> 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

> library(class);

> train\_target = covid\_train[,16];

> train\_target;

[1] 2 2 2 2 2 2 2 1 2 2 1 2 2 1 2 1 1 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 1 2 1 2

[64] 1 2 2 2 1 2 2 2 1 2 2 2 2 2 2 1 2 1 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 1 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[127] 2 2 1 2 2 2 1 2 2 2 1 2 2 2 1 2 2 1 2 1 2 1 2 1 2 2 1 1 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 1 2 2 2

[190] 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 1 1 2 2 1 2 1 1 2 1 1 1 1 2

[253] 2 1 2 1 2 2 2 2 1 2 2 2 2 1 2 2 1 1 1 2 2 1 2 2 2 2 2 1 2 2 1 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 2 2 2 2

[316] 2 2 2 2 2 2 2 1 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 2 1 1 1 1 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2

[379] 2 1 2 2 2 2 2 2 2 1 1 2 2 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 1 1 2 2 1 2 2 2 2 2 2 2 2 1 1 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 1 2 2

[442] 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 1 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2

[505] 2 2 2 2 2 2 1 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 1 2 2 2 1 2 2 2 2 2 2 1 2 2 1 2 2 2 1 2 2 2 2 2

[568] 1 2 2 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 1 1 2 2 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 1 2 2

[631] 1 2 2 1 2 1 2 2 2 2 1 2 2 1 2 2 1 2 2 1 1 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 1 1 2 1 1 2 2 1 1 2 1 2 1 2 1 2 1 2 1 2 2 1 1 2 2 2

[694] 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 1 1 1 2 1 1 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 1 2 1 2 1 1 1 2 1 2 2 2 1

[757] 2 1 2 2 2 2 1 1 2 2 1 2 2 1 1 2 1 2 1 2 2 1 2 2 2 2 2 1 2 1 1 2 1 2 1 2 2 2 2 2 1 2 2 2 2 2 1 1 1 1 2 2 2 1 2 2 2 2 1 2 2 2 2

[820] 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 1 1 1 2 2 2 2 1 1 2 1 2 2 2 1 2 2 2 1 2 1 1 2 2 2 2 2 2 2 2 2 1 1 2 2 1 2 2 2 1 2

[883] 1 1 2 1 2 2 1 2 2 2 2 1 1 2 2 2 1 2 2 2 2 1 1 2 1 2 2 2 1 1 2 1 2 2 2 1 1 1 1 1 2 2 2 2 2 1 1 1 1 1 2 1 1 1 2 2 2 1 2 2 2 2 2

[946] 2 2 2 1 2 1 2 2 1 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 1 1 1 1 2 2 2 1 2 2 1 2 2

[ reached getOption("max.print") -- omitted 31356 entries ]

Levels: 1 2

> test\_target = covid\_test[,16];

> test\_target;

[1] 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2

[64] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2

[127] 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2

[190] 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 2 1 2 2 1 2 2 2 2 2 1 2 2 2 2

[253] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2

[316] 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 2 1 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[379] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[442] 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2

[505] 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[568] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2

[631] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[694] 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2

[757] 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[820] 1 2 2 2 2 2 2 2 2 1 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

[883] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2

[946] 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2

[ reached getOption("max.print") -- omitted 4920 entries ]

Levels: 1 2

> prediction = knn(covid\_train[,1:15], covid\_test[,1:15], cl=train\_target, k=3, prob=F);

> # prediction

> table(test\_target, prediction)

prediction

test\_target 1 2

1 1752 1208

2 315 2645

> comparison=cbind(covid\_test,prediction);

> comparison=as.data.frame(comparison);

> print(paste("test 건수 : ",nrow(covid\_test)));

[1] "test 건수 : 5920"

> predictCorrect = comparison[comparison$is\_dead == comparison$prediction,];

> print(paste("사망여부 예측성공 건수 : ", nrow(predictCorrect)));

[1] "사망여부 예측성공 건수 : 4397"

> print(paste("사망여부 예측 정확도 : " ,nrow(predictCorrect)/nrow(covid\_test))); # 74%

[1] "사망여부 예측 정확도 : 0.742736486486486"

> dead\_train\_target = covid\_dead\_train[,16];

> dead\_train\_target;

[1] 10 7 1 6 3 16 21 14 30 20 16 13 10 16 7 8 8 10 14 3 10 7 11 22 5 4 11 6 10 5 14 10 7 6 10 8 3 2 5 5 7 11

[43] 3 8 4 7 15 9 7 12 11 22 16 8 4 6 18 12 7 5 7 3 7 4 3 31 13 8 3 21 14 7 22 5 8 36 5 4 11 4 6 11 2 7

[85] 5 9 6 15 18 17 35 2 13 6 8 2 6 11 15 5 6 25 16 12 8 8 1 9 7 11 6 10 7 6 3 5 8 29 19 4 11 10 9 8 28 11

[127] 8 7 17 31 29 30 7 8 18 5 25 5 7 1 7 7 7 8 8 27 6 10 15 6 10 14 11 10 14 6 17 20 13 6 7 3 6 6 7 7 4 19

[169] 13 14 8 12 4 15 5 20 19 4 12 5 10 21 9 20 13 10 8 8 10 11 3 17 9 13 5 4 8 20 7 9 8 8 28 3 7 9 10 14 7 10

[211] 7 9 10 12 7 9 4 7 6 4 4 27 1 9 23 4 14 7 7 17 27 21 31 21 10 22 10 11 22 48 16 16 12 19 9 6 10 17 7 12 4 11

[253] 17 6 5 14 14 9 18 4 7 10 4 7 13 12 13 14 10 10 5 16 36 17 8 29 19 31 7 2 11 2 18 7 17 5 15 12 11 5 11 12 54 6

[295] 12 7 6 9 10 2 2 10 9 17 7 8 14 9 8 2 4 11 14 8 11 33 10 7 3 8 10 4 24 5 9 4 4 6 14 19 8 8 17 16 7 12

[337] 16 9 13 4 13 8 3 15 8 8 9 6 11 6 6 17 13 8 4 17 23 5 1 12 12 4 2 9 7 3 27 11 9 31 9 4 8 7 12 15 30 19

[379] 12 5 6 11 3 4 34 19 3 11 5 5 10 21 19 10 4 4 1 7 13 5 6 6 18 5 7 30 5 4 4 11 11 8 3 7 8 1 4 17 9 12

[421] 24 8 7 25 5 6 12 5 4 27 8 7 5 9 6 4 13 7 6 4 23 27 15 12 10 2 9 41 7 5 12 6 2 8 20 15 15 24 6 10 19 5

[463] 12 12 13 4 6 7 14 12 4 12 10 3 9 4 5 10 16 8 25 9 16 8 18 6 15 5 8 6 4 3 18 7 17 5 5 24 8 26 10 21 7 11

[505] 7 4 17 11 7 7 2 19 9 11 48 13 6 17 2 8 6 9 1 24 7 4 13 12 21 15 7 36 21 14 1 16 3 7 5 11 14 16 3 14 9 17

[547] 1 5 6 12 4 6 4 11 6 12 5 6 8 12 10 9 9 10 27 11 9 8 14 11 10 8 6 15 6 4 6 14 8 9 8 15 3 12 6 40 14 3

[589] 13 7 7 22 17 8 6 11 19 11 10 8 19 24 8 18 9 19 10 18 15 7 11 9 10 7 10 15 13 10 12 11 12 26 8 12 9 8 11 12 18 11

[631] 5 7 29 5 6 13 5 17 7 11 15 9 3 11 11 12 3 14 5 5 3 22 21 8 30 4 2 6 5 4 9 18 10 13 3 17 13 8 6 14 6 6

[673] 17 6 3 15 6 8 3 4 31 7 6 18 20 22 9 7 20 5 9 8 0 41 11 4 8 3 7 4 14 7 12 0 16 12 6 17 4 4 4 12 19 8

[715] 13 10 10 13 12 10 7 21 8 13 19 11 16 23 10 8 7 9 19 11 7 8 9 4 6 15 8 6 16 5 14 18 16 21 9 12 5 5 8 10 4 7

[757] 14 5 8 6 7 15 6 10 9 10 7 3 6 6 16 14 1 7 25 29 24 9 9 13 12 7 7 5 13 3 9 9 4 11 4 14 4 11 9 12 8 5

[799] 11 16 5 13 6 20 3 20 3 27 11 11 13 16 4 5 31 16 37 8 2 3 20 2 18 15 7 5 5 16 7 4 17 11 9 12 11 16 21 7 7 16

[841] 7 15 10 12 10 18 3 24 8 5 9 18 6 13 12 11 4 10 8 11 31 4 8 6 20 12 11 9 13 8 4 5 12 20 6 8 17 14 14 9 5 6

[883] 8 13 11 4 16 13 5 11 9 24 2 8 4 12 18 17 7 2 11 10 10 0 13 4 23 9 12 14 5 4 12 27 12 5 12 13 7 27 18 4 9 3

[925] 13 5 4 16 4 5 9 18 17 9 27 6 9 5 10 4 12 13 7 26 9 15 16 7 7 10 1 14 11 14 5 12 12 32 10 14 7 7 10 6 9 10

[967] 5 12 6 10 6 10 13 24 19 13 11 2 10 5 9 5 5 19 8 42 10 10 5 6 11 7 7 4 13 8 11 11 6 24

[ reached getOption("max.print") -- omitted 2600 entries ]

> dead\_test\_target = covid\_dead\_test[,16];

> dead\_test\_target;

[1] 3 5 16 4 18 10 9 6 6 8 3 11 25 6 9 5 4 1 13 4 8 18 9 21 10 2 9 7 3 6 18 7 6 9 5 10 2 11 9 22 4 4

[43] 4 1 2 6 11 2 7 7 19 3 2 5 12 1 6 3 7 4 13 5 16 5 4 2 6 19 14 0 5 18 10 11 14 6 1 4 10 5 11 14 6 28

[85] 4 1 10 29 3 13 4 2 7 1 7 1 7 5 27 1 21 11 8 34 5 28 10 4 21 15 2 7 24 9 6 93 2 7 8 5 1 6 18 20 9 5

[127] 14 2 18 4 2 9 10 9 3 3 4 7 6 25 1 3 15 3 17 7 12 3 6 6 9 25 6 9 7 21 5 7 28 3 3 10 5 8 21 8 12 2

[169] 10 1 5 19 10 4 4 9 9 16 4 9 2 0 2 7 9 5 11 4 3 13 3 2 3 16 30 7 8 4 19 6 5 8 7 4 3 5 15 0 5 7

[211] 11 5 49 3 3 3 3 9 9 6 9 3 4 6 7 1 6 12 10 14 12 13 18 4 17 8 10 1 6 6 13 2 8 7 2 21 18 22 4 10 6 4

[253] 7 12 8 16 3 0 14 7 2 22 15 4 12 4 14 6 29 6 7 12 4 6 21 4 26 5 8 12 7 4 5 7 2 2 13 16 10 14 4 7 9 7

[295] 26 6 7 9 14 2 7 4 7 3 3 3 22 9 7 17 4 10 13 18 3 5 3 19 2 19 15 8 3 3 7 11 7 7 0 13 5 5 17 6 8 16

[337] 4 21 8 3 3 6 9 4 5 9 4 3 14 9 9 13 8 6 9 17 24 18 7 7 7 3 5 9 4 6 18 8 21 3 5 4 6 10 14 9 9 14

[379] 15 3 15 12 32 11 9 43 13 28 1 19 5 9 6 5 6 8 20 15 8 21 5 5 12 9 9 7 3 27 7 1 4 12 18 10 24 8 3 0 21 22

> prediction\_dead = knn(covid\_dead\_train[,1:15], covid\_dead\_test[,1:15], cl=dead\_train\_target, k=3, prob=F);

> comparison\_dead=cbind(covid\_dead\_test,prediction\_dead);

> comparison\_dead=as.data.frame(comparison\_dead);

> comparison\_dead$prediction\_dead = round(comparison\_dead$prediction\_dead);

Error in Math.factor(c(13L, 5L, 18L, 8L, 4L, 5L, 13L, 10L, 5L, 9L, 5L, :

요인(factors)에 대하여 의미있는 ‘round’가 아닙니다.

> print(paste("test 건수 : ", nrow(covid\_dead\_test)));

[1] "test 건수 : 420"

> # 투병일수 예측성공 기준 설정

> deadPredictCorrectCreteria = 5;

> # comparison\_dead$prediction\_dead numeric으로 형변환하여 연산

> deadPredictCorrect = comparison\_dead[abs(comparison\_dead$day\_cnt-as.numeric(comparison\_dead$prediction\_dead))<=deadPredictCorrectCreteria, 0];

> print(paste("투병일수 예측성공 건수(",deadPredictCorrectCreteria,"일) : " , nrow(deadPredictCorrect)));

[1] "투병일수 예측성공 건수( 5 일) : 234"

> print(paste("투병일수 예측 정확도(",deadPredictCorrectCreteria,"일) : ", nrow(deadPredictCorrect) / nrow(covid\_dead\_test)));

[1] "투병일수 예측 정확도( 5 일) : 0.557142857142857"

>

> # 5일 : 54.5%, 7일 : 65%, 10일 : 80%