> setwd('C:/Rdirectory/data\_mining/data\_mining\_covid');

> covid\_train = read.csv("covid\_train.csv", header=T);

> covid\_train$is\_dead = ifelse(covid\_train$is\_dead == 2,0,1);

> str(covid\_train);

'data.frame': 32356 obs. of 16 variables:

$ sex : int 1 1 1 1 1 1 1 1 1 1 ...

$ patient\_type : int 2 2 2 2 2 2 2 2 2 2 ...

$ pneumonia : int 2 2 1 1 1 2 1 1 1 2 ...

$ age : num 25 52 51 67 59 52 54 78 80 40 ...

$ diabetes : int 2 2 2 1 1 1 2 2 1 2 ...

$ copd : int 2 2 2 2 2 2 2 2 2 2 ...

$ asthma : int 2 2 2 2 2 2 2 2 2 2 ...

$ inmsupr : int 2 2 2 2 2 2 2 2 2 2 ...

$ hypertension : int 2 2 2 1 2 1 2 1 1 2 ...

$ other\_disease : int 2 2 2 2 2 2 2 2 2 2 ...

$ cardiovascular : int 2 2 2 2 2 1 2 2 2 2 ...

$ obesity : int 2 1 2 1 2 2 2 1 1 2 ...

$ renal\_chronic : int 2 2 2 2 2 2 2 2 2 2 ...

$ tobacco : int 2 1 2 2 2 2 2 2 2 2 ...

$ contact\_other\_covid: int 1 1 1 1 1 2 2 1 1 2 ...

$ is\_dead : num 0 0 0 0 0 0 0 1 0 0 ...

> 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 0

2 1 1 0

3 2 1 0

4 2 1 0

5 2 1 0

6 2 2 0

> covid\_test = read.csv("covid\_test.csv", header=T);

> covid\_test$is\_dead = ifelse(covid\_test$is\_dead == 2,0,1);

> str(covid\_test);

'data.frame': 5920 obs. of 16 variables:

$ sex : int 1 1 1 1 1 1 1 1 1 1 ...

$ patient\_type : int 2 2 2 2 2 2 2 2 2 2 ...

$ pneumonia : int 2 2 1 1 1 1 2 2 2 1 ...

$ age : num 52 36 0 85 75 19 46 60 62 39 ...

$ diabetes : int 2 2 2 2 2 2 2 2 1 2 ...

$ copd : int 2 2 2 1 2 2 2 2 2 2 ...

$ asthma : int 2 2 2 2 2 2 2 2 2 2 ...

$ inmsupr : int 2 2 2 2 2 2 2 2 2 2 ...

$ hypertension : int 2 2 2 1 1 2 2 2 2 1 ...

$ other\_disease : int 1 2 2 2 2 2 2 2 2 2 ...

$ cardiovascular : int 2 2 2 1 2 2 2 2 2 2 ...

$ obesity : int 2 2 2 1 2 2 2 2 2 2 ...

$ renal\_chronic : int 2 2 2 2 2 2 2 2 2 2 ...

$ tobacco : int 2 2 2 2 2 2 2 2 2 2 ...

$ contact\_other\_covid: int 2 2 2 2 2 2 1 2 2 2 ...

$ is\_dead : num 0 0 1 0 0 0 0 0 0 0 ...

> 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 0

2 2 2 0

3 2 2 1

4 2 2 0

5 2 2 0

6 2 2 0

> covid\_dead\_train = read.csv("covid\_dead\_train.csv", header=T);

> str(covid\_dead\_train);

'data.frame': 3600 obs. of 16 variables:

$ sex : int 1 1 1 1 1 1 1 1 1 1 ...

$ patient\_type : int 2 2 2 2 2 2 2 2 2 2 ...

$ pneumonia : int 1 1 2 1 1 1 1 1 1 2 ...

$ age : int 78 65 58 78 92 70 71 65 56 72 ...

$ diabetes : int 2 2 2 2 2 2 1 1 2 2 ...

$ copd : int 2 2 2 2 2 2 1 2 2 2 ...

$ asthma : int 2 2 2 2 2 2 2 2 2 2 ...

$ inmsupr : int 2 2 2 2 2 2 2 2 2 2 ...

$ hypertension : int 1 2 2 2 1 2 1 1 2 1 ...

$ other\_disease : int 2 2 1 2 2 2 2 2 2 2 ...

$ cardiovascular : int 2 2 2 2 2 2 2 2 2 2 ...

$ obesity : int 1 1 2 1 2 1 1 1 2 2 ...

$ renal\_chronic : int 2 2 2 2 2 2 2 2 2 1 ...

$ tobacco : int 2 2 2 2 2 2 1 2 2 2 ...

$ contact\_other\_covid: int 1 2 1 1 2 2 1 2 2 2 ...

$ day\_cnt : int 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);

> str(covid\_dead\_test);

'data.frame': 420 obs. of 16 variables:

$ sex : int 1 1 1 1 1 1 1 1 1 1 ...

$ patient\_type : int 2 2 2 2 2 2 2 2 2 2 ...

$ pneumonia : int 1 1 1 1 1 1 1 1 1 1 ...

$ age : int 66 68 49 61 47 56 68 72 78 64 ...

$ diabetes : int 1 2 2 1 1 2 1 1 1 1 ...

$ copd : int 2 2 2 2 2 2 2 1 2 2 ...

$ asthma : int 2 2 2 2 1 2 2 2 2 2 ...

$ inmsupr : int 2 2 2 2 2 2 2 2 2 2 ...

$ hypertension : int 1 2 2 1 2 2 1 1 2 2 ...

$ other\_disease : int 2 2 2 2 2 2 2 2 2 2 ...

$ cardiovascular : int 2 2 2 2 2 2 2 2 2 2 ...

$ obesity : int 1 2 1 2 2 2 1 1 1 2 ...

$ renal\_chronic : int 2 2 2 2 2 2 2 2 2 2 ...

$ tobacco : int 2 2 2 2 2 2 2 2 2 2 ...

$ contact\_other\_covid: int 1 2 2 2 2 2 2 2 2 2 ...

$ day\_cnt : int 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

> logisticFit = glm(is\_dead~., data =covid\_train, family = binomial);

> summary(logisticFit);

Call:

glm(formula = is\_dead ~ ., family = binomial, data = covid\_train)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.1938 -1.0331 0.1002 0.9628 2.5469

Coefficients: (2 not defined because of singularities)

Estimate Std. Error z value Pr(>|z|)

(Intercept) 1.1843568 0.2839009 4.172 3.02e-05 \*\*\*

sex NA NA NA NA

patient\_type NA NA NA NA

pneumonia -1.5240030 0.0321392 -47.419 < 2e-16 \*\*\*

age 0.0283538 0.0007523 37.692 < 2e-16 \*\*\*

diabetes -0.2805133 0.0275182 -10.194 < 2e-16 \*\*\*

copd 0.0964173 0.0547015 1.763 0.07797 .

asthma 0.0621049 0.0706011 0.880 0.37904

inmsupr -0.0877809 0.0653804 -1.343 0.17940

hypertension -0.1291804 0.0284484 -4.541 5.60e-06 \*\*\*

other\_disease -0.1554985 0.0618984 -2.512 0.01200 \*

cardiovascular -0.0007630 0.0528927 -0.014 0.98849

obesity -0.1824579 0.0283143 -6.444 1.16e-10 \*\*\*

renal\_chronic -0.3180567 0.0539932 -5.891 3.85e-09 \*\*\*

tobacco 0.1576789 0.0605036 2.606 0.00916 \*\*

contact\_other\_covid 0.3335346 0.0280642 11.885 < 2e-16 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 44855 on 32355 degrees of freedom

Residual deviance: 38157 on 32342 degrees of freedom

AIC: 38185

Number of Fisher Scoring iterations: 4

> prediction = predict(logisticFit, newdata=covid\_test, type = "response");

Warning message:

In predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :

prediction from a rank-deficient fit may be misleading

> print(prediction);

1 2 3 4 5 6 7 8 9 10 11 12

0.22395214 0.13564511 0.20608690 0.78200396 0.71239429 0.30790042 0.12988930 0.23659304 0.30275166 0.47160644 0.35909413 0.52998193

13 14 15 16 17 18 19 20 21 22 23 24

0.46692670 0.08173393 0.50803707 0.73521004 0.25128221 0.72383548 0.26205178 0.14950024 0.06395997 0.54011057 0.14676921 0.72070948

25 26 27 28 29 30 31 32 33 34 35 36

0.61442913 0.26134527 0.13495748 0.21786466 0.40820809 0.30266888 0.06945972 0.45167220 0.07407453 0.62411763 0.17567534 0.13980910

37 38 39 40 41 42 43 44 45 46 47 48

0.63961213 0.49840017 0.71434981 0.41817153 0.67565902 0.10106234 0.10045427 0.21551982 0.70327081 0.05646719 0.08662176 0.25551906

49 50 51 52 53 54 55 56 57 58 59 60

0.33258021 0.78135753 0.58742153 0.47084491 0.75307526 0.21505257 0.69060214 0.51016391 0.23593965 0.13980910 0.55950121 0.38090084

61 62 63 64 65 66 67 68 69 70 71 72

0.22081983 0.27287663 0.37135337 0.27935886 0.77468454 0.37963385 0.13735326 0.40662508 0.73138606 0.10304559 0.47824454 0.46631437

73 74 75 76 77 78 79 80 81 82 83 84

0.24698788 0.27535268 0.07027782 0.24993164 0.06196715 0.57654004 0.36260474 0.48692462 0.15349438 0.10633196 0.54548152 0.32011447

85 86 87 88 89 90 91 92 93 94 95 96

0.46536325 0.13235496 0.13235496 0.54011057 0.51649154 0.53844311 0.20608690 0.44823319 0.39141845 0.57047661 0.13709784 0.28185601

97 98 99 100 101 102 103 104 105 106 107 108

0.39884487 0.08566854 0.26229686 0.05497539 0.12988930 0.08440458 0.23371199 0.68175348 0.78857040 0.25632883 0.42867734 0.49201163

109 110 111 112 113 114 115 116 117 118 119 120

0.15680296 0.76401410 0.04811820 0.73750582 0.35325419 0.69465639 0.82580079 0.16930057 0.63887427 0.75949136 0.15401376 0.08223905

121 122 123 124 125 126 127 128 129 130 131 132

0.56503310 0.10366755 0.22526199 0.28254367 0.05352078 0.09602581 0.14243224 0.53755458 0.38383965 0.24365801 0.76191453 0.14243224

133 134 135 136 137 138 139 140 141 142 143 144

0.12671830 0.49529708 0.11760034 0.12288849 0.17736160 0.50307682 0.11594822 0.40662508 0.16322598 0.79638341 0.66932510 0.45359163

145 146 147 148 149 150 151 152 153 154 155 156

0.51892254 0.07136019 0.65318447 0.12057468 0.22555989 0.79499557 0.69401440 0.18593690 0.11760034 0.20608690 0.15682127 0.13643390

157 158 159 160 161 162 163 164 165 166 167 168

0.14243224 0.19362367 0.10366755 0.82591052 0.13643390 0.06251759 0.38636823 0.28563944 0.08834948 0.63173663 0.75655269 0.46635156

169 170 171 172 173 174 175 176 177 178 179 180

0.47784855 0.36280070 0.38468546 0.52234652 0.09359260 0.28563944 0.47884786 0.71816851 0.08173393 0.11144626 0.15314136 0.62199125

181 182 183 184 185 186 187 188 189 190 191 192

0.05497539 0.17017012 0.60696914 0.32631665 0.11428511 0.21987749 0.81749189 0.40115260 0.11184217 0.13235496 0.05352078 0.60792394

193 194 195 196 197 198 199 200 201 202 203 204

0.30189141 0.23531245 0.66185833 0.41251629 0.72824893 0.12388108 0.47341951 0.41647861 0.43728588 0.44540714 0.15680296 0.28762329

205 206 207 208 209 210 211 212 213 214 215 216

0.60769053 0.47339886 0.03893256 0.82437868 0.29305997 0.05386176 0.74295778 0.73003930 0.12913269 0.63677417 0.04769732 0.24175241

217 218 219 220 221 222 223 224 225 226 227 228

0.29557483 0.20263173 0.09179825 0.63437454 0.40500681 0.06666169 0.11690570 0.09663675 0.10070027 0.55416052 0.58526230 0.48489984

229 230 231 232 233 234 235 236 237 238 239 240

0.27615248 0.21460870 0.59336221 0.15680296 0.16058836 0.18877144 0.72387173 0.11307333 0.63384973 0.22526199 0.27431755 0.13244260

241 242 243 244 245 246 247 248 249 250 251 252

0.31055853 0.60928935 0.38636823 0.09544482 0.59405699 0.20986829 0.57961047 0.44657394 0.11428511 0.17736160 0.49598849 0.32166244

253 254 255 256 257 258 259 260 261 262 263 264

0.67979820 0.11606801 0.30790042 0.16975875 0.29291145 0.67068810 0.53474352 0.45675106 0.61815719 0.52599657 0.12564479 0.08484867

265 266 267 268 269 270 271 272 273 274 275 276

0.33415883 0.31485438 0.20718713 0.68250856 0.10905653 0.70461661 0.09851541 0.44123146 0.44942098 0.19808914 0.33859969 0.55950121

277 278 279 280 281 282 283 284 285 286 287 288

0.17937348 0.71224866 0.13980910 0.34543846 0.11184217 0.38654914 0.19029850 0.10633196 0.50160749 0.45199994 0.54473144 0.55409785

289 290 291 292 293 294 295 296 297 298 299 300

0.52032631 0.55175378 0.05352078 0.16468190 0.66653295 0.07805914 0.76855023 0.33485777 0.08968333 0.20263173 0.07604289 0.12671830

301 302 303 304 305 306 307 308 309 310 311 312

0.30536181 0.44657394 0.21302931 0.08012422 0.53245427 0.48349530 0.48047211 0.26728718 0.73380248 0.39850793 0.11184217 0.09851541

313 314 315 316 317 318 319 320 321 322 323 324

0.20263173 0.12671830 0.12988930 0.42508571 0.08012422 0.48797612 0.32369722 0.53063602 0.17692375 0.19356244 0.23212144 0.30608988

325 326 327 328 329 330 331 332 333 334 335 336

0.76904282 0.31141665 0.55421648 0.56115460 0.26309685 0.54075602 0.57727839 0.85060766 0.16542570 0.17244414 0.62700690 0.16685078

337 338 339 340 341 342 343 344 345 346 347 348

0.10366755 0.08889152 0.80094256 0.16937745 0.82269700 0.81760570 0.27421027 0.24012428 0.36634729 0.62112415 0.51016391 0.76855023

349 350 351 352 353 354 355 356 357 358 359 360

0.07557269 0.09772856 0.21076468 0.59706293 0.50469111 0.72783786 0.11468980 0.35821434 0.50869507 0.21256508 0.26860359 0.16542570

361 362 363 364 365 366 367 368 369 370 371 372

0.38636823 0.46633627 0.72706258 0.55875555 0.14593076 0.55639633 0.11401013 0.56573403 0.16930057 0.56812446 0.72950301 0.75307526

373 374 375 376 377 378 379 380 381 382 383 384

0.10905653 0.70327081 0.26314603 0.34336355 0.40599000 0.12057468 0.09602581 0.39776200 0.79072871 0.58124486 0.16843535 0.23024879

385 386 387 388 389 390 391 392 393 394 395 396

0.60696914 0.10840615 0.19026692 0.54548152 0.13312756 0.09792099 0.48677548 0.43822665 0.22650409 0.31542260 0.63230723 0.21459387

397 398 399 400 401 402 403 404 405 406 407 408

0.59713203 0.46229564 0.63150938 0.11401013 0.73950800 0.13643390 0.60769053 0.22597456 0.12057468 0.26693500 0.69813411 0.59022324

409 410 411 412 413 414 415 416 417 418 419 420

0.68939921 0.46229564 0.21425095 0.36475867 0.25656260 0.51443108 0.05352078 0.57342664 0.30275166 0.46229564 0.60091065 0.33091287

421 422 423 424 425 426 427 428 429 430 431 432

0.72159615 0.30333698 0.53844311 0.58723661 0.07407453 0.18923510 0.06451289 0.63653123 0.52234652 0.50406022 0.16064124 0.07757753

433 434 435 436 437 438 439 440 441 442 443 444

0.19867536 0.32359838 0.68456602 0.52599657 0.15401376 0.23725949 0.24175241 0.71891402 0.57570239 0.13235496 0.36640618 0.61095876

445 446 447 448 449 450 451 452 453 454 455 456

0.32435979 0.68249724 0.41747869 0.13564511 0.60769053 0.24742097 0.46767971 0.44123146 0.28254367 0.39001928 0.25768526 0.28571948

457 458 459 460 461 462 463 464 465 466 467 468

0.78663458 0.05352078 0.13980910 0.54778878 0.03893256 0.20648949 0.25170155 0.13564511 0.68722350 0.29032854 0.13711074 0.50307682

469 470 471 472 473 474 475 476 477 478 479 480

0.58344694 0.70000223 0.33407158 0.43563517 0.37599757 0.70986089 0.54714494 0.13312756 0.10840615 0.24175241 0.67008385 0.39329389

481 482 483 484 485 486 487 488 489 490 491 492

0.13900394 0.61706028 0.45359163 0.07604289 0.23890185 0.66658933 0.16500849 0.18068764 0.55022644 0.11117725 0.78119020 0.74233116

493 494 495 496 497 498 499 500 501 502 503 504

0.68521885 0.08662176 0.49713537 0.63802336 0.39011258 0.58418193 0.54235865 0.51184091 0.35265293 0.26314603 0.59336221 0.12057468

505 506 507 508 509 510 511 512 513 514 515 516

0.30330771 0.46767971 0.70676137 0.08889152 0.32427384 0.14243224 0.45535276 0.45359163 0.46062780 0.54473144 0.45172575 0.39980236

517 518 519 520 521 522 523 524 525 526 527 528

0.49198439 0.08012422 0.22927805 0.47075176 0.55554954 0.09121482 0.16843535 0.33235029 0.44962446 0.63802336 0.14639080 0.15401376

529 530 531 532 533 534 535 536 537 538 539 540

0.54639531 0.20835637 0.10366755 0.12732586 0.08173393 0.23947261 0.51724692 0.17981918 0.09602581 0.31550729 0.32818529 0.64454570

541 542 543 544 545 546 547 548 549 550 551 552

0.65409099 0.15685482 0.40671971 0.11401013 0.31550729 0.74418906 0.46935086 0.65893862 0.51016391 0.59409187 0.58723661 0.08223905

553 554 555 556 557 558 559 560 561 562 563 564

0.10106234 0.74733297 0.25761025 0.23659304 0.55730598 0.14593076 0.56253903 0.56846409 0.62777335 0.60091065 0.51885921 0.17981918

565 566 567 568 569 570 571 572 573 574 575 576

0.57482601 0.68053207 0.66675887 0.53612916 0.12597749 0.67565902 0.60091065 0.40368568 0.61595783 0.53844311 0.14243224 0.45359163

577 578 579 580 581 582 583 584 585 586 587 588

0.41185740 0.58581125 0.53954183 0.75886379 0.45985478 0.44657394 0.21734965 0.75599512 0.13900394 0.29409528 0.21096869 0.16064124

589 590 591 592 593 594 595 596 597 598 599 600

0.41800623 0.23882819 0.06844763 0.12057468 0.15401376 0.50307682 0.18833500 0.76360602 0.16064124 0.06666169 0.32444859 0.51816734

601 602 603 604 605 606 607 608 609 610 611 612

0.18068764 0.09792099 0.48214757 0.16064124 0.66658933 0.12361371 0.58066733 0.70192849 0.39141845 0.19808914 0.25356715 0.42739654

613 614 615 616 617 618 619 620 621 622 623 624

0.77696211 0.70258616 0.56515111 0.55192181 0.52432301 0.71771045 0.51108507 0.10366755 0.37404497 0.38939311 0.69020973 0.48800855

625 626 627 628 629 630 631 632 633 634 635 636

0.31826514 0.64690179 0.12288849 0.22105375 0.10366755 0.72890569 0.80001513 0.38559120 0.21194872 0.19808914 0.62777335 0.16542570

637 638 639 640 641 642 643 644 645 646 647 648

0.54091245 0.22370990 0.81368242 0.10106234 0.54853801 0.65318447 0.33205978 0.44748517 0.32166244 0.69928141 0.30933241 0.11117725

649 650 651 652 653 654 655 656 657 658 659 660

0.37799636 0.50701768 0.52145392 0.57133526 0.16843535 0.64454570 0.30099596 0.68787334 0.53698211 0.54548152 0.11690570 0.08276420

661 662 663 664 665 666 667 668 669 670 671 672

0.70852910 0.16838042 0.62580318 0.72572876 0.26314603 0.10366755 0.47641835 0.16450122 0.11117725 0.03893256 0.03893256 0.62482700

673 674 675 676 677 678 679 680 681 682 683 684

0.74347677 0.14593076 0.51724692 0.42565903 0.11986485 0.75826333 0.66941477 0.27996825 0.73022981 0.16895230 0.17340399 0.77630500

685 686 687 688 689 690 691 692 693 694 695 696

0.52118237 0.16154821 0.12288849 0.59877380 0.66170531 0.12671830 0.76904282 0.12495622 0.60769053 0.35217606 0.75363733 0.70327081

697 698 699 700 701 702 703 704 705 706 707 708

0.57124763 0.78918435 0.15401376 0.09851541 0.38627525 0.45128579 0.14593076 0.44748517 0.06298299 0.53080523 0.45359163 0.53844311

709 710 711 712 713 714 715 716 717 718 719 720

0.35780087 0.78060332 0.60769053 0.09359260 0.37670756 0.13200797 0.66658933 0.09253011 0.61442913 0.11760034 0.09359260 0.62112415

721 722 723 724 725 726 727 728 729 730 731 732

0.22650409 0.63367264 0.73447197 0.15774443 0.57342664 0.43116287 0.09121482 0.30073664 0.73777372 0.09792099 0.12913269 0.45750174

733 734 735 736 737 738 739 740 741 742 743 744

0.71891402 0.39099000 0.17201289 0.12057468 0.75414189 0.31550729 0.45359163 0.09359260 0.11401013 0.22813937 0.66941477 0.18068764

745 746 747 748 749 750 751 752 753 754 755 756

0.32166244 0.05646719 0.14593076 0.54150715 0.23151017 0.22867248 0.63065115 0.15680296 0.73687061 0.29154101 0.21046559 0.61594557

757 758 759 760 761 762 763 764 765 766 767 768

0.48437736 0.16064124 0.12913269 0.24175241 0.38636823 0.18068764 0.11117725 0.04235329 0.10106234 0.10840615 0.45426571 0.27421027

769 770 771 772 773 774 775 776 777 778 779 780

0.49366185 0.19362367 0.08173393 0.08662176 0.60427625 0.35821434 0.38627525 0.38636823 0.16450122 0.16450122 0.50616030 0.61371226

781 782 783 784 785 786 787 788 789 790 791 792

0.10840615 0.24698788 0.08223905 0.50786502 0.36475867 0.46229564 0.47245132 0.51184091 0.13709784 0.52032631 0.12913269 0.77909236

793 794 795 796 797 798 799 800 801 802 803 804

0.19113486 0.55720602 0.20608690 0.72326669 0.26314603 0.57047661 0.44427478 0.31240985 0.46029083 0.10569591 0.29007601 0.21672323

805 806 807 808 809 810 811 812 813 814 815 816

0.22157506 0.08889152 0.10106234 0.63723080 0.82258569 0.67565902 0.76132608 0.69809748 0.72387173 0.15985033 0.70858842 0.22157506

817 818 819 820 821 822 823 824 825 826 827 828

0.36475867 0.29680005 0.47075176 0.58880097 0.24119834 0.24566912 0.14243224 0.21672323 0.44758214 0.75886379 0.65386941 0.39596472

829 830 831 832 833 834 835 836 837 838 839 840

0.10840615 0.73687061 0.56040953 0.34200552 0.60928935 0.80711670 0.10811412 0.12057468 0.19808914 0.57342664 0.62994626 0.57342664

841 842 843 844 845 846 847 848 849 850 851 852

0.39366721 0.44823319 0.23216666 0.63301023 0.36260474 0.08012422 0.25463304 0.12671830 0.15314136 0.12288849 0.78119020 0.68411464

853 854 855 856 857 858 859 860 861 862 863 864

0.70915341 0.21194872 0.63723080 0.40662508 0.16801208 0.59061935 0.73996473 0.67241039 0.61371226 0.54856500 0.76453404 0.31279495

865 866 867 868 869 870 871 872 873 874 875 876

0.33258021 0.10304559 0.07407453 0.58581125 0.15680296 0.14950024 0.14411746 0.36936885 0.68456602 0.52210221 0.27119134 0.34404590

877 878 879 880 881 882 883 884 885 886 887 888

0.17981918 0.70324062 0.24612507 0.68249724 0.39141845 0.18593690 0.48890177 0.30411654 0.28266470 0.49666831 0.18923510 0.09121482

889 890 891 892 893 894 895 896 897 898 899 900

0.43647867 0.51596971 0.28894772 0.37221434 0.16843535 0.22105375 0.62777335 0.29146001 0.09066002 0.12361371 0.38468546 0.11064503

901 902 903 904 905 906 907 908 909 910 911 912

0.35821434 0.09851541 0.68184138 0.07805914 0.46062780 0.55950121 0.10633196 0.18168330 0.31542260 0.20263173 0.61734722 0.35821434

913 914 915 916 917 918 919 920 921 922 923 924

0.46621677 0.25598605 0.33492301 0.25229901 0.43957749 0.65481936 0.19808914 0.20608690 0.31397524 0.56344597 0.37799636 0.32444859

925 926 927 928 929 930 931 932 933 934 935 936

0.08440458 0.45359163 0.11468980 0.16937745 0.12671830 0.09792099 0.37581051 0.62402257 0.32435979 0.27743759 0.20835637 0.45841679

937 938 939 940 941 942 943 944 945 946 947 948

0.08223905 0.20263173 0.09544482 0.04763403 0.09602581 0.73492306 0.60084982 0.61144254 0.19808914 0.55416052 0.09121482 0.35047273

949 950 951 952 953 954 955 956 957 958 959 960

0.51724692 0.21144391 0.18123400 0.76052417 0.14950024 0.31397524 0.37957556 0.11401013 0.67219467 0.70915341 0.45128579 0.10304559

961 962 963 964 965 966 967 968 969 970 971 972

0.12146098 0.58650320 0.55175378 0.28577810 0.58418193 0.54317292 0.21076468 0.10304559 0.30899440 0.14676921 0.20608690 0.08173393

973 974 975 976 977 978 979 980 981 982 983 984

0.38939311 0.47281306 0.56179448 0.15035558 0.15314136 0.17750580 0.10971821 0.55175378 0.13235496 0.64529706 0.44035030 0.12057468

985 986 987 988 989 990 991 992 993 994 995 996

0.73816980 0.81270183 0.15401376 0.62777335 0.64092563 0.14325396 0.45831620 0.06709968 0.30089711 0.36475867 0.70398663 0.36631513

997 998 999 1000

0.76346827 0.09792099 0.24465404 0.15401376

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

> prediction=round(prediction);

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

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

> comparison

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

7 1 2 2 46 2 2 2 2 2 2 2 2 2

8 1 2 2 60 2 2 2 2 2 2 2 2 2

9 1 2 2 62 1 2 2 2 2 2 2 2 2

10 1 2 1 39 2 2 2 2 1 2 2 2 2

11 1 2 2 60 1 2 2 2 1 2 2 1 2

12 1 2 1 43 2 2 1 2 1 2 2 1 2

13 1 2 1 33 1 2 2 2 2 2 2 2 2

14 1 2 2 16 2 2 2 2 2 2 2 2 2

15 1 2 2 88 1 2 2 2 1 2 2 2 2

16 1 2 1 79 2 2 2 2 1 2 1 2 2

17 1 2 2 65 2 2 1 2 2 2 2 2 2

18 1 2 1 66 1 2 2 2 2 2 2 2 1

19 1 2 2 56 2 2 1 2 1 2 2 1 2

20 1 2 2 40 2 2 2 2 2 2 2 2 2

21 1 2 2 24 2 2 2 2 2 2 2 2 2

22 1 2 1 65 2 2 2 2 2 2 2 2 2

23 1 2 2 51 2 2 2 2 2 2 2 2 2

24 1 2 1 81 2 2 2 2 2 2 2 2 2

25 1 2 1 64 2 2 2 2 2 2 2 2 2

26 1 2 2 70 2 2 2 2 2 2 2 1 2

27 1 2 2 43 2 2 2 2 1 2 2 2 2

28 1 2 2 68 2 2 2 2 2 2 2 2 2

29 1 2 2 77 2 2 2 2 2 2 2 2 1

30 1 2 1 20 1 2 2 2 2 2 2 2 2

31 1 2 2 17 2 2 2 2 1 2 2 2 2

32 1 2 2 80 1 2 2 2 1 2 1 2 2

33 1 2 2 24 2 2 2 2 2 2 2 2 2

34 1 2 1 51 1 2 2 2 1 2 2 2 2

35 1 2 2 54 2 2 2 2 1 2 2 2 2

36 1 2 2 49 2 2 2 2 2 2 2 2 2

37 1 2 1 57 1 2 1 1 2 2 2 2 2

38 1 2 1 33 2 2 2 1 2 2 1 2 1

39 1 2 1 70 1 2 2 2 2 2 2 2 2

40 1 2 2 64 1 2 2 2 1 2 2 2 1

41 1 2 1 59 1 2 2 2 1 2 2 2 2

42 1 2 2 36 2 2 2 2 2 2 2 2 2

43 1 2 2 24 2 2 2 2 2 2 2 2 2

44 1 2 1 2 2 2 2 2 2 2 2 2 2

45 1 2 1 78 2 2 2 2 2 2 2 2 2

46 1 2 2 2 2 2 2 2 2 2 2 2 2

47 1 2 2 30 2 2 2 2 2 2 2 2 2

48 1 2 2 67 2 1 2 2 2 2 2 2 2

49 1 2 1 23 2 2 2 2 2 2 2 2 2

50 1 2 1 75 1 1 2 2 1 2 2 1 2

51 1 2 1 60 2 2 2 2 2 2 1 2 2

52 1 2 2 62 1 2 2 1 1 2 2 1 1

53 1 2 1 77 1 2 2 2 2 2 2 2 2

54 1 2 2 48 2 2 2 1 1 2 2 2 2

55 1 2 1 55 1 2 2 2 1 2 2 1 2

56 1 2 1 49 2 2 2 2 2 2 2 2 2

57 1 2 1 3 2 2 2 1 2 2 1 2 2

58 1 2 2 49 2 2 2 2 2 2 2 2 2

tobacco contact\_other\_covid is\_dead prediction

1 2 2 0 0

2 2 2 0 0

3 2 2 1 0

4 2 2 0 1

5 2 2 0 1

6 2 2 0 0

7 2 1 0 0

8 2 2 0 0

9 2 2 0 0

10 2 2 0 0

11 2 2 0 0

12 2 2 0 1

13 2 2 0 0

14 2 2 0 0

15 2 2 0 1

16 2 2 1 1

17 2 2 0 0

18 1 2 1 1

19 2 2 0 0

20 2 2 0 0

21 1 1 0 0

22 2 1 0 1

23 2 1 0 0

24 2 2 0 1

25 2 2 0 1

26 2 1 0 0

27 2 1 0 0

28 2 1 0 0

29 2 2 0 0

30 2 1 0 0

31 2 1 0 0

32 2 2 0 0

33 2 1 0 0

34 2 2 0 1

35 2 1 0 0

36 2 1 0 0

37 2 2 0 1

38 2 2 1 0

39 2 2 0 1

40 2 2 0 0

41 2 2 0 1

42 2 1 0 0

43 2 2 0 0

44 2 2 0 0

45 2 2 0 1

46 2 2 0 0

47 2 1 0 0

48 2 2 0 0

49 2 2 0 0

50 2 2 1 1

51 2 2 0 1

52 2 2 0 0

53 2 2 1 1

54 2 2 0 0

55 2 2 0 1

56 2 2 0 1

57 2 2 0 0

58 2 1 0 0

[ reached 'max' / getOption("max.print") -- omitted 5862 rows ]

> 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 : 3606"

> print(paste("사망여부 예측 정확도 : " ,nrow(predictCorrect)/nrow(covid\_test))); # 60.9%

[1] "사망여부 예측 정확도 : 0.609121621621622"

> logisticFit\_dead = glm(day\_cnt~., data =covid\_dead\_train, family = gaussian );

> summary(logisticFit\_dead);

Call:

glm(formula = day\_cnt ~ ., family = gaussian, data = covid\_dead\_train)

Deviance Residuals:

Min 1Q Median 3Q Max

-11.840 -5.118 -1.803 3.269 43.002

Coefficients: (2 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 6.789655 2.815601 2.411 0.01594 \*

sex NA NA NA NA

patient\_type NA NA NA NA

pneumonia -1.146756 0.420694 -2.726 0.00644 \*\*

age -0.005432 0.008215 -0.661 0.50855

diabetes 0.371892 0.264079 1.408 0.15914

copd 0.748439 0.510778 1.465 0.14293

asthma 0.089652 0.765361 0.117 0.90676

inmsupr 1.076194 0.672527 1.600 0.10964

hypertension 0.197530 0.272466 0.725 0.46852

other\_disease -1.026140 0.606404 -1.692 0.09070 .

cardiovascular 0.734718 0.490061 1.499 0.13390

obesity -0.024487 0.273988 -0.089 0.92879

renal\_chronic 1.587909 0.500762 3.171 0.00153 \*\*

tobacco -0.244416 0.633829 -0.386 0.69980

contact\_other\_covid -0.477998 0.292047 -1.637 0.10178

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 55.09196)

Null deviance: 199711 on 3599 degrees of freedom

Residual deviance: 197560 on 3586 degrees of freedom

AIC: 24665

Number of Fisher Scoring iterations: 2

> prediction\_dead = predict(logisticFit\_dead, newdata=covid\_dead\_test, type = "response");

Warning message:

In predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :

prediction from a rank-deficient fit may be misleading

> prediction\_dead;

1 2 3 4 5 6 7 8 9 10 11 12 13

11.284051 11.340123 11.467814 10.808724 10.992647 11.405304 10.795189 10.025022 10.938401 10.989958 9.965371 9.803143 10.380684

14 15 16 17 18 19 20 21 22 23 24 25 26

9.970803 10.486725 11.655165 11.242351 10.524705 11.177944 10.201559 10.919346 11.136616 10.285707 9.636161 11.785531 10.969333

27 28 29 30 31 32 33 34 35 36 37 38 39

11.774667 8.052332 11.047036 11.709483 10.763706 9.512237 11.348315 11.055139 11.017663 7.362656 8.325906 11.061037 11.022549

40 41 42 43 44 45 46 47 48 49 50 51 52

11.522274 9.949076 11.676896 11.709483 9.264486 12.062550 10.881323 10.497494 11.147268 10.508397 8.155484 12.521112 11.592744

53 54 55 56 57 58 59 60 61 62 63 64 65

9.558993 11.378145 11.402632 10.562727 9.396617 7.837416 11.481349 9.326778 10.819588 9.762500 9.732036 10.759838 9.637570

66 67 68 69 70 71 72 73 74 75 76 77 78

9.542700 10.125466 10.797861 11.394441 10.293828 11.378145 11.915893 11.266839 11.669168 10.307434 11.353747 11.372714 10.136516

79 80 81 82 83 84 85 86 87 88 89 90 91

11.291237 11.589984 10.575388 9.900826 11.563603 11.416168 11.823537 9.675592 11.033412 11.456950 11.332146 11.470485 9.041129

92 93 94 95 96 97 98 99 100 101 102 103 104

9.139338 8.658336 10.486631 11.964779 9.302046 12.089709 11.420431 10.258360 8.986128 10.887529 10.380899 10.307434 11.186279

105 106 107 108 109 110 111 112 113 114 115 116 117

10.404179 10.260673 11.402776 11.232405 9.111011 10.951936 10.892187 11.039390 11.088276 11.038844 10.884769 7.643194 10.750618

118 119 120 121 122 123 124 125 126 127 128 129 130

11.698619 11.044276 9.513773 10.816142 11.693187 11.245797 11.069436 11.158889 8.882289 11.484109 9.995290 7.593897 11.530235

131 132 133 134 135 136 137 138 139 140 141 142 143

10.700089 10.399774 11.126298 9.984975 11.323828 11.125752 8.689633 11.378145 10.819588 9.663319 10.500447 10.353558 10.781565

144 145 146 147 148 149 150 151 152 153 154 155 156

8.591434 10.285707 12.043257 11.704051 11.115435 10.389592 11.333265 10.176338 11.033412 11.524803 10.875891 10.470387 11.031660

157 158 159 160 161 162 163 164 165 166 167 168 169

9.423149 11.636185 11.953915 9.902764 9.095884 8.619947 10.481199 11.534485 11.372714 11.530235 8.473341 10.139188 10.683794

170 171 172 173 174 175 176 177 178 179 180 181 182

10.502978 10.727465 11.845280 10.508358 10.488480 11.818105 11.198897 11.818895 10.989958 10.830451 11.704051 11.356418 12.730191

183 184 185 186 187 188 189 190 191 192 193 194 195

8.014310 10.002225 9.706773 10.033313 10.951936 10.803292 10.621701 11.377029 10.808724 9.699991 8.714093 11.589984 11.370727

196 197 198 199 200 201 202 203 204 205 206 207 208

11.164321 10.160777 10.348165 9.472087 11.403305 9.163825 10.819588 10.266104 10.488480 11.682324 9.155634 10.973663 11.405304

209 210 211 212 213 214 215 216 217 218 219 220 221

11.587312 11.983834 12.198652 10.962800 11.481349 12.220059 11.709483 12.298863 8.411644 9.456367 10.814156 9.630183 9.209951

222 223 224 225 226 227 228 229 230 231 232 233 234

10.849506 11.709483 10.348127 9.635602 8.131413 10.008727 10.919346 9.280126 11.595416 11.085604 11.565585 10.662067 8.783850

235 236 237 238 239 240 241 242 243 244 245 246 247

11.061117 11.682311 10.672930 12.268719 11.367282 11.247783 10.231390 11.496527 11.093708 10.293263 12.262344 11.780099 11.492212

248 249 250 251 252 253 254 255 256 257 258 259 260

9.554371 9.669620 11.494972 11.877871 11.109457 11.818121 9.951836 9.553563 9.938213 11.356418 11.708353 11.246028 10.149509

261 262 263 264 265 266 267 268 269 270 271 272 273

11.367282 11.367282 11.560153 11.262324 10.231390 10.834130 10.716384 9.925225 11.098594 11.511411 11.164321 10.743543 11.318396

274 275 276 277 278 279 280 281 282 283 284 285 286

10.804398 11.733957 11.272956 10.932897 11.180616 10.825019 10.350889 11.318396 11.361850 10.155345 10.182504 10.101028 10.426881

287 288 289 290 291 292 293 294 295 296 297 298 299

11.953915 10.323730 11.894166 10.258549 11.683101 10.416070 9.919059 11.280374 11.627994 10.022997 9.946406 10.405206 11.093162

300 301 302 303 304 305 306 307 308 309 310 311 312

11.093162 10.908482 11.106785 11.088276 11.410736 8.041469 10.759838 11.364610 7.925085 11.840622 10.781565 11.418928 10.806052

313 314 315 316 317 318 319 320 321 322 323 324 325

10.973663 11.500404 10.811484 11.226830 11.861575 11.394441 11.088276 11.411653 11.685773 11.953915 11.183376 11.365527 10.166209

326 327 328 329 330 331 332 333 334 335 336 337 338

11.511957 11.440655 10.700089 10.312815 10.562714 9.799749 11.427031 11.967539 11.818121 10.296571 11.481349 11.033412 9.997962

339 340 341 342 343 344 345 346 347 348 349 350 351

11.126298 11.120867 10.323730 10.792429 10.231390 10.139050 10.838643 9.638921 10.540224 10.755602 9.911618 9.960576 11.216110

352 353 354 355 356 357 358 359 360 361 362 363 364

11.435223 9.716286 9.173520 11.780099 11.231488 10.797861 10.968231 11.867007 10.797861 11.470485 10.274844 10.225958 10.941073

365 366 367 368 369 370 371 372 373 374 375 376 377

11.487012 11.356418 10.700089 11.368199 11.497644 10.727248 9.296859 9.924679 9.501704 10.956526 10.892187 11.435223 10.881323

378 379 380 381 382 383 384 385 386 387 388 389 390

10.198799 10.361752 10.405206 11.269725 11.943052 11.194382 11.323828 11.573689 11.256661 11.099140 11.262092 10.710952 11.839848

391 392 393 394 395 396 397 398 399 400 401 402 403

10.546419 11.312964 10.876665 11.093162 11.224301 10.160777 10.808724 11.880631 10.892961 11.481349 7.588330 11.087730 10.164454

404 405 406 407 408 409 410 411 412 413 414 415 416

11.232262 10.122755 10.898392 10.090164 10.681050 10.807160 12.017928 11.360095 10.857610 11.061117 10.177072 11.457093 9.989858

417 418 419 420

9.871712 9.307723 10.887529 12.002801

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

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

> comparison\_dead=as.data.frame(comparison\_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) : 217"

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

[1] "투병일수 예측 정확도( 5 일) : 0.516666666666667"