.910608 12.656348 10.210475 11.206875 11.420949 12.108226 11.731331 7.983684  [481] 12.074009 11.103755 10.621573 9.965060 9.590538 9.686177 11.555387 10.311782 11.390496 11.369685 8.918545 10.647292  [493] 12.970907 12.949473 8.618902 12.453622 10.675786 12.202362 10.596851 10.910912 10.682842 12.338519 10.651548 11.057098  [505] 11.444981 12.542804 10.802470 11.629663 9.306293 10.150769 11.863771 9.946850 10.859899 10.624312 12.731364 11.235653  [517] 11.090359 13.128592 11.620469 10.529652 11.384838 12.574186 11.821391 12.479548 11.275587 11.604719 11.658993 9.182825  [529] 12.202362 8.943304 10.670809 11.453698 11.451210 10.116048 11.917967 10.954452 8.552186 11.887141 11.590248 10.252637  [541] 13.001859 12.135165 9.437208 10.673458 12.340921 9.445675 8.102439 9.438352 9.410125 11.629663 10.735683 9.880389  [553] 12.375693 10.644504 10.392454 9.984320 10.180623 9.280949 12.340921 10.855133 12.054559 8.857870 9.982255 10.621573  [565] 10.585550 10.657394 11.782404 6.949444 10.206390 10.098189 12.249933 8.929219 9.592224 10.596922 13.006744 12.901969  [577] 12.544647 9.806522 9.787011 11.917967 9.053798 10.923944 9.330643 11.267196 10.780134 10.400679 11.946578 8.953737  [589] 12.456895 8.847708 11.755941 11.131003 11.081049 11.038216 7.520051 9.349121 11.413946 12.475760 8.413771 11.316243  [601] 11.978337 11.255828 9.898463 10.207425 12.358510 10.473137 12.636674 12.225409 10.715826 10.225797 11.420949 10.608551  [613] 10.580352 9.517086 8.311560 11.288295 8.813736 10.652902 10.670809 11.898236 11.308271 12.218191 6.489918 9.608780  [625] 11.978404 13.291810 10.140219 12.057599 11.325534 9.959597 12.202362 10.756839 12.619693 11.669818 11.413946 11.758957  [637] 11.065044 11.767533 11.789077 11.775265 10.497681 11.690632 10.803910 10.895084 12.137386 10.371097 12.094034 10.132325  [649] 10.852265 8.986719 9.075618 12.105333 10.087034 10.214491 12.847512 10.746307 11.133895 9.933755 10.818399 11.013885  [661] 11.702863 12.359528 10.985150 8.894301 9.299862 11.119036 12.039019 11.558957 11.575580 9.810415 10.645897 7.177011  [673] 11.041374 8.868241 10.409500 11.878436 11.222784 10.331518 10.586069 11.803053 11.751137 9.223830 8.884417 12.563973  [685] 10.857131 11.156135 10.663375 9.945550 12.074009 9.984320 11.579936 12.901969 11.237011 12.876590 11.123005 10.041393  [697] 10.645897 11.541522 11.784993 10.149620 11.911018 12.103267 11.662270 9.928444 11.337421 12.047591 8.897186 12.041337  [709] 11.081049 7.854469 12.141431 9.751983 12.565665 10.628407 11.146996 9.185429 11.132231 9.819982 11.444454 10.169503  [721] 10.682842 10.113442 11.411792 11.034830 10.696499 11.215601 10.116048 11.258655 11.687206 10.267359 11.378552 11.605695  [733] 12.338519 10.614061 11.174694 11.016127 11.973962 12.481944 9.674063 10.651212 13.128592 11.506679 10.624312 9.152040  [745] 10.244056 10.878770 12.743584 13.006744 12.249933 11.567522 12.671727 10.818774 9.565364 11.474170 11.013928 11.685780  [757] 10.673458 12.157803 9.965060 10.177891 11.012932 9.856553 10.750429 12.225409 11.310438 10.558914 10.026672 11.258210  [769] 10.653851 8.960303 10.819315 12.901969 10.908876 11.206875 10.670809 12.019271 10.923300 12.475760 12.715429 11.310438  [781] 11.215601 12.827042 9.091430 8.202998 10.473137 10.382395 10.040520 11.375549 12.203217 11.254036 10.816117 11.090359  [793] 5.739832 11.015616 11.326456 11.100793 11.898236 8.718502 10.727766 10.662335 10.702366 11.810870 8.628774 12.218191  [805] 10.816644 9.946850 10.001391 10.731597 8.325996 12.521233 10.630526 12.106951 10.624312 10.142772 13.212450 11.782404  [817] 10.515530 12.864526 10.169503 9.799010 12.200470 11.789077 10.943924 11.039885 11.594491 8.402365 10.742263 11.143214  [829] 9.368837 11.335445 12.043976 10.640271 11.137648 11.824506 12.214285 11.028781 11.106380 11.308271 11.851888 9.687215  [841] 8.395133 12.197454 10.857131 12.531170 9.747989 9.201926 12.338519 10.097386 11.814012 11.697716 11.586192 8.910629  [853] 13.074511 11.946578 10.477228 11.237011 10.275101 11.215601 11.538453 11.051895 12.451428 6.161517 12.135165 9.946850  [865] 13.128592 12.313282 8.851196 11.626959 12.456895 9.077732 9.543759 12.059014 12.330002 13.344627 11.827691 9.468903  [877] 12.475760 12.442480 10.683959 11.575407 12.544647 11.887141 12.249933 10.558321 10.670809 9.940952 11.504599 10.843308  [889] 9.622704 13.218754 12.275930 11.661071 8.119809 11.197998 12.479238 7.870608 10.049451 10.720671 10.832232 8.627828  [901] 12.102863 8.884417 11.444981 11.604719 10.371149 10.275101 11.174694 11.559852 11.291452 9.408202 7.177011 10.645897  [913] 9.643135 12.364496 10.682842 12.236728 9.251692 12.737277 11.692004 12.824055 9.396444 10.444127 11.753997 11.665876  [925] 11.417961 9.993957 10.736874 11.405201 12.585340 11.308271 11.016127 9.340986 9.690119 9.994955 13.206017 10.802470  [937] 10.818399 11.073216 11.335445 10.349654 9.387085 11.316243 10.512784 12.073591 12.761367 11.099879 11.361774 11.013424  [949] 12.901969 11.250066 8.881910 11.904969 11.697716 10.885705 11.755941 10.895532 10.380311 12.341365 11.078886 11.962855  [961] 11.375851 10.699739 9.327006 9.912038 12.445335 12.059014 9.772161 10.673458 8.674949 11.154514 10.105012 9.626315  [973] 12.124198 11.564784 11.355003 11.774148 11.419056 10.349654 8.382575 11.693377 10.614061 11.031798 9.294884 10.699739  [985] 12.010788 12.456895 10.169443 12.121956 11.477507 12.102863 10.690226 11.384838 11.755941 8.628774 12.440616 12.671727  [997] 10.558321 12.338519 10.853825 12.249933  [ reached getOption("max.print") -- omitted 2600 entries ]  > summary(gbmFit\_dead)  var rel.inf  age age 60.9549510  other\_disease other\_disease 5.0883333  diabetes diabetes 4.7258973  pneumonia pneumonia 4.3945709  renal\_chronic renal\_chronic 4.3564975  hypertension hypertension 4.3290353  contact\_other\_covid contact\_other\_covid 3.7024914  obesity obesity 3.2963920  cardiovascular cardiovascular 3.1756628  inmsupr inmsupr 2.2868121  copd copd 2.1455230  tobacco tobacco 0.8325123  asthma asthma 0.7113210  > prediction\_dead = predict.gbm(gbmFit\_dead, newdata=covid\_dead\_test[], type="response");  Using 1000 trees...  텍스트, 스크린샷, 라인, 그래프이(가) 표시된 사진  자동 생성된 설명  > summary(prediction);  Min. 1st Qu. Median Mean 3rd Qu. Max.  0.02966 0.28037 0.50000 0.50000 0.71963 0.97034  > prediction\_dead  [1] 11.906391 12.249933 12.609922 11.417961 10.785523 12.202362 11.187778 10.405954 10.405331 12.135165 9.130760 9.862998  [13] 8.633583 8.990883 10.192231 9.025150 9.443307 9.377692 12.557493 10.362785 10.580352 10.177891 10.612727 10.646944  [25] 11.520033 7.987922 11.164947 8.419568 11.034830 8.402365 11.413946 9.493797 11.322529 11.051895 8.395133 7.869271  [37] 7.827210 9.794137 10.985057 12.585340 9.724893 8.041350 8.402365 11.213336 7.659258 10.149620 8.711566 8.704659  [49] 12.133104 8.816336 11.887221 11.199307 7.497029 12.338519 12.446803 8.906060 10.485799 8.355858 10.549075 8.149537  [61] 11.215601 8.347860 7.608532 10.690226 9.389141 8.845349 9.262072 11.538453 11.665876 9.771886 12.338519 12.671727  [73] 9.289706 11.025210 9.692281 11.384838 12.367476 11.096270 10.757247 7.718390 10.769860 7.613504 9.640274 12.445335  [85] 8.074315 9.848042 11.411792 11.854550 10.664841 10.575786 8.476569 9.254882 9.692294 10.015140 12.330002 8.952165  [97] 10.734431 9.869352 7.070932 9.140597 10.616641 12.372382 9.692281 9.574844 11.923730 11.573452 10.896383 10.895084  [109] 10.029290 10.871086 10.002121 8.943304 10.756839 12.108226 10.261783 8.584113 9.976235 8.161940 11.973023 7.941180  [121] 7.456632 6.282479 11.883796 10.844953 12.225409 8.883694 11.221552 8.519485 9.350916 10.608551 9.608780 9.850673  [133] 11.285070 10.306725 11.316243 10.237996 10.310911 12.338519 11.215601 9.952864 9.686177 10.011737 11.946578 9.054420  [145] 10.612727 10.733689 8.402365 11.637563 9.928035 7.512478 10.304527 11.411792 10.935373 9.813320 9.781626 11.210455  [157] 8.690230 13.135926 11.361774 8.650233 8.908878 10.966421 9.502012 11.107518 12.367476 10.608551 6.750278 10.619449  [169] 10.000943 9.305253 8.742120 13.094371 5.739832 8.621915 10.533881 7.546737 10.553973 12.135165 10.624312 8.402365  [181] 12.456895 14.350137 8.520024 10.438969 10.223259 9.914541 10.871086 11.405073 11.456291 12.241369 11.417961 10.321613  [193] 8.763836 7.718390 11.744509 12.521233 10.547901 10.298739 10.500746 12.508093 8.781569 11.215601 11.862803 8.621915  [205] 6.360009 9.598042 11.755941 12.202362 11.498674 11.363301 12.483666 11.551237 10.549075 11.042399 8.402365 11.061159  [217] 8.861685 9.846020 10.821925 5.286954 10.059810 10.331518 8.402365 10.050434 10.878602 7.048481 10.288098 10.580352  [229] 7.638452 8.792563 9.528863 9.905839 8.834027 6.516542 9.993957 12.109163 9.438552 11.320652 12.531170 9.565364  [241] 11.453698 15.493751 10.682842 10.863758 12.393232 11.549328 10.853305 9.872134 10.102161 10.696499 12.453622 9.928444  [253] 12.544647 8.376169 8.277317 9.434399 12.456895 12.812917 10.584312 10.406207 12.531170 12.531170 10.321653 10.302992  [265] 11.453698 11.165826 10.392454 8.707679 9.955155 12.047591 12.521233 11.081049 11.559852 10.024572 11.083589 10.402269  [277] 11.245293 12.102863 10.621573 9.224818 11.559852 12.761367 10.576047 10.566711 8.718502 9.179849 11.361774 9.072401  [289] 13.112815 11.106407 12.592462 11.001772 8.593646 10.857131 12.109344 10.899529 8.431266 10.200539 10.142772 10.142772  [301] 10.693509 10.806094 10.756839 13.006744 8.633357 10.690226 12.329652 7.901606 10.247416 11.946578 11.662270 11.668100  [313] 11.755941 10.611494 11.206875 11.214640 12.747204 11.665876 10.756839 10.221431 12.565665 11.361774 11.962855 11.111734  [325] 11.188973 9.306293 12.285820 9.608780 10.560522 9.429739 10.557955 11.620469 11.605695 12.544647 10.954211 10.549075  [337] 11.411792 9.886532 11.285070 11.528679 9.072401 11.758390 11.453698 10.179136 10.628407 9.697245 8.807652 11.145909  [349] 10.668424 9.061166 10.861817 12.805477 9.417806 9.812289 11.549328 9.077732 11.538453 11.685780 12.970907 11.538453  [361] 10.575786 12.121956 11.269097 11.308271 12.501900 12.456895 9.608780 10.923351 10.895532 10.409500 8.314666 9.911612  [373] 9.589438 9.434596 10.002121 12.805477 10.149620 11.342099 9.466719 10.200539 12.452614 11.601736 10.597653 11.316243  [385] 10.306490 10.868688 10.802470 10.771182 10.486373 13.520746 10.258924 11.687206 11.288295 10.142772 9.798938 10.547901  [397] 11.417961 13.137960 10.929583 10.549075 6.277150 10.568397 10.899635 11.602954 9.075982 11.147169 8.442235 11.237920  [409] 9.872362 12.273115 10.935166 10.598487 9.993957 10.489647 12.643438 8.646944 9.817286 7.955431 10.616641 10.946730  > comparison\_dead=cbind(covid\_dead\_test,prediction\_dead);  > comparison\_dead=as.data.frame(comparison\_dead);  > comparison\_dead$prediction\_dead = round(comparison\_dead$prediction\_dead);  > print(paste("test 건수 : ", nrow(covid\_dead\_test)));  [1] "test 건수 : 420"  > # 투병일수 예측성공 기준 설정  > deadPredictCorrectCreteria = 5;  > deadPredictCorrect = comparison\_dead[abs(comparison\_dead$day\_cnt-comparison\_dead$prediction\_dead)<=deadPredictCorrectCreteria, 0];  > print(paste("투병일수 예측성공 건수(",deadPredictCorrectCreteria,"일) : " , nrow(deadPredictCorrect)));  [1] "투병일수 예측성공 건수( 5 일) : 225"  > print(paste("투병일수 예측 정확도(",deadPredictCorrectCreteria,"일) : ", nrow(deadPredictCorrect) / nrow(covid\_dead\_test)));  [1] "투병일수 예측 정확도( 5 일) : 0.535714285714286" |
|  |
| |  | | --- | | > | |