FDS homework 3

Sadiya Amreen- 2079690

2022-10-20

**Problem 1 (15 points):**

**For this problem, you will perform a straightforward training and evaluation of a decision tree, as well as #generate rules by hand. Load the breast\_cancer\_updated.csv data. These data are visual features computed #from samples of breast tissue being evaluated for cancer1.**

# Getting the current directory

print(getwd())

## [1] "C:/Sadiya Studies/Data Science/DS441-Fundamts DS/homework"

#Setting directory  
setwd("C:/Sadiya Studies/Data Science/DS441-Fundamts DS/homework")  
#Reading Data from the .CSV file  
cncrdata<- read.csv("breast\_cancer\_updated.csv")  
print(cncrdata)

## IDNumber ClumpThickness UniformCellSize UniformCellShape MarginalAdhesion  
## 1 1000025 5 1 1 1  
## 2 1002945 5 4 4 5  
## 3 1015425 3 1 1 1  
## 4 1016277 6 8 8 1  
## 5 1017023 4 1 1 3  
## 6 1017122 8 10 10 8  
## 7 1018099 1 1 1 1  
## 8 1018561 2 1 2 1  
## 9 1033078 2 1 1 1  
## 10 1033078 4 2 1 1  
## 11 1035283 1 1 1 1  
## 12 1036172 2 1 1 1  
## 13 1041801 5 3 3 3  
## 14 1043999 1 1 1 1  
## 15 1044572 8 7 5 10  
## 16 1047630 7 4 6 4  
## 17 1048672 4 1 1 1  
## 18 1049815 4 1 1 1  
## 19 1050670 10 7 7 6  
## 20 1050718 6 1 1 1  
## 21 1054590 7 3 2 10  
## 22 1054593 10 5 5 3  
## 23 1056784 3 1 1 1  
## 24 1057013 8 4 5 1  
## 25 1059552 1 1 1 1  
## 26 1065726 5 2 3 4  
## 27 1066373 3 2 1 1  
## 28 1066979 5 1 1 1  
## 29 1067444 2 1 1 1  
## 30 1070935 1 1 3 1  
## 31 1070935 3 1 1 1  
## 32 1071760 2 1 1 1  
## 33 1072179 10 7 7 3  
## 34 1074610 2 1 1 2  
## 35 1075123 3 1 2 1  
## 36 1079304 2 1 1 1  
## 37 1080185 10 10 10 8  
## 38 1081791 6 2 1 1  
## 39 1084584 5 4 4 9  
## 40 1091262 2 5 3 3  
## 41 1096800 6 6 6 9  
## 42 1099510 10 4 3 1  
## 43 1100524 6 10 10 2  
## 44 1102573 5 6 5 6  
## 45 1103608 10 10 10 4  
## 46 1103722 1 1 1 1  
## 47 1105257 3 7 7 4  
## 48 1105524 1 1 1 1  
## 49 1106095 4 1 1 3  
## 50 1106829 7 8 7 2  
## 51 1108370 9 5 8 1  
## 52 1108449 5 3 3 4  
## 53 1110102 10 3 6 2  
## 54 1110503 5 5 5 8  
## 55 1110524 10 5 5 6  
## 56 1111249 10 6 6 3  
## 57 1112209 8 10 10 1  
## 58 1113038 8 2 4 1  
## 59 1113483 5 2 3 1  
## 60 1113906 9 5 5 2  
## 61 1115282 5 3 5 5  
## 62 1115293 1 1 1 1  
## 63 1116116 9 10 10 1  
## 64 1116132 6 3 4 1  
## 65 1116192 1 1 1 1  
## 66 1116998 10 4 2 1  
## 67 1117152 4 1 1 1  
## 68 1118039 5 3 4 1  
## 69 1120559 8 3 8 3  
## 70 1121732 1 1 1 1  
## 71 1121919 5 1 3 1  
## 72 1123061 6 10 2 8  
## 73 1124651 1 3 3 2  
## 74 1125035 9 4 5 10  
## 75 1126417 10 6 4 1  
## 76 1131294 1 1 2 1  
## 77 1132347 1 1 4 1  
## 78 1133041 5 3 1 2  
## 79 1133136 3 1 1 1  
## 80 1136142 2 1 1 1  
## 81 1137156 2 2 2 1  
## 82 1143978 4 1 1 2  
## 83 1143978 5 2 1 1  
## 84 1147044 3 1 1 1  
## 85 1147699 3 5 7 8  
## 86 1147748 5 10 6 1  
## 87 1148278 3 3 6 4  
## 88 1148873 3 6 6 6  
## 89 1152331 4 1 1 1  
## 90 1155546 2 1 1 2  
## 91 1156272 1 1 1 1  
## 92 1156948 3 1 1 2  
## 93 1157734 4 1 1 1  
## 94 1158247 1 1 1 1  
## 95 1160476 2 1 1 1  
## 96 1164066 1 1 1 1  
## 97 1165297 2 1 1 2  
## 98 1165790 5 1 1 1  
## 99 1165926 9 6 9 2  
## 100 1166630 7 5 6 10  
## 101 1166654 10 3 5 1  
## 102 1167439 2 3 4 4  
## 103 1167471 4 1 2 1  
## 104 1168359 8 2 3 1  
## 105 1168736 10 10 10 10  
## 106 1169049 7 3 4 4  
## 107 1170419 10 10 10 8  
## 108 1170420 1 6 8 10  
## 109 1171710 1 1 1 1  
## 110 1171710 6 5 4 4  
## 111 1171795 1 3 1 2  
## 112 1171845 8 6 4 3  
## 113 1172152 10 3 3 10  
## 114 1173216 10 10 10 3  
## 115 1173235 3 3 2 1  
## 116 1173347 1 1 1 1  
## 117 1173347 8 3 3 1  
## 118 1173509 4 5 5 10  
## 119 1173514 1 1 1 1  
## 120 1173681 3 2 1 1  
## 121 1174057 1 1 2 2  
## 122 1174057 4 2 1 1  
## 123 1174131 10 10 10 2  
## 124 1174428 5 3 5 1  
## 125 1175937 5 4 6 7  
## 126 1176406 1 1 1 1  
## 127 1176881 7 5 3 7  
## 128 1177027 3 1 1 1  
## 129 1177399 8 3 5 4  
## 130 1177512 1 1 1 1  
## 131 1178580 5 1 3 1  
## 132 1179818 2 1 1 1  
## 133 1180194 5 10 8 10  
## 134 1180523 3 1 1 1  
## 135 1180831 3 1 1 1  
## 136 1181356 5 1 1 1  
## 137 1182404 4 1 1 1  
## 138 1182410 3 1 1 1  
## 139 1183240 4 1 2 1  
## 140 1183246 1 1 1 1  
## 141 1183516 3 1 1 1  
## 142 1183911 2 1 1 1  
## 143 1183983 9 5 5 4  
## 144 1184184 1 1 1 1  
## 145 1184241 2 1 1 1  
## 146 1184840 1 1 3 1  
## 147 1185609 3 4 5 2  
## 148 1185610 1 1 1 1  
## 149 1187457 3 1 1 3  
## 150 1187805 8 8 7 4  
## 151 1188472 1 1 1 1  
## 152 1189266 7 2 4 1  
## 153 1189286 10 10 8 6  
## 154 1190394 4 1 1 1  
## 155 1190485 1 1 1 1  
## 156 1192325 5 5 5 6  
## 157 1193091 1 2 2 1  
## 158 1193210 2 1 1 1  
## 159 1193683 1 1 2 1  
## 160 1196295 9 9 10 3  
## 161 1196915 10 7 7 4  
## 162 1197080 4 1 1 1  
## 163 1197270 3 1 1 1  
## 164 1197440 1 1 1 2  
## 165 1197510 5 1 1 1  
## 166 1197979 4 1 1 1  
## 167 1197993 5 6 7 8  
## 168 1198128 10 8 10 10  
## 169 1198641 3 1 1 1  
## 170 1199219 1 1 1 2  
## 171 1199731 3 1 1 1  
## 172 1199983 1 1 1 1  
## 173 1200772 1 1 1 1  
## 174 1200847 6 10 10 10  
## 175 1200892 8 6 5 4  
## 176 1200952 5 8 7 7  
## 177 1201834 2 1 1 1  
## 178 1201936 5 10 10 3  
## 179 1202125 4 1 1 1  
## 180 1202812 5 3 3 3  
## 181 1203096 1 1 1 1  
## 182 1204242 1 1 1 1  
## 183 1204898 6 1 1 1  
## 184 1205138 5 8 8 8  
## 185 1205579 8 7 6 4  
## 186 1206089 2 1 1 1  
## 187 1206695 1 5 8 6  
## 188 1206841 10 5 6 10  
## 189 1207986 5 8 4 10  
## 190 1208301 1 2 3 1  
## 191 1210963 10 10 10 8  
## 192 1211202 7 5 10 10  
## 193 1212232 5 1 1 1  
## 194 1212251 1 1 1 1  
## 195 1212422 3 1 1 1  
## 196 1212422 4 1 1 1  
## 197 1213375 8 4 4 5  
## 198 1213383 5 1 1 4  
## 199 1214092 1 1 1 1  
## 200 1214556 3 1 1 1  
## 201 1214966 9 7 7 5  
## 202 1216694 10 8 8 4  
## 203 1216947 1 1 1 1  
## 204 1217051 5 1 1 1  
## 205 1217264 1 1 1 1  
## 206 1218105 5 10 10 9  
## 207 1218741 10 10 9 3  
## 208 1218860 1 1 1 1  
## 209 1218860 1 1 1 1  
## 210 1219406 5 1 1 1  
## 211 1219525 8 10 10 10  
## 212 1219859 8 10 8 8  
## 213 1220330 1 1 1 1  
## 214 1221863 10 10 10 10  
## 215 1222047 10 10 10 10  
## 216 1222936 8 7 8 7  
## 217 1223282 1 1 1 1  
## 218 1223426 1 1 1 1  
## 219 1223793 6 10 7 7  
## 220 1223967 6 1 3 1  
## 221 1224329 1 1 1 2  
## 222 1225799 10 6 4 3  
## 223 1226012 4 1 1 3  
## 224 1226612 7 5 6 3  
## 225 1227210 10 5 5 6  
## 226 1227244 1 1 1 1  
## 227 1227481 10 5 7 4  
## 228 1228152 8 9 9 5  
## 229 1228311 1 1 1 1  
## 230 1230175 10 10 10 3  
## 231 1230688 7 4 7 4  
## 232 1231387 6 8 7 5  
## 233 1231706 8 4 6 3  
## 234 1232225 10 4 5 5  
## 235 1236043 3 3 2 1  
## 236 1241232 3 1 4 1  
## 237 1241559 10 8 8 2  
## 238 1241679 9 8 8 5  
## 239 1242364 8 10 10 8  
## 240 1243256 10 4 3 2  
## 241 1270479 5 1 3 3  
## 242 1276091 3 1 1 3  
## 243 1277018 2 1 1 1  
## 244 128059 1 1 1 1  
## 245 1285531 1 1 1 1  
## 246 1287775 5 1 1 2  
## 247 144888 8 10 10 8  
## 248 145447 8 4 4 1  
## 249 167528 4 1 1 1  
## 250 169356 3 1 1 1  
## 251 183913 1 2 2 1  
## 252 191250 10 4 4 10  
## 253 1017023 6 3 3 5  
## 254 1100524 6 10 10 2  
## 255 1116116 9 10 10 1  
## 256 1168736 5 6 6 2  
## 257 1182404 3 1 1 1  
## 258 1182404 3 1 1 1  
## 259 1198641 3 1 1 1  
## 260 242970 5 7 7 1  
## 261 255644 10 5 8 10  
## 262 263538 5 10 10 6  
## 263 274137 8 8 9 4  
## 264 303213 10 4 4 10  
## 265 314428 7 9 4 10  
## 266 1182404 5 1 4 1  
## 267 1198641 10 10 6 3  
## 268 320675 3 3 5 2  
## 269 324427 10 8 8 2  
## 270 385103 1 1 1 1  
## 271 390840 8 4 7 1  
## 272 411453 5 1 1 1  
## 273 320675 3 3 5 2  
## 274 428903 7 2 4 1  
## 275 431495 3 1 1 1  
## 276 432809 3 1 3 1  
## 277 434518 3 1 1 1  
## 278 452264 1 1 1 1  
## 279 456282 1 1 1 1  
## 280 476903 10 5 7 3  
## 281 486283 3 1 1 1  
## 282 486662 2 1 1 2  
## 283 488173 1 4 3 10  
## 284 492268 10 4 6 1  
## 285 508234 7 4 5 10  
## 286 527363 8 10 10 10  
## 287 529329 10 10 10 10  
## 288 535331 3 1 1 1  
## 289 543558 6 1 3 1  
## 290 555977 5 6 6 8  
## 291 560680 1 1 1 1  
## 292 561477 1 1 1 1  
## 293 563649 8 8 8 1  
## 294 601265 10 4 4 6  
## 295 606140 1 1 1 1  
## 296 606722 5 5 7 8  
## 297 616240 5 3 4 3  
## 298 61634 5 4 3 1  
## 299 625201 8 2 1 1  
## 300 63375 9 1 2 6  
## 301 635844 8 4 10 5  
## 302 636130 1 1 1 1  
## 303 640744 10 10 10 7  
## 304 646904 1 1 1 1  
## 305 653777 8 3 4 9  
## 306 659642 10 8 4 4  
## 307 666090 1 1 1 1  
## 308 666942 1 1 1 1  
## 309 667204 7 8 7 6  
## 310 673637 3 1 1 1  
## 311 684955 2 1 1 1  
## 312 688033 1 1 1 1  
## 313 691628 8 6 4 10  
## 314 693702 1 1 1 1  
## 315 704097 1 1 1 1  
## 316 704168 4 6 5 6  
## 317 706426 5 5 5 2  
## 318 709287 6 8 7 8  
## 319 718641 1 1 1 1  
## 320 721482 4 4 4 4  
## 321 730881 7 6 3 2  
## 322 733639 3 1 1 1  
## 323 733639 3 1 1 1  
## 324 733823 5 4 6 10  
## 325 740492 1 1 1 1  
## 326 743348 3 2 2 1  
## 327 752904 10 1 1 1  
## 328 756136 1 1 1 1  
## 329 760001 8 10 3 2  
## 330 760239 10 4 6 4  
## 331 76389 10 4 7 2  
## 332 764974 5 1 1 1  
## 333 770066 5 2 2 2  
## 334 785208 5 4 6 6  
## 335 785615 8 6 7 3  
## 336 792744 1 1 1 1  
## 337 797327 6 5 5 8  
## 338 798429 1 1 1 1  
## 339 704097 1 1 1 1  
## 340 806423 8 5 5 5  
## 341 809912 10 3 3 1  
## 342 810104 1 1 1 1  
## 343 814265 2 1 1 1  
## 344 814911 1 1 1 1  
## 345 822829 7 6 4 8  
## 346 826923 1 1 1 1  
## 347 830690 5 2 2 2  
## 348 831268 1 1 1 1  
## 349 832226 3 4 4 10  
## 350 832567 4 2 3 5  
## 351 836433 5 1 1 3  
## 352 837082 2 1 1 1  
## 353 846832 3 4 5 3  
## 354 850831 2 7 10 10  
## 355 855524 1 1 1 1  
## 356 857774 4 1 1 1  
## 357 859164 5 3 3 1  
## 358 859350 8 10 10 7  
## 359 866325 8 10 5 3  
## 360 873549 10 3 5 4  
## 361 877291 6 10 10 10  
## 362 877943 3 10 3 10  
## 363 888169 3 2 2 1  
## 364 888523 4 4 4 2  
## 365 896404 2 1 1 1  
## 366 897172 2 1 1 1  
## 367 95719 6 10 10 10  
## 368 160296 5 8 8 10  
## 369 342245 1 1 3 1  
## 370 428598 1 1 3 1  
## 371 492561 4 3 2 1  
## 372 493452 1 1 3 1  
## 373 493452 4 1 2 1  
## 374 521441 5 1 1 2  
## 375 560680 3 1 2 1  
## 376 636437 1 1 1 1  
## 377 640712 1 1 1 1  
## 378 654244 1 1 1 1  
## 379 657753 3 1 1 4  
## 380 685977 5 3 4 1  
## 381 805448 1 1 1 1  
## 382 846423 10 6 3 6  
## 383 1002504 3 2 2 2  
## 384 1022257 2 1 1 1  
## 385 1026122 2 1 1 1  
## 386 1071084 3 3 2 2  
## 387 1080233 7 6 6 3  
## 388 1114570 5 3 3 2  
## 389 1114570 2 1 1 1  
## 390 1116715 5 1 1 1  
## 391 1131411 1 1 1 2  
## 392 1151734 10 8 7 4  
## 393 1156017 3 1 1 1  
## 394 1158247 1 1 1 1  
## 395 1158405 1 2 3 1  
## 396 1168278 3 1 1 1  
## 397 1176187 3 1 1 1  
## 398 1196263 4 1 1 1  
## 399 1196475 3 2 1 1  
## 400 1206314 1 2 3 1  
## 401 1211265 3 10 8 7  
## 402 1213784 3 1 1 1  
## 403 1223003 5 3 3 1  
## 404 1223306 3 1 1 1  
## 405 1223543 1 2 1 3  
## 406 1229929 1 1 1 1  
## 407 1231853 4 2 2 1  
## 408 1234554 1 1 1 1  
## 409 1236837 2 3 2 2  
## 410 1237674 3 1 2 1  
## 411 1238021 1 1 1 1  
## 412 1238464 1 1 1 1  
## 413 1238633 10 10 10 6  
## 414 1238915 5 1 2 1  
## 415 1238948 8 5 6 2  
## 416 1239232 3 3 2 6  
## 417 1239347 8 7 8 5  
## 418 1239967 1 1 1 1  
## 419 1240337 5 2 2 2  
## 420 1253505 2 3 1 1  
## 421 1255384 3 2 2 3  
## 422 1257200 10 10 10 7  
## 423 1257648 4 3 3 1  
## 424 1257815 5 1 3 1  
## 425 1257938 3 1 1 1  
## 426 1258549 9 10 10 10  
## 427 1258556 5 3 6 1  
## 428 1266154 8 7 8 2  
## 429 1272039 1 1 1 1  
## 430 1276091 2 1 1 1  
## 431 1276091 1 3 1 1  
## 432 1276091 5 1 1 3  
## 433 1277629 5 1 1 1  
## 434 1293439 3 2 2 3  
## 435 1293439 6 9 7 5  
## 436 1294562 10 8 10 1  
## 437 1295186 10 10 10 1  
## 438 527337 4 1 1 1  
## 439 558538 4 1 3 3  
## 440 566509 5 1 1 1  
## 441 608157 10 4 3 10  
## 442 677910 5 2 2 4  
## 443 734111 1 1 1 3  
## 444 734111 1 1 1 1  
## 445 780555 5 1 1 6  
## 446 827627 2 1 1 1  
## 447 1049837 1 1 1 1  
## 448 1058849 5 1 1 1  
## 449 1182404 1 1 1 1  
## 450 1193544 5 7 9 8  
## 451 1201870 4 1 1 3  
## 452 1202253 5 1 1 1  
## 453 1227081 3 1 1 3  
## 454 1230994 4 5 5 8  
## 455 1238410 2 3 1 1  
## 456 1246562 10 2 2 1  
## 457 1257470 10 6 5 8  
## 458 1259008 8 8 9 6  
## 459 1266124 5 1 2 1  
## 460 1267898 5 1 3 1  
## 461 1268313 5 1 1 3  
## 462 1268804 3 1 1 1  
## 463 1276091 6 1 1 3  
## 464 1280258 4 1 1 1  
## 465 1293966 4 1 1 1  
## 466 1296572 10 9 8 7  
## 467 1298416 10 6 6 2  
## 468 1299596 6 6 6 5  
## 469 1105524 4 1 1 1  
## 470 1181685 1 1 2 1  
## 471 1211594 3 1 1 1  
## 472 1238777 6 1 1 3  
## 473 1257608 6 1 1 1  
## 474 1269574 4 1 1 1  
## 475 1277145 5 1 1 1  
## 476 1287282 3 1 1 1  
## 477 1296025 4 1 2 1  
## 478 1296263 4 1 1 1  
## 479 1296593 5 2 1 1  
## 480 1299161 4 8 7 10  
## 481 1301945 5 1 1 1  
## 482 1302428 5 3 2 4  
## 483 1318169 9 10 10 10  
## 484 474162 8 7 8 5  
## 485 787451 5 1 2 1  
## 486 1002025 1 1 1 3  
## 487 1070522 3 1 1 1  
## 488 1073960 10 10 10 10  
## 489 1076352 3 6 4 10  
## 490 1084139 6 3 2 1  
## 491 1115293 1 1 1 1  
## 492 1119189 5 8 9 4  
## 493 1133991 4 1 1 1  
## 494 1142706 5 10 10 10  
## 495 1155967 5 1 2 10  
## 496 1170945 3 1 1 1  
## 497 1181567 1 1 1 1  
## 498 1182404 4 2 1 1  
## 499 1204558 4 1 1 1  
## 500 1217952 4 1 1 1  
## 501 1224565 6 1 1 1  
## 502 1238186 4 1 1 1  
## 503 1253917 4 1 1 2  
## 504 1265899 4 1 1 1  
## 505 1268766 1 1 1 1  
## 506 1277268 3 3 1 1  
## 507 1286943 8 10 10 10  
## 508 1295508 1 1 1 1  
## 509 1297327 5 1 1 1  
## 510 1297522 2 1 1 1  
## 511 1298360 1 1 1 1  
## 512 1299924 5 1 1 1  
## 513 1299994 5 1 1 1  
## 514 1304595 3 1 1 1  
## 515 1306282 6 6 7 10  
## 516 1313325 4 10 4 7  
## 517 1320077 1 1 1 1  
## 518 1320077 1 1 1 1  
## 519 1320304 3 1 2 2  
## 520 1330439 4 7 8 3  
## 521 333093 1 1 1 1  
## 522 369565 4 1 1 1  
## 523 412300 10 4 5 4  
## 524 672113 7 5 6 10  
## 525 749653 3 1 1 1  
## 526 769612 3 1 1 2  
## 527 769612 4 1 1 1  
## 528 798429 4 1 1 1  
## 529 807657 6 1 3 2  
## 530 8233704 4 1 1 1  
## 531 837480 7 4 4 3  
## 532 867392 4 2 2 1  
## 533 869828 1 1 1 1  
## 534 1043068 3 1 1 1  
## 535 1056171 2 1 1 1  
## 536 1061990 1 1 3 2  
## 537 1113061 5 1 1 1  
## 538 1116192 5 1 2 1  
## 539 1135090 4 1 1 1  
## 540 1145420 6 1 1 1  
## 541 1158157 5 1 1 1  
## 542 1171578 3 1 1 1  
## 543 1174841 5 3 1 1  
## 544 1184586 4 1 1 1  
## 545 1186936 2 1 3 2  
## 546 1197527 5 1 1 1  
## 547 1222464 6 10 10 10  
## 548 1240603 2 1 1 1  
## 549 1240603 3 1 1 1  
## 550 1241035 7 8 3 7  
## 551 1287971 3 1 1 1  
## 552 1289391 1 1 1 1  
## 553 1299924 3 2 2 2  
## 554 1306339 4 4 2 1  
## 555 1313658 3 1 1 1  
## 556 1313982 4 3 1 1  
## 557 1321264 5 2 2 2  
## 558 1321321 5 1 1 3  
## 559 1321348 2 1 1 1  
## 560 1321931 5 1 1 1  
## 561 1321942 5 1 1 1  
## 562 1321942 5 1 1 1  
## 563 1328331 1 1 1 1  
## 564 1328755 3 1 1 1  
## 565 1331405 4 1 1 1  
## 566 1331412 5 7 10 10  
## 567 1333104 3 1 2 1  
## 568 1334071 4 1 1 1  
## 569 1343068 8 4 4 1  
## 570 1343374 10 10 8 10  
## 571 1344121 8 10 4 4  
## 572 142932 7 6 10 5  
## 573 183936 3 1 1 1  
## 574 324382 1 1 1 1  
## 575 378275 10 9 7 3  
## 576 385103 5 1 2 1  
## 577 690557 5 1 1 1  
## 578 695091 1 1 1 1  
## 579 695219 1 1 1 1  
## 580 824249 1 1 1 1  
## 581 871549 5 1 2 1  
## 582 878358 5 7 10 6  
## 583 1107684 6 10 5 5  
## 584 1115762 3 1 1 1  
## 585 1217717 5 1 1 6  
## 586 1239420 1 1 1 1  
## 587 1254538 8 10 10 10  
## 588 1261751 5 1 1 1  
## 589 1268275 9 8 8 9  
## 590 1272166 5 1 1 1  
## 591 1294261 4 10 8 5  
## 592 1295529 2 5 7 6  
## 593 1298484 10 3 4 5  
## 594 1311875 5 1 2 1  
## 595 1315506 4 8 6 3  
## 596 1320141 5 1 1 1  
## 597 1325309 4 1 2 1  
## 598 1333063 5 1 3 1  
## 599 1333495 3 1 1 1  
## 600 1334659 5 2 4 1  
## 601 1336798 3 1 1 1  
## 602 1344449 1 1 1 1  
## 603 1350568 4 1 1 1  
## 604 1352663 5 4 6 8  
## 605 188336 5 3 2 8  
## 606 352431 10 5 10 3  
## 607 353098 4 1 1 2  
## 608 411453 1 1 1 1  
## 609 557583 5 10 10 10  
## 610 636375 5 1 1 1  
## 611 736150 10 4 3 10  
## 612 803531 5 10 10 10  
## 613 822829 8 10 10 10  
## 614 1016634 2 3 1 1  
## 615 1031608 2 1 1 1  
## 616 1041043 4 1 3 1  
## 617 1042252 3 1 1 1  
## 618 1057067 1 1 1 1  
## 619 1061990 4 1 1 1  
## 620 1073836 5 1 1 1  
## 621 1083817 3 1 1 1  
## 622 1096352 6 3 3 3  
## 623 1140597 7 1 2 3  
## 624 1149548 1 1 1 1  
## 625 1174009 5 1 1 2  
## 626 1183596 3 1 3 1  
## 627 1190386 4 6 6 5  
## 628 1190546 2 1 1 1  
## 629 1213273 2 1 1 1  
## 630 1218982 4 1 1 1  
## 631 1225382 6 2 3 1  
## 632 1235807 5 1 1 1  
## 633 1238777 1 1 1 1  
## 634 1253955 8 7 4 4  
## 635 1257366 3 1 1 1  
## 636 1260659 3 1 4 1  
## 637 1268952 10 10 7 8  
## 638 1275807 4 2 4 3  
## 639 1277792 4 1 1 1  
## 640 1277792 5 1 1 3  
## 641 1285722 4 1 1 3  
## 642 1288608 3 1 1 1  
## 643 1290203 3 1 1 1  
## 644 1294413 1 1 1 1  
## 645 1299596 2 1 1 1  
## 646 1303489 3 1 1 1  
## 647 1311033 1 2 2 1  
## 648 1311108 1 1 1 3  
## 649 1315807 5 10 10 10  
## 650 1318671 3 1 1 1  
## 651 1319609 3 1 1 2  
## 652 1323477 1 2 1 3  
## 653 1324572 5 1 1 1  
## 654 1324681 4 1 1 1  
## 655 1325159 3 1 1 1  
## 656 1326892 3 1 1 1  
## 657 1330361 5 1 1 1  
## 658 1333877 5 4 5 1  
## 659 1334015 7 8 8 7  
## 660 1334667 1 1 1 1  
## 661 1339781 1 1 1 1  
## 662 1339781 4 1 1 1  
## 663 13454352 1 1 3 1  
## 664 1345452 1 1 3 1  
## 665 1345593 3 1 1 3  
## 666 1347749 1 1 1 1  
## 667 1347943 5 2 2 2  
## 668 1348851 3 1 1 1  
## 669 1350319 5 7 4 1  
## 670 1350423 5 10 10 8  
## 671 1352848 3 10 7 8  
## 672 1353092 3 2 1 2  
## 673 1354840 2 1 1 1  
## 674 1354840 5 3 2 1  
## 675 1355260 1 1 1 1  
## 676 1365075 4 1 4 1  
## 677 1365328 1 1 2 1  
## 678 1368267 5 1 1 1  
## 679 1368273 1 1 1 1  
## 680 1368882 2 1 1 1  
## 681 1369821 10 10 10 10  
## 682 1371026 5 10 10 10  
## 683 1371920 5 1 1 1  
## 684 466906 1 1 1 1  
## 685 466906 1 1 1 1  
## 686 534555 1 1 1 1  
## 687 536708 1 1 1 1  
## 688 566346 3 1 1 1  
## 689 603148 4 1 1 1  
## 690 654546 1 1 1 1  
## 691 654546 1 1 1 3  
## 692 695091 5 10 10 5  
## 693 714039 3 1 1 1  
## 694 763235 3 1 1 1  
## 695 776715 3 1 1 1  
## 696 841769 2 1 1 1  
## 697 888820 5 10 10 3  
## 698 897471 4 8 6 4  
## 699 897471 4 8 8 5  
## EpithelialCellSize BareNuclei BlandChromatin NormalNucleoli Mitoses  
## 1 2 1 3 1 1  
## 2 7 10 3 2 1  
## 3 2 2 3 1 1  
## 4 3 4 3 7 1  
## 5 2 1 3 1 1  
## 6 7 10 9 7 1  
## 7 2 10 3 1 1  
## 8 2 1 3 1 1  
## 9 2 1 1 1 5  
## 10 2 1 2 1 1  
## 11 1 1 3 1 1  
## 12 2 1 2 1 1  
## 13 2 3 4 4 1  
## 14 2 3 3 1 1  
## 15 7 9 5 5 4  
## 16 6 1 4 3 1  
## 17 2 1 2 1 1  
## 18 2 1 3 1 1  
## 19 4 10 4 1 2  
## 20 2 1 3 1 1  
## 21 5 10 5 4 4  
## 22 6 7 7 10 1  
## 23 2 1 2 1 1  
## 24 2 NA 7 3 1  
## 25 2 1 3 1 1  
## 26 2 7 3 6 1  
## 27 1 1 2 1 1  
## 28 2 1 2 1 1  
## 29 2 1 2 1 1  
## 30 2 1 1 1 1  
## 31 1 1 2 1 1  
## 32 2 1 3 1 1  
## 33 8 5 7 4 3  
## 34 2 1 3 1 1  
## 35 2 1 2 1 1  
## 36 2 1 2 1 1  
## 37 6 1 8 9 1  
## 38 1 1 7 1 1  
## 39 2 10 5 6 1  
## 40 6 7 7 5 1  
## 41 6 NA 7 8 1  
## 42 3 3 6 5 2  
## 43 8 10 7 3 3  
## 44 10 1 3 1 1  
## 45 8 1 8 10 1  
## 46 2 1 2 1 2  
## 47 4 9 4 8 1  
## 48 2 1 2 1 1  
## 49 2 1 3 1 1  
## 50 4 8 3 8 2  
## 51 2 3 2 1 5  
## 52 2 4 3 4 1  
## 53 3 5 4 10 2  
## 54 10 8 7 3 7  
## 55 8 8 7 1 1  
## 56 4 5 3 6 1  
## 57 3 6 3 9 1  
## 58 5 1 5 4 4  
## 59 6 10 5 1 1  
## 60 2 2 5 1 1  
## 61 3 3 4 10 1  
## 62 2 2 2 1 1  
## 63 10 8 3 3 1  
## 64 5 2 3 9 1  
## 65 2 1 2 1 1  
## 66 3 2 4 3 10  
## 67 2 1 3 1 1  
## 68 8 10 4 9 1  
## 69 4 9 8 9 8  
## 70 2 1 3 2 1  
## 71 2 1 2 1 1  
## 72 10 2 7 8 10  
## 73 2 1 7 2 1  
## 74 6 10 4 8 1  
## 75 3 4 3 2 3  
## 76 2 2 4 2 1  
## 77 2 1 2 1 1  
## 78 2 1 2 1 1  
## 79 2 3 3 1 1  
## 80 3 1 2 1 1  
## 81 1 1 7 1 1  
## 82 2 1 2 1 1  
## 83 2 1 3 1 1  
## 84 2 2 7 1 1  
## 85 8 9 7 10 7  
## 86 10 4 4 10 10  
## 87 5 8 4 4 1  
## 88 5 10 6 8 3  
## 89 2 1 3 1 1  
## 90 3 1 2 1 1  
## 91 2 1 3 1 1  
## 92 2 1 1 1 1  
## 93 2 1 3 1 1  
## 94 2 1 2 1 1  
## 95 2 1 3 1 1  
## 96 2 1 3 1 1  
## 97 2 1 1 1 1  
## 98 2 1 3 1 1  
## 99 10 6 2 9 10  
## 100 5 10 7 9 4  
## 101 10 5 3 10 2  
## 102 2 5 2 5 1  
## 103 2 1 3 1 1  
## 104 6 3 7 1 1  
## 105 10 1 8 8 8  
## 106 3 3 3 2 7  
## 107 2 10 4 1 1  
## 108 8 10 5 7 1  
## 109 2 1 2 3 1  
## 110 3 9 7 8 3  
## 111 2 2 5 3 2  
## 112 5 9 3 1 1  
## 113 2 10 7 3 3  
## 114 10 8 8 1 1  
## 115 2 3 3 1 1  
## 116 2 5 1 1 1  
## 117 2 2 3 2 1  
## 118 4 10 7 5 8  
## 119 4 3 1 1 1  
## 120 2 2 3 1 1  
## 121 2 1 3 1 1  
## 122 2 2 3 1 1  
## 123 10 10 5 3 3  
## 124 8 10 5 3 1  
## 125 9 7 8 10 1  
## 126 2 1 2 1 1  
## 127 4 10 7 5 5  
## 128 2 1 3 1 1  
## 129 5 10 1 6 2  
## 130 10 1 1 1 1  
## 131 2 1 2 1 1  
## 132 2 1 3 1 1  
## 133 8 10 3 6 3  
## 134 2 1 2 2 1  
## 135 3 1 2 1 1  
## 136 2 2 3 3 1  
## 137 2 1 2 1 1  
## 138 2 1 1 1 1  
## 139 2 1 2 1 1  
## 140 1 NA 2 1 1  
## 141 2 1 1 1 1  
## 142 2 1 1 1 1  
## 143 4 5 4 3 3  
## 144 2 5 1 1 1  
## 145 2 1 2 1 1  
## 146 2 NA 2 1 1  
## 147 6 8 4 1 1  
## 148 3 2 2 1 1  
## 149 8 1 5 8 1  
## 150 10 10 7 8 7  
## 151 1 1 3 1 1  
## 152 6 10 5 4 3  
## 153 4 5 8 10 1  
## 154 2 3 1 1 1  
## 155 2 1 1 1 1  
## 156 3 10 3 1 1  
## 157 2 1 2 1 1  
## 158 2 1 3 1 1  
## 159 3 NA 1 1 1  
## 160 6 10 7 10 6  
## 161 5 10 5 7 2  
## 162 2 1 3 2 1  
## 163 2 1 3 1 1  
## 164 1 3 1 1 7  
## 165 2 NA 3 1 1  
## 166 2 2 3 2 1  
## 167 8 10 3 10 3  
## 168 6 1 3 1 10  
## 169 2 1 3 1 1  
## 170 1 1 1 1 1  
## 171 2 1 1 1 1  
## 172 2 1 3 1 1  
## 173 2 1 2 1 1  
## 174 8 10 10 10 7  
## 175 3 10 6 1 1  
## 176 10 10 5 7 1  
## 177 2 1 3 1 1  
## 178 8 1 5 10 3  
## 179 2 1 3 1 1  
## 180 6 10 3 1 1  
## 181 1 1 3 1 1  
## 182 2 1 1 1 1  
## 183 2 1 3 1 1  
## 184 5 10 7 8 1  
## 185 4 10 5 1 1  
## 186 1 1 3 1 1  
## 187 5 8 7 10 1  
## 188 6 10 7 7 10  
## 189 5 8 9 10 1  
## 190 2 1 3 1 1  
## 191 6 8 7 10 1  
## 192 10 10 4 10 3  
## 193 2 1 2 1 1  
## 194 2 1 3 1 1  
## 195 2 1 3 1 1  
## 196 2 1 3 1 1  
## 197 4 7 7 8 2  
## 198 2 1 3 1 1  
## 199 2 1 1 1 1  
## 200 2 1 2 1 1  
## 201 5 10 7 8 3  
## 202 10 10 8 1 1  
## 203 2 1 3 1 1  
## 204 2 1 3 1 1  
## 205 2 1 3 1 1  
## 206 6 10 7 10 5  
## 207 7 5 3 5 1  
## 208 1 1 3 1 1  
## 209 1 1 3 1 1  
## 210 1 1 3 1 1  
## 211 5 10 8 10 6  
## 212 4 8 7 7 1  
## 213 2 1 3 1 1  
## 214 7 10 7 10 4  
## 215 3 10 10 6 1  
## 216 5 5 5 10 2  
## 217 2 1 2 1 1  
## 218 2 1 3 1 1  
## 219 6 4 8 10 2  
## 220 2 1 3 1 1  
## 221 2 1 3 1 1  
## 222 10 10 9 10 1  
## 223 1 5 2 1 1  
## 224 3 8 7 4 1  
## 225 3 10 7 9 2  
## 226 2 1 2 1 1  
## 227 4 10 8 9 1  
## 228 3 5 7 7 1  
## 229 1 1 3 1 1  
## 230 10 10 9 10 1  
## 231 3 7 7 6 1  
## 232 6 8 8 9 2  
## 233 3 1 4 3 1  
## 234 5 10 4 1 1  
## 235 3 1 3 6 1  
## 236 2 NA 3 1 1  
## 237 8 10 4 8 10  
## 238 6 2 4 10 4  
## 239 6 9 3 10 10  
## 240 3 10 5 3 2  
## 241 2 2 2 3 1  
## 242 1 1 3 1 1  
## 243 2 1 3 1 1  
## 244 2 5 5 1 1  
## 245 2 1 3 1 1  
## 246 2 2 3 1 1  
## 247 5 10 7 8 1  
## 248 2 9 3 3 1  
## 249 2 1 3 6 1  
## 250 2 NA 3 1 1  
## 251 2 1 1 1 1  
## 252 2 10 5 3 3  
## 253 3 10 3 5 3  
## 254 8 10 7 3 3  
## 255 10 8 3 3 1  
## 256 4 10 3 6 1  
## 257 2 1 1 1 1  
## 258 2 1 2 1 1  
## 259 2 1 3 1 1  
## 260 5 8 3 4 1  
## 261 3 10 5 1 3  
## 262 10 10 10 6 5  
## 263 5 10 7 8 1  
## 264 6 10 5 5 1  
## 265 10 3 5 3 3  
## 266 2 1 3 2 1  
## 267 3 10 4 3 2  
## 268 3 10 7 1 1  
## 269 3 4 8 7 8  
## 270 2 1 3 1 1  
## 271 3 10 3 9 2  
## 272 2 1 3 1 1  
## 273 3 10 7 1 1  
## 274 3 4 3 3 1  
## 275 2 1 3 2 1  
## 276 2 NA 2 1 1  
## 277 2 1 2 1 1  
## 278 2 1 2 1 1  
## 279 2 1 3 1 1  
## 280 3 7 3 3 8  
## 281 2 1 3 1 1  
## 282 2 1 3 1 1  
## 283 4 10 5 6 1  
## 284 2 10 5 3 1  
## 285 2 10 3 8 2  
## 286 8 10 10 7 3  
## 287 10 10 4 10 10  
## 288 3 1 2 1 1  
## 289 4 5 5 10 1  
## 290 6 10 4 10 4  
## 291 2 1 1 1 1  
## 292 2 1 3 1 1  
## 293 2 NA 6 10 1  
## 294 2 10 2 3 1  
## 295 2 NA 2 1 1  
## 296 6 10 7 4 1  
## 297 4 5 4 7 1  
## 298 2 NA 2 3 1  
## 299 5 1 1 1 1  
## 300 4 10 7 7 2  
## 301 4 4 7 10 1  
## 302 2 1 3 1 1  
## 303 9 10 7 10 10  
## 304 2 1 3 1 1  
## 305 3 10 3 3 1  
## 306 4 10 3 10 4  
## 307 2 1 3 1 1  
## 308 2 1 3 1 1  
## 309 4 3 8 8 4  
## 310 2 5 5 1 1  
## 311 3 1 2 1 1  
## 312 2 1 1 1 1  
## 313 10 1 3 5 1  
## 314 2 1 1 1 1  
## 315 1 1 2 1 1  
## 316 7 NA 4 9 1  
## 317 5 10 4 3 1  
## 318 6 8 8 9 1  
## 319 5 1 3 1 1  
## 320 6 5 7 3 1  
## 321 5 10 7 4 6  
## 322 2 NA 3 1 1  
## 323 2 1 3 1 1  
## 324 2 10 4 1 1  
## 325 2 1 3 1 1  
## 326 2 1 2 3 1  
## 327 2 10 5 4 1  
## 328 2 1 2 1 1  
## 329 6 4 3 10 1  
## 330 5 10 7 1 1  
## 331 2 8 6 1 1  
## 332 2 1 3 1 2  
## 333 2 1 2 2 1  
## 334 4 10 4 3 1  
## 335 3 10 3 4 2  
## 336 2 1 1 1 1  
## 337 4 10 3 4 1  
## 338 2 1 3 1 1  
## 339 1 1 2 1 1  
## 340 2 10 4 3 1  
## 341 2 10 7 6 1  
## 342 2 1 3 1 1  
## 343 2 1 1 1 1  
## 344 2 1 1 1 1  
## 345 10 10 9 5 3  
## 346 2 1 1 1 1  
## 347 3 1 1 3 1  
## 348 1 1 1 3 1  
## 349 5 1 3 3 1  
## 350 3 8 7 6 1  
## 351 2 1 1 1 1  
## 352 2 1 3 1 1  
## 353 7 3 4 6 1  
## 354 7 10 4 9 4  
## 355 2 1 2 1 1  
## 356 3 1 2 2 1  
## 357 3 3 3 3 3  
## 358 10 10 7 3 8  
## 359 8 4 4 10 3  
## 360 3 7 3 5 3  
## 361 10 10 8 10 10  
## 362 6 10 5 1 4  
## 363 4 3 2 1 1  
## 364 2 3 2 1 1  
## 365 2 1 3 1 1  
## 366 2 1 2 1 1  
## 367 8 10 7 10 7  
## 368 5 10 8 10 3  
## 369 2 1 1 1 1  
## 370 1 1 2 1 1  
## 371 3 1 2 1 1  
## 372 2 1 1 1 1  
## 373 2 1 2 1 1  
## 374 2 1 2 1 1  
## 375 2 1 2 1 1  
## 376 2 1 1 1 1  
## 377 2 1 2 1 1  
## 378 1 1 2 1 1  
## 379 3 1 2 2 1  
## 380 4 1 3 1 1  
## 381 2 1 1 1 1  
## 382 4 10 7 8 4  
## 383 2 1 3 2 1  
## 384 2 1 1 1 1  
## 385 2 1 1 1 1  
## 386 3 1 1 2 3  
## 387 2 10 7 1 1  
## 388 3 1 3 1 1  
## 389 2 1 2 2 1  
## 390 3 2 2 2 1  
## 391 2 1 2 1 1  
## 392 3 10 7 9 1  
## 393 2 1 2 1 1  
## 394 1 1 1 1 1  
## 395 2 1 2 1 1  
## 396 2 1 2 1 1  
## 397 2 1 3 1 1  
## 398 2 1 1 1 1  
## 399 2 1 2 2 1  
## 400 2 1 1 1 1  
## 401 6 9 9 3 8  
## 402 2 1 1 1 1  
## 403 2 1 2 1 1  
## 404 2 4 1 1 1  
## 405 2 1 1 2 1  
## 406 2 1 2 1 1  
## 407 2 1 2 1 1  
## 408 2 1 2 1 1  
## 409 2 2 3 1 1  
## 410 2 1 2 1 1  
## 411 2 1 2 1 1  
## 412 1 NA 2 1 1  
## 413 8 4 8 5 1  
## 414 2 1 3 1 1  
## 415 3 10 6 6 1  
## 416 3 3 3 5 1  
## 417 10 10 7 2 1  
## 418 2 1 2 1 1  
## 419 2 2 3 2 2  
## 420 5 1 1 1 1  
## 421 2 3 3 1 1  
## 422 10 10 8 2 1  
## 423 2 1 3 3 1  
## 424 2 1 2 1 1  
## 425 2 1 1 1 1  
## 426 10 10 10 10 1  
## 427 2 1 1 1 1  
## 428 4 2 5 10 1  
## 429 2 1 2 1 1  
## 430 2 1 2 1 1  
## 431 2 1 2 2 1  
## 432 4 1 3 2 1  
## 433 2 1 2 2 1  
## 434 2 1 1 1 1  
## 435 5 8 4 2 1  
## 436 3 10 5 1 1  
## 437 6 1 2 8 1  
## 438 2 1 1 1 1  
## 439 2 1 1 1 1  
## 440 2 1 1 1 1  
## 441 4 10 10 1 1  
## 442 2 4 1 1 1  
## 443 2 3 1 1 1  
## 444 2 2 1 1 1  
## 445 3 1 2 1 1  
## 446 2 1 1 1 1  
## 447 2 1 1 1 1  
## 448 2 1 1 1 1  
## 449 1 1 1 1 1  
## 450 6 10 8 10 1  
## 451 1 1 2 1 1  
## 452 2 1 1 1 1  
## 453 2 1 1 1 1  
## 454 6 10 10 7 1  
## 455 3 1 1 1 1  
## 456 2 6 1 1 2  
## 457 5 10 8 6 1  
## 458 6 3 10 10 1  
## 459 2 1 1 1 1  
## 460 2 1 1 1 1  
## 461 2 1 1 1 1  
## 462 2 5 1 1 1  
## 463 2 1 1 1 1  
## 464 2 1 1 2 1  
## 465 2 1 1 1 1  
## 466 6 4 7 10 3  
## 467 4 10 9 7 1  
## 468 4 10 7 6 2  
## 469 2 1 1 1 1  
## 470 2 1 2 1 1  
## 471 1 1 2 1 1  
## 472 2 1 1 1 1  
## 473 1 1 1 1 1  
## 474 2 1 1 1 1  
## 475 2 1 1 1 1  
## 476 2 1 1 1 1  
## 477 2 1 1 1 1  
## 478 2 1 1 1 1  
## 479 2 1 1 1 1  
## 480 4 10 7 5 1  
## 481 1 1 1 1 1  
## 482 2 1 1 1 1  
## 483 10 5 10 10 10  
## 484 5 10 9 10 1  
## 485 2 1 1 1 1  
## 486 1 3 1 1 1  
## 487 1 1 2 1 1  
## 488 6 10 8 1 5  
## 489 3 3 3 4 1  
## 490 3 4 4 1 1  
## 491 2 1 1 1 1  
## 492 3 10 7 1 1  
## 493 1 1 2 1 1  
## 494 6 10 6 5 2  
## 495 4 5 2 1 1  
## 496 1 1 2 1 1  
## 497 1 1 1 1 1  
## 498 2 1 1 1 1  
## 499 2 1 2 1 1  
## 500 2 1 2 1 1  
## 501 2 1 3 1 1  
## 502 2 1 2 1 1  
## 503 2 1 2 1 1  
## 504 2 1 3 1 1  
## 505 2 1 1 1 1  
## 506 2 1 1 1 1  
## 507 7 5 4 8 7  
## 508 2 4 1 1 1  
## 509 2 1 1 1 1  
## 510 2 1 1 1 1  
## 511 2 1 1 1 1  
## 512 2 1 2 1 1  
## 513 2 1 1 1 1  
## 514 1 1 2 1 1  
## 515 3 10 8 10 2  
## 516 3 10 9 10 1  
## 517 1 1 1 1 1  
## 518 1 1 2 1 1  
## 519 2 1 1 1 1  
## 520 4 10 9 1 1  
## 521 3 1 1 1 1  
## 522 3 1 1 1 1  
## 523 3 5 7 3 1  
## 524 4 10 5 3 1  
## 525 2 1 2 1 1  
## 526 2 1 1 1 1  
## 527 2 1 1 1 1  
## 528 2 1 3 1 1  
## 529 2 1 1 1 1  
## 530 1 1 2 1 1  
## 531 4 10 6 9 1  
## 532 2 1 2 1 1  
## 533 1 1 3 1 1  
## 534 2 1 2 1 1  
## 535 2 1 2 1 1  
## 536 2 1 3 1 1  
## 537 2 1 3 1 1  
## 538 2 1 3 1 1  
## 539 2 1 2 1 1  
## 540 2 1 2 1 1  
## 541 2 2 2 1 1  
## 542 2 1 1 1 1  
## 543 2 1 1 1 1  
## 544 2 1 2 1 1  
## 545 2 1 2 1 1  
## 546 2 1 2 1 1  
## 547 4 10 7 10 1  
## 548 1 1 1 1 1  
## 549 1 1 1 1 1  
## 550 4 5 7 8 2  
## 551 2 1 2 1 1  
## 552 2 1 3 1 1  
## 553 2 1 4 2 1  
## 554 2 5 2 1 2  
## 555 2 1 1 1 1  
## 556 2 1 4 8 1  
## 557 1 1 2 1 1  
## 558 2 1 1 1 1  
## 559 2 1 2 1 1  
## 560 2 1 2 1 1  
## 561 2 1 3 1 1  
## 562 2 1 3 1 1  
## 563 2 1 3 1 1  
## 564 2 1 2 1 1  
## 565 2 1 3 2 1  
## 566 5 10 10 10 1  
## 567 2 1 3 1 1  
## 568 2 3 2 1 1  
## 569 6 10 2 5 2  
## 570 6 5 10 3 1  
## 571 8 10 8 2 1  
## 572 3 10 9 10 2  
## 573 2 1 2 1 1  
## 574 2 1 2 1 1  
## 575 4 2 7 7 1  
## 576 2 1 3 1 1  
## 577 2 1 2 1 1  
## 578 2 1 2 1 1  
## 579 2 1 2 1 1  
## 580 2 1 3 1 1  
## 581 2 1 2 1 1  
## 582 5 10 7 5 1  
## 583 4 10 6 10 1  
## 584 2 1 1 1 1  
## 585 3 1 1 1 1  
## 586 2 1 1 1 1  
## 587 6 10 10 10 1  
## 588 2 1 2 2 1  
## 589 6 3 4 1 1  
## 590 2 1 1 1 1  
## 591 4 1 10 1 1  
## 592 4 10 7 6 1  
## 593 3 10 4 1 1  
## 594 2 1 1 1 1  
## 595 4 10 7 1 1  
## 596 2 1 2 1 1  
## 597 2 1 2 1 1  
## 598 2 1 3 1 1  
## 599 2 1 2 1 1  
## 600 1 1 1 1 1  
## 601 2 1 2 1 1  
## 602 1 1 2 1 1  
## 603 2 1 2 1 1  
## 604 4 1 8 10 1  
## 605 5 10 8 1 2  
## 606 5 8 7 8 3  
## 607 2 1 1 1 1  
## 608 2 1 1 1 1  
## 609 10 10 10 1 1  
## 610 2 1 1 1 1  
## 611 3 10 7 1 2  
## 612 5 2 8 5 1  
## 613 6 10 10 10 10  
## 614 2 1 2 1 1  
## 615 1 1 2 1 1  
## 616 2 1 2 1 1  
## 617 2 1 2 1 1  
## 618 1 NA 1 1 1  
## 619 2 1 2 1 1  
## 620 2 1 2 1 1  
## 621 2 1 2 1 1  
## 622 3 2 6 1 1  
## 623 2 1 2 1 1  
## 624 2 1 1 1 1  
## 625 1 1 2 1 1  
## 626 3 4 1 1 1  
## 627 7 6 7 7 3  
## 628 2 5 1 1 1  
## 629 2 1 1 1 1  
## 630 2 1 1 1 1  
## 631 2 1 1 1 1  
## 632 2 1 2 1 1  
## 633 2 1 1 1 1  
## 634 5 3 5 10 1  
## 635 2 1 1 1 1  
## 636 2 1 1 1 1  
## 637 7 1 10 10 3  
## 638 2 2 2 1 1  
## 639 2 1 1 1 1  
## 640 2 1 1 1 1  
## 641 2 1 1 1 1  
## 642 2 1 2 1 1  
## 643 2 1 2 1 1  
## 644 2 1 1 1 1  
## 645 2 1 1 1 1  
## 646 2 1 2 1 1  
## 647 2 1 1 1 1  
## 648 2 1 1 1 1  
## 649 10 2 10 10 10  
## 650 2 1 2 1 1  
## 651 3 4 1 1 1  
## 652 2 1 2 1 1  
## 653 2 1 2 2 1  
## 654 2 1 2 1 1  
## 655 2 1 3 1 1  
## 656 2 1 2 1 1  
## 657 2 1 2 1 1  
## 658 8 1 3 6 1  
## 659 3 10 7 2 3  
## 660 2 1 1 1 1  
## 661 2 1 2 1 1  
## 662 2 1 3 1 1  
## 663 2 1 2 1 1  
## 664 2 1 2 1 1  
## 665 2 1 2 1 1  
## 666 2 1 1 1 1  
## 667 2 1 1 1 2  
## 668 2 1 3 1 1  
## 669 6 1 7 10 3  
## 670 5 5 7 10 1  
## 671 5 8 7 4 1  
## 672 2 1 3 1 1  
## 673 2 1 3 1 1  
## 674 3 1 1 1 1  
## 675 2 1 2 1 1  
## 676 2 1 1 1 1  
## 677 2 1 2 1 1  
## 678 2 1 1 1 1  
## 679 2 1 1 1 1  
## 680 2 1 1 1 1  
## 681 5 10 10 10 7  
## 682 4 10 5 6 3  
## 683 2 1 3 2 1  
## 684 2 1 1 1 1  
## 685 2 1 1 1 1  
## 686 2 1 1 1 1  
## 687 2 1 1 1 1  
## 688 2 1 2 3 1  
## 689 2 1 1 1 1  
## 690 2 1 1 1 8  
## 691 2 1 1 1 1  
## 692 4 5 4 4 1  
## 693 2 1 1 1 1  
## 694 2 1 2 1 2  
## 695 3 2 1 1 1  
## 696 2 1 1 1 1  
## 697 7 3 8 10 2  
## 698 3 4 10 6 1  
## 699 4 5 10 4 1  
## Class  
## 1 benign  
## 2 benign  
## 3 benign  
## 4 benign  
## 5 benign  
## 6 malignant  
## 7 benign  
## 8 benign  
## 9 benign  
## 10 benign  
## 11 benign  
## 12 benign  
## 13 malignant  
## 14 benign  
## 15 malignant  
## 16 malignant  
## 17 benign  
## 18 benign  
## 19 malignant  
## 20 benign  
## 21 malignant  
## 22 malignant  
## 23 benign  
## 24 malignant  
## 25 benign  
## 26 malignant  
## 27 benign  
## 28 benign  
## 29 benign  
## 30 benign  
## 31 benign  
## 32 benign  
## 33 malignant  
## 34 benign  
## 35 benign  
## 36 benign  
## 37 malignant  
## 38 benign  
## 39 malignant  
## 40 malignant  
## 41 benign  
## 42 malignant  
## 43 malignant  
## 44 malignant  
## 45 malignant  
## 46 benign  
## 47 malignant  
## 48 benign  
## 49 benign  
## 50 malignant  
## 51 malignant  
## 52 malignant  
## 53 malignant  
## 54 malignant  
## 55 malignant  
## 56 malignant  
## 57 malignant  
## 58 malignant  
## 59 malignant  
## 60 malignant  
## 61 malignant  
## 62 benign  
## 63 malignant  
## 64 malignant  
## 65 benign  
## 66 malignant  
## 67 benign  
## 68 malignant  
## 69 malignant  
## 70 benign  
## 71 benign  
## 72 malignant  
## 73 benign  
## 74 malignant  
## 75 malignant  
## 76 benign  
## 77 benign  
## 78 benign  
## 79 benign  
## 80 benign  
## 81 benign  
## 82 benign  
## 83 benign  
## 84 benign  
## 85 malignant  
## 86 malignant  
## 87 malignant  
## 88 malignant  
## 89 benign  
## 90 benign  
## 91 benign  
## 92 benign  
## 93 benign  
## 94 benign  
## 95 benign  
## 96 benign  
## 97 benign  
## 98 benign  
## 99 malignant  
## 100 malignant  
## 101 malignant  
## 102 malignant  
## 103 benign  
## 104 malignant  
## 105 malignant  
## 106 malignant  
## 107 malignant  
## 108 malignant  
## 109 benign  
## 110 malignant  
## 111 benign  
## 112 malignant  
## 113 malignant  
## 114 malignant  
## 115 benign  
## 116 benign  
## 117 benign  
## 118 malignant  
## 119 benign  
## 120 benign  
## 121 benign  
## 122 benign  
## 123 malignant  
## 124 malignant  
## 125 malignant  
## 126 benign  
## 127 malignant  
## 128 benign  
## 129 malignant  
## 130 benign  
## 131 benign  
## 132 benign  
## 133 malignant  
## 134 benign  
## 135 benign  
## 136 benign  
## 137 benign  
## 138 benign  
## 139 benign  
## 140 benign  
## 141 benign  
## 142 benign  
## 143 malignant  
## 144 benign  
## 145 benign  
## 146 benign  
## 147 malignant  
## 148 benign  
## 149 benign  
## 150 malignant  
## 151 benign  
## 152 malignant  
## 153 malignant  
## 154 benign  
## 155 benign  
## 156 malignant  
## 157 benign  
## 158 benign  
## 159 benign  
## 160 malignant  
## 161 malignant  
## 162 benign  
## 163 benign  
## 164 benign  
## 165 benign  
## 166 benign  
## 167 malignant  
## 168 malignant  
## 169 benign  
## 170 benign  
## 171 benign  
## 172 benign  
## 173 benign  
## 174 malignant  
## 175 malignant  
## 176 malignant  
## 177 benign  
## 178 malignant  
## 179 benign  
## 180 malignant  
## 181 benign  
## 182 benign  
## 183 benign  
## 184 malignant  
## 185 malignant  
## 186 benign  
## 187 malignant  
## 188 malignant  
## 189 malignant  
## 190 benign  
## 191 malignant  
## 192 malignant  
## 193 benign  
## 194 benign  
## 195 benign  
## 196 benign  
## 197 benign  
## 198 benign  
## 199 benign  
## 200 benign  
## 201 malignant  
## 202 malignant  
## 203 benign  
## 204 benign  
## 205 benign  
## 206 malignant  
## 207 malignant  
## 208 benign  
## 209 benign  
## 210 benign  
## 211 malignant  
## 212 malignant  
## 213 benign  
## 214 malignant  
## 215 malignant  
## 216 malignant  
## 217 benign  
## 218 benign  
## 219 malignant  
## 220 benign  
## 221 benign  
## 222 malignant  
## 223 malignant  
## 224 malignant  
## 225 malignant  
## 226 benign  
## 227 malignant  
## 228 malignant  
## 229 benign  
## 230 malignant  
## 231 malignant  
## 232 malignant  
## 233 benign  
## 234 malignant  
## 235 benign  
## 236 benign  
## 237 malignant  
## 238 malignant  
## 239 malignant  
## 240 malignant  
## 241 benign  
## 242 benign  
## 243 benign  
## 244 benign  
## 245 benign  
## 246 benign  
## 247 malignant  
## 248 malignant  
## 249 benign  
## 250 benign  
## 251 benign  
## 252 malignant  
## 253 benign  
## 254 malignant  
## 255 malignant  
## 256 malignant  
## 257 benign  
## 258 benign  
## 259 benign  
## 260 benign  
## 261 malignant  
## 262 malignant  
## 263 malignant  
## 264 malignant  
## 265 malignant  
## 266 benign  
## 267 malignant  
## 268 malignant  
## 269 malignant  
## 270 benign  
## 271 malignant  
## 272 benign  
## 273 malignant  
## 274 malignant  
## 275 benign  
## 276 benign  
## 277 benign  
## 278 benign  
## 279 benign  
## 280 malignant  
## 281 benign  
## 282 benign  
## 283 malignant  
## 284 malignant  
## 285 malignant  
## 286 malignant  
## 287 malignant  
## 288 benign  
## 289 malignant  
## 290 malignant  
## 291 benign  
## 292 benign  
## 293 malignant  
## 294 malignant  
## 295 benign  
## 296 malignant  
## 297 benign  
## 298 benign  
## 299 benign  
## 300 malignant  
## 301 malignant  
## 302 benign  
## 303 malignant  
## 304 benign  
## 305 malignant  
## 306 malignant  
## 307 benign  
## 308 benign  
## 309 malignant  
## 310 benign  
## 311 benign  
## 312 benign  
## 313 malignant  
## 314 benign  
## 315 benign  
## 316 benign  
## 317 malignant  
## 318 malignant  
## 319 benign  
## 320 benign  
## 321 malignant  
## 322 benign  
## 323 benign  
## 324 malignant  
## 325 benign  
## 326 benign  
## 327 malignant  
## 328 benign  
## 329 malignant  
## 330 malignant  
## 331 malignant  
## 332 benign  
## 333 benign  
## 334 malignant  
## 335 malignant  
## 336 benign  
## 337 malignant  
## 338 benign  
## 339 benign  
## 340 malignant  
## 341 malignant  
## 342 benign  
## 343 benign  
## 344 benign  
## 345 malignant  
## 346 benign  
## 347 benign  
## 348 benign  
## 349 malignant  
## 350 malignant  
## 351 benign  
## 352 benign  
## 353 benign  
## 354 malignant  
## 355 benign  
## 356 benign  
## 357 malignant  
## 358 malignant  
## 359 malignant  
## 360 malignant  
## 361 malignant  
## 362 malignant  
## 363 benign  
## 364 benign  
## 365 benign  
## 366 benign  
## 367 malignant  
## 368 malignant  
## 369 benign  
## 370 benign  
## 371 benign  
## 372 benign  
## 373 benign  
## 374 benign  
## 375 benign  
## 376 benign  
## 377 benign  
## 378 benign  
## 379 benign  
## 380 benign  
## 381 benign  
## 382 malignant  
## 383 benign  
## 384 benign  
## 385 benign  
## 386 benign  
## 387 malignant  
## 388 benign  
## 389 benign  
## 390 benign  
## 391 benign  
## 392 malignant  
## 393 benign  
## 394 benign  
## 395 benign  
## 396 benign  
## 397 benign  
## 398 benign  
## 399 benign  
## 400 benign  
## 401 malignant  
## 402 benign  
## 403 benign  
## 404 benign  
## 405 benign  
## 406 benign  
## 407 benign  
## 408 benign  
## 409 benign  
## 410 benign  
## 411 benign  
## 412 benign  
## 413 malignant  
## 414 benign  
## 415 malignant  
## 416 benign  
## 417 malignant  
## 418 benign  
## 419 benign  
## 420 benign  
## 421 benign  
## 422 malignant  
## 423 benign  
## 424 benign  
## 425 benign  
## 426 malignant  
## 427 benign  
## 428 malignant  
## 429 benign  
## 430 benign  
## 431 benign  
## 432 benign  
## 433 benign  
## 434 benign  
## 435 benign  
## 436 malignant  
## 437 malignant  
## 438 benign  
## 439 benign  
## 440 benign  
## 441 malignant  
## 442 benign  
## 443 benign  
## 444 benign  
## 445 benign  
## 446 benign  
## 447 benign  
## 448 benign  
## 449 benign  
## 450 malignant  
## 451 benign  
## 452 benign  
## 453 benign  
## 454 malignant  
## 455 benign  
## 456 malignant  
## 457 malignant  
## 458 malignant  
## 459 benign  
## 460 benign  
## 461 benign  
## 462 benign  
## 463 benign  
## 464 benign  
## 465 benign  
## 466 malignant  
## 467 malignant  
## 468 malignant  
## 469 benign  
## 470 benign  
## 471 benign  
## 472 benign  
## 473 benign  
## 474 benign  
## 475 benign  
## 476 benign  
## 477 benign  
## 478 benign  
## 479 benign  
## 480 malignant  
## 481 benign  
## 482 benign  
## 483 malignant  
## 484 malignant  
## 485 benign  
## 486 benign  
## 487 benign  
## 488 malignant  
## 489 malignant  
## 490 malignant  
## 491 benign  
## 492 malignant  
## 493 benign  
## 494 malignant  
## 495 benign  
## 496 benign  
## 497 benign  
## 498 benign  
## 499 benign  
## 500 benign  
## 501 benign  
## 502 benign  
## 503 benign  
## 504 benign  
## 505 benign  
## 506 benign  
## 507 malignant  
## 508 benign  
## 509 benign  
## 510 benign  
## 511 benign  
## 512 benign  
## 513 benign  
## 514 benign  
## 515 malignant  
## 516 malignant  
## 517 benign  
## 518 benign  
## 519 benign  
## 520 malignant  
## 521 benign  
## 522 benign  
## 523 malignant  
## 524 malignant  
## 525 benign  
## 526 benign  
## 527 benign  
## 528 benign  
## 529 benign  
## 530 benign  
## 531 malignant  
## 532 benign  
## 533 benign  
## 534 benign  
## 535 benign  
## 536 benign  
## 537 benign  
## 538 benign  
## 539 benign  
## 540 benign  
## 541 benign  
## 542 benign  
## 543 benign  
## 544 benign  
## 545 benign  
## 546 benign  
## 547 malignant  
## 548 benign  
## 549 benign  
## 550 malignant  
## 551 benign  
## 552 benign  
## 553 benign  
## 554 benign  
## 555 benign  
## 556 benign  
## 557 benign  
## 558 benign  
## 559 benign  
## 560 benign  
## 561 benign  
## 562 benign  
## 563 benign  
## 564 benign  
## 565 benign  
## 566 malignant  
## 567 benign  
## 568 benign  
## 569 malignant  
## 570 malignant  
## 571 malignant  
## 572 malignant  
## 573 benign  
## 574 benign  
## 575 malignant  
## 576 benign  
## 577 benign  
## 578 benign  
## 579 benign  
## 580 benign  
## 581 benign  
## 582 malignant  
## 583 malignant  
## 584 benign  
## 585 benign  
## 586 benign  
## 587 malignant  
## 588 benign  
## 589 malignant  
## 590 benign  
## 591 malignant  
## 592 malignant  
## 593 malignant  
## 594 benign  
## 595 malignant  
## 596 benign  
## 597 benign  
## 598 benign  
## 599 benign  
## 600 benign  
## 601 benign  
## 602 benign  
## 603 benign  
## 604 malignant  
## 605 malignant  
## 606 malignant  
## 607 benign  
## 608 benign  
## 609 malignant  
## 610 benign  
## 611 malignant  
## 612 malignant  
## 613 malignant  
## 614 benign  
## 615 benign  
## 616 benign  
## 617 benign  
## 618 benign  
## 619 benign  
## 620 benign  
## 621 benign  
## 622 benign  
## 623 benign  
## 624 benign  
## 625 benign  
## 626 benign  
## 627 malignant  
## 628 benign  
## 629 benign  
## 630 benign  
## 631 benign  
## 632 benign  
## 633 benign  
## 634 malignant  
## 635 benign  
## 636 benign  
## 637 malignant  
## 638 benign  
## 639 benign  
## 640 benign  
## 641 benign  
## 642 benign  
## 643 benign  
## 644 benign  
## 645 benign  
## 646 benign  
## 647 benign  
## 648 benign  
## 649 malignant  
## 650 benign  
## 651 benign  
## 652 benign  
## 653 benign  
## 654 benign  
## 655 benign  
## 656 benign  
## 657 benign  
## 658 benign  
## 659 malignant  
## 660 benign  
## 661 benign  
## 662 benign  
## 663 benign  
## 664 benign  
## 665 benign  
## 666 benign  
## 667 benign  
## 668 benign  
## 669 malignant  
## 670 malignant  
## 671 malignant  
## 672 benign  
## 673 benign  
## 674 benign  
## 675 benign  
## 676 benign  
## 677 benign  
## 678 benign  
## 679 benign  
## 680 benign  
## 681 malignant  
## 682 malignant  
## 683 benign  
## 684 benign  
## 685 benign  
## 686 benign  
## 687 benign  
## 688 benign  
## 689 benign  
## 690 benign  
## 691 benign  
## 692 malignant  
## 693 benign  
## 694 benign  
## 695 benign  
## 696 benign  
## 697 malignant  
## 698 malignant  
## 699 malignant

# Loading Libraries  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1   
## ✔ readr 2.1.2 ✔ forcats 0.5.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(ggplot2)  
library(caret)

## Loading required package: lattice  
##   
## Attaching package: 'caret'  
##   
## The following object is masked from 'package:purrr':  
##   
## lift

library(lattice)  
library(rpart)  
library(rattle)

## Loading required package: bitops  
## Rattle: A free graphical interface for data science with R.  
## Version 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.

# Q) As a preprocessing step, remove the ID Number

# column and exclude rows with NA from the dataset.

# As a pre processing step, removing the ID Number column   
cncr\_ID <- cncrdata %>% select(-c("IDNumber"))  
  
# Excluding rows with NA from the data set  
cncr\_IDNA <- na.omit(cncr\_ID)  
  
print(cncr\_IDNA)

## ClumpThickness UniformCellSize UniformCellShape MarginalAdhesion  
## 1 5 1 1 1  
## 2 5 4 4 5  
## 3 3 1 1 1  
## 4 6 8 8 1  
## 5 4 1 1 3  
## 6 8 10 10 8  
## 7 1 1 1 1  
## 8 2 1 2 1  
## 9 2 1 1 1  
## 10 4 2 1 1  
## 11 1 1 1 1  
## 12 2 1 1 1  
## 13 5 3 3 3  
## 14 1 1 1 1  
## 15 8 7 5 10  
## 16 7 4 6 4  
## 17 4 1 1 1  
## 18 4 1 1 1  
## 19 10 7 7 6  
## 20 6 1 1 1  
## 21 7 3 2 10  
## 22 10 5 5 3  
## 23 3 1 1 1  
## 25 1 1 1 1  
## 26 5 2 3 4  
## 27 3 2 1 1  
## 28 5 1 1 1  
## 29 2 1 1 1  
## 30 1 1 3 1  
## 31 3 1 1 1  
## 32 2 1 1 1  
## 33 10 7 7 3  
## 34 2 1 1 2  
## 35 3 1 2 1  
## 36 2 1 1 1  
## 37 10 10 10 8  
## 38 6 2 1 1  
## 39 5 4 4 9  
## 40 2 5 3 3  
## 42 10 4 3 1  
## 43 6 10 10 2  
## 44 5 6 5 6  
## 45 10 10 10 4  
## 46 1 1 1 1  
## 47 3 7 7 4  
## 48 1 1 1 1  
## 49 4 1 1 3  
## 50 7 8 7 2  
## 51 9 5 8 1  
## 52 5 3 3 4  
## 53 10 3 6 2  
## 54 5 5 5 8  
## 55 10 5 5 6  
## 56 10 6 6 3  
## 57 8 10 10 1  
## 58 8 2 4 1  
## 59 5 2 3 1  
## 60 9 5 5 2  
## 61 5 3 5 5  
## 62 1 1 1 1  
## 63 9 10 10 1  
## 64 6 3 4 1  
## 65 1 1 1 1  
## 66 10 4 2 1  
## 67 4 1 1 1  
## 68 5 3 4 1  
## 69 8 3 8 3  
## 70 1 1 1 1  
## 71 5 1 3 1  
## 72 6 10 2 8  
## 73 1 3 3 2  
## 74 9 4 5 10  
## 75 10 6 4 1  
## 76 1 1 2 1  
## 77 1 1 4 1  
## 78 5 3 1 2  
## 79 3 1 1 1  
## 80 2 1 1 1  
## 81 2 2 2 1  
## 82 4 1 1 2  
## 83 5 2 1 1  
## 84 3 1 1 1  
## 85 3 5 7 8  
## 86 5 10 6 1  
## 87 3 3 6 4  
## 88 3 6 6 6  
## 89 4 1 1 1  
## 90 2 1 1 2  
## 91 1 1 1 1  
## 92 3 1 1 2  
## 93 4 1 1 1  
## 94 1 1 1 1  
## 95 2 1 1 1  
## 96 1 1 1 1  
## 97 2 1 1 2  
## 98 5 1 1 1  
## 99 9 6 9 2  
## 100 7 5 6 10  
## 101 10 3 5 1  
## 102 2 3 4 4  
## 103 4 1 2 1  
## 104 8 2 3 1  
## 105 10 10 10 10  
## 106 7 3 4 4  
## 107 10 10 10 8  
## 108 1 6 8 10  
## 109 1 1 1 1  
## 110 6 5 4 4  
## 111 1 3 1 2  
## 112 8 6 4 3  
## 113 10 3 3 10  
## 114 10 10 10 3  
## 115 3 3 2 1  
## 116 1 1 1 1  
## 117 8 3 3 1  
## 118 4 5 5 10  
## 119 1 1 1 1  
## 120 3 2 1 1  
## 121 1 1 2 2  
## 122 4 2 1 1  
## 123 10 10 10 2  
## 124 5 3 5 1  
## 125 5 4 6 7  
## 126 1 1 1 1  
## 127 7 5 3 7  
## 128 3 1 1 1  
## 129 8 3 5 4  
## 130 1 1 1 1  
## 131 5 1 3 1  
## 132 2 1 1 1  
## 133 5 10 8 10  
## 134 3 1 1 1  
## 135 3 1 1 1  
## 136 5 1 1 1  
## 137 4 1 1 1  
## 138 3 1 1 1  
## 139 4 1 2 1  
## 141 3 1 1 1  
## 142 2 1 1 1  
## 143 9 5 5 4  
## 144 1 1 1 1  
## 145 2 1 1 1  
## 147 3 4 5 2  
## 148 1 1 1 1  
## 149 3 1 1 3  
## 150 8 8 7 4  
## 151 1 1 1 1  
## 152 7 2 4 1  
## 153 10 10 8 6  
## 154 4 1 1 1  
## 155 1 1 1 1  
## 156 5 5 5 6  
## 157 1 2 2 1  
## 158 2 1 1 1  
## 160 9 9 10 3  
## 161 10 7 7 4  
## 162 4 1 1 1  
## 163 3 1 1 1  
## 164 1 1 1 2  
## 166 4 1 1 1  
## 167 5 6 7 8  
## 168 10 8 10 10  
## 169 3 1 1 1  
## 170 1 1 1 2  
## 171 3 1 1 1  
## 172 1 1 1 1  
## 173 1 1 1 1  
## 174 6 10 10 10  
## 175 8 6 5 4  
## 176 5 8 7 7  
## 177 2 1 1 1  
## 178 5 10 10 3  
## 179 4 1 1 1  
## 180 5 3 3 3  
## 181 1 1 1 1  
## 182 1 1 1 1  
## 183 6 1 1 1  
## 184 5 8 8 8  
## 185 8 7 6 4  
## 186 2 1 1 1  
## 187 1 5 8 6  
## 188 10 5 6 10  
## 189 5 8 4 10  
## 190 1 2 3 1  
## 191 10 10 10 8  
## 192 7 5 10 10  
## 193 5 1 1 1  
## 194 1 1 1 1  
## 195 3 1 1 1  
## 196 4 1 1 1  
## 197 8 4 4 5  
## 198 5 1 1 4  
## 199 1 1 1 1  
## 200 3 1 1 1  
## 201 9 7 7 5  
## 202 10 8 8 4  
## 203 1 1 1 1  
## 204 5 1 1 1  
## 205 1 1 1 1  
## 206 5 10 10 9  
## 207 10 10 9 3  
## 208 1 1 1 1  
## 209 1 1 1 1  
## 210 5 1 1 1  
## 211 8 10 10 10  
## 212 8 10 8 8  
## 213 1 1 1 1  
## 214 10 10 10 10  
## 215 10 10 10 10  
## 216 8 7 8 7  
## 217 1 1 1 1  
## 218 1 1 1 1  
## 219 6 10 7 7  
## 220 6 1 3 1  
## 221 1 1 1 2  
## 222 10 6 4 3  
## 223 4 1 1 3  
## 224 7 5 6 3  
## 225 10 5 5 6  
## 226 1 1 1 1  
## 227 10 5 7 4  
## 228 8 9 9 5  
## 229 1 1 1 1  
## 230 10 10 10 3  
## 231 7 4 7 4  
## 232 6 8 7 5  
## 233 8 4 6 3  
## 234 10 4 5 5  
## 235 3 3 2 1  
## 237 10 8 8 2  
## 238 9 8 8 5  
## 239 8 10 10 8  
## 240 10 4 3 2  
## 241 5 1 3 3  
## 242 3 1 1 3  
## 243 2 1 1 1  
## 244 1 1 1 1  
## 245 1 1 1 1  
## 246 5 1 1 2  
## 247 8 10 10 8  
## 248 8 4 4 1  
## 249 4 1 1 1  
## 251 1 2 2 1  
## 252 10 4 4 10  
## 253 6 3 3 5  
## 254 6 10 10 2  
## 255 9 10 10 1  
## 256 5 6 6 2  
## 257 3 1 1 1  
## 258 3 1 1 1  
## 259 3 1 1 1  
## 260 5 7 7 1  
## 261 10 5 8 10  
## 262 5 10 10 6  
## 263 8 8 9 4  
## 264 10 4 4 10  
## 265 7 9 4 10  
## 266 5 1 4 1  
## 267 10 10 6 3  
## 268 3 3 5 2  
## 269 10 8 8 2  
## 270 1 1 1 1  
## 271 8 4 7 1  
## 272 5 1 1 1  
## 273 3 3 5 2  
## 274 7 2 4 1  
## 275 3 1 1 1  
## 277 3 1 1 1  
## 278 1 1 1 1  
## 279 1 1 1 1  
## 280 10 5 7 3  
## 281 3 1 1 1  
## 282 2 1 1 2  
## 283 1 4 3 10  
## 284 10 4 6 1  
## 285 7 4 5 10  
## 286 8 10 10 10  
## 287 10 10 10 10  
## 288 3 1 1 1  
## 289 6 1 3 1  
## 290 5 6 6 8  
## 291 1 1 1 1  
## 292 1 1 1 1  
## 294 10 4 4 6  
## 296 5 5 7 8  
## 297 5 3 4 3  
## 299 8 2 1 1  
## 300 9 1 2 6  
## 301 8 4 10 5  
## 302 1 1 1 1  
## 303 10 10 10 7  
## 304 1 1 1 1  
## 305 8 3 4 9  
## 306 10 8 4 4  
## 307 1 1 1 1  
## 308 1 1 1 1  
## 309 7 8 7 6  
## 310 3 1 1 1  
## 311 2 1 1 1  
## 312 1 1 1 1  
## 313 8 6 4 10  
## 314 1 1 1 1  
## 315 1 1 1 1  
## 317 5 5 5 2  
## 318 6 8 7 8  
## 319 1 1 1 1  
## 320 4 4 4 4  
## 321 7 6 3 2  
## 323 3 1 1 1  
## 324 5 4 6 10  
## 325 1 1 1 1  
## 326 3 2 2 1  
## 327 10 1 1 1  
## 328 1 1 1 1  
## 329 8 10 3 2  
## 330 10 4 6 4  
## 331 10 4 7 2  
## 332 5 1 1 1  
## 333 5 2 2 2  
## 334 5 4 6 6  
## 335 8 6 7 3  
## 336 1 1 1 1  
## 337 6 5 5 8  
## 338 1 1 1 1  
## 339 1 1 1 1  
## 340 8 5 5 5  
## 341 10 3 3 1  
## 342 1 1 1 1  
## 343 2 1 1 1  
## 344 1 1 1 1  
## 345 7 6 4 8  
## 346 1 1 1 1  
## 347 5 2 2 2  
## 348 1 1 1 1  
## 349 3 4 4 10  
## 350 4 2 3 5  
## 351 5 1 1 3  
## 352 2 1 1 1  
## 353 3 4 5 3  
## 354 2 7 10 10  
## 355 1 1 1 1  
## 356 4 1 1 1  
## 357 5 3 3 1  
## 358 8 10 10 7  
## 359 8 10 5 3  
## 360 10 3 5 4  
## 361 6 10 10 10  
## 362 3 10 3 10  
## 363 3 2 2 1  
## 364 4 4 4 2  
## 365 2 1 1 1  
## 366 2 1 1 1  
## 367 6 10 10 10  
## 368 5 8 8 10  
## 369 1 1 3 1  
## 370 1 1 3 1  
## 371 4 3 2 1  
## 372 1 1 3 1  
## 373 4 1 2 1  
## 374 5 1 1 2  
## 375 3 1 2 1  
## 376 1 1 1 1  
## 377 1 1 1 1  
## 378 1 1 1 1  
## 379 3 1 1 4  
## 380 5 3 4 1  
## 381 1 1 1 1  
## 382 10 6 3 6  
## 383 3 2 2 2  
## 384 2 1 1 1  
## 385 2 1 1 1  
## 386 3 3 2 2  
## 387 7 6 6 3  
## 388 5 3 3 2  
## 389 2 1 1 1  
## 390 5 1 1 1  
## 391 1 1 1 2  
## 392 10 8 7 4  
## 393 3 1 1 1  
## 394 1 1 1 1  
## 395 1 2 3 1  
## 396 3 1 1 1  
## 397 3 1 1 1  
## 398 4 1 1 1  
## 399 3 2 1 1  
## 400 1 2 3 1  
## 401 3 10 8 7  
## 402 3 1 1 1  
## 403 5 3 3 1  
## 404 3 1 1 1  
## 405 1 2 1 3  
## 406 1 1 1 1  
## 407 4 2 2 1  
## 408 1 1 1 1  
## 409 2 3 2 2  
## 410 3 1 2 1  
## 411 1 1 1 1  
## 413 10 10 10 6  
## 414 5 1 2 1  
## 415 8 5 6 2  
## 416 3 3 2 6  
## 417 8 7 8 5  
## 418 1 1 1 1  
## 419 5 2 2 2  
## 420 2 3 1 1  
## 421 3 2 2 3  
## 422 10 10 10 7  
## 423 4 3 3 1  
## 424 5 1 3 1  
## 425 3 1 1 1  
## 426 9 10 10 10  
## 427 5 3 6 1  
## 428 8 7 8 2  
## 429 1 1 1 1  
## 430 2 1 1 1  
## 431 1 3 1 1  
## 432 5 1 1 3  
## 433 5 1 1 1  
## 434 3 2 2 3  
## 435 6 9 7 5  
## 436 10 8 10 1  
## 437 10 10 10 1  
## 438 4 1 1 1  
## 439 4 1 3 3  
## 440 5 1 1 1  
## 441 10 4 3 10  
## 442 5 2 2 4  
## 443 1 1 1 3  
## 444 1 1 1 1  
## 445 5 1 1 6  
## 446 2 1 1 1  
## 447 1 1 1 1  
## 448 5 1 1 1  
## 449 1 1 1 1  
## 450 5 7 9 8  
## 451 4 1 1 3  
## 452 5 1 1 1  
## 453 3 1 1 3  
## 454 4 5 5 8  
## 455 2 3 1 1  
## 456 10 2 2 1  
## 457 10 6 5 8  
## 458 8 8 9 6  
## 459 5 1 2 1  
## 460 5 1 3 1  
## 461 5 1 1 3  
## 462 3 1 1 1  
## 463 6 1 1 3  
## 464 4 1 1 1  
## 465 4 1 1 1  
## 466 10 9 8 7  
## 467 10 6 6 2  
## 468 6 6 6 5  
## 469 4 1 1 1  
## 470 1 1 2 1  
## 471 3 1 1 1  
## 472 6 1 1 3  
## 473 6 1 1 1  
## 474 4 1 1 1  
## 475 5 1 1 1  
## 476 3 1 1 1  
## 477 4 1 2 1  
## 478 4 1 1 1  
## 479 5 2 1 1  
## 480 4 8 7 10  
## 481 5 1 1 1  
## 482 5 3 2 4  
## 483 9 10 10 10  
## 484 8 7 8 5  
## 485 5 1 2 1  
## 486 1 1 1 3  
## 487 3 1 1 1  
## 488 10 10 10 10  
## 489 3 6 4 10  
## 490 6 3 2 1  
## 491 1 1 1 1  
## 492 5 8 9 4  
## 493 4 1 1 1  
## 494 5 10 10 10  
## 495 5 1 2 10  
## 496 3 1 1 1  
## 497 1 1 1 1  
## 498 4 2 1 1  
## 499 4 1 1 1  
## 500 4 1 1 1  
## 501 6 1 1 1  
## 502 4 1 1 1  
## 503 4 1 1 2  
## 504 4 1 1 1  
## 505 1 1 1 1  
## 506 3 3 1 1  
## 507 8 10 10 10  
## 508 1 1 1 1  
## 509 5 1 1 1  
## 510 2 1 1 1  
## 511 1 1 1 1  
## 512 5 1 1 1  
## 513 5 1 1 1  
## 514 3 1 1 1  
## 515 6 6 7 10  
## 516 4 10 4 7  
## 517 1 1 1 1  
## 518 1 1 1 1  
## 519 3 1 2 2  
## 520 4 7 8 3  
## 521 1 1 1 1  
## 522 4 1 1 1  
## 523 10 4 5 4  
## 524 7 5 6 10  
## 525 3 1 1 1  
## 526 3 1 1 2  
## 527 4 1 1 1  
## 528 4 1 1 1  
## 529 6 1 3 2  
## 530 4 1 1 1  
## 531 7 4 4 3  
## 532 4 2 2 1  
## 533 1 1 1 1  
## 534 3 1 1 1  
## 535 2 1 1 1  
## 536 1 1 3 2  
## 537 5 1 1 1  
## 538 5 1 2 1  
## 539 4 1 1 1  
## 540 6 1 1 1  
## 541 5 1 1 1  
## 542 3 1 1 1  
## 543 5 3 1 1  
## 544 4 1 1 1  
## 545 2 1 3 2  
## 546 5 1 1 1  
## 547 6 10 10 10  
## 548 2 1 1 1  
## 549 3 1 1 1  
## 550 7 8 3 7  
## 551 3 1 1 1  
## 552 1 1 1 1  
## 553 3 2 2 2  
## 554 4 4 2 1  
## 555 3 1 1 1  
## 556 4 3 1 1  
## 557 5 2 2 2  
## 558 5 1 1 3  
## 559 2 1 1 1  
## 560 5 1 1 1  
## 561 5 1 1 1  
## 562 5 1 1 1  
## 563 1 1 1 1  
## 564 3 1 1 1  
## 565 4 1 1 1  
## 566 5 7 10 10  
## 567 3 1 2 1  
## 568 4 1 1 1  
## 569 8 4 4 1  
## 570 10 10 8 10  
## 571 8 10 4 4  
## 572 7 6 10 5  
## 573 3 1 1 1  
## 574 1 1 1 1  
## 575 10 9 7 3  
## 576 5 1 2 1  
## 577 5 1 1 1  
## 578 1 1 1 1  
## 579 1 1 1 1  
## 580 1 1 1 1  
## 581 5 1 2 1  
## 582 5 7 10 6  
## 583 6 10 5 5  
## 584 3 1 1 1  
## 585 5 1 1 6  
## 586 1 1 1 1  
## 587 8 10 10 10  
## 588 5 1 1 1  
## 589 9 8 8 9  
## 590 5 1 1 1  
## 591 4 10 8 5  
## 592 2 5 7 6  
## 593 10 3 4 5  
## 594 5 1 2 1  
## 595 4 8 6 3  
## 596 5 1 1 1  
## 597 4 1 2 1  
## 598 5 1 3 1  
## 599 3 1 1 1  
## 600 5 2 4 1  
## 601 3 1 1 1  
## 602 1 1 1 1  
## 603 4 1 1 1  
## 604 5 4 6 8  
## 605 5 3 2 8  
## 606 10 5 10 3  
## 607 4 1 1 2  
## 608 1 1 1 1  
## 609 5 10 10 10  
## 610 5 1 1 1  
## 611 10 4 3 10  
## 612 5 10 10 10  
## 613 8 10 10 10  
## 614 2 3 1 1  
## 615 2 1 1 1  
## 616 4 1 3 1  
## 617 3 1 1 1  
## 619 4 1 1 1  
## 620 5 1 1 1  
## 621 3 1 1 1  
## 622 6 3 3 3  
## 623 7 1 2 3  
## 624 1 1 1 1  
## 625 5 1 1 2  
## 626 3 1 3 1  
## 627 4 6 6 5  
## 628 2 1 1 1  
## 629 2 1 1 1  
## 630 4 1 1 1  
## 631 6 2 3 1  
## 632 5 1 1 1  
## 633 1 1 1 1  
## 634 8 7 4 4  
## 635 3 1 1 1  
## 636 3 1 4 1  
## 637 10 10 7 8  
## 638 4 2 4 3  
## 639 4 1 1 1  
## 640 5 1 1 3  
## 641 4 1 1 3  
## 642 3 1 1 1  
## 643 3 1 1 1  
## 644 1 1 1 1  
## 645 2 1 1 1  
## 646 3 1 1 1  
## 647 1 2 2 1  
## 648 1 1 1 3  
## 649 5 10 10 10  
## 650 3 1 1 1  
## 651 3 1 1 2  
## 652 1 2 1 3  
## 653 5 1 1 1  
## 654 4 1 1 1  
## 655 3 1 1 1  
## 656 3 1 1 1  
## 657 5 1 1 1  
## 658 5 4 5 1  
## 659 7 8 8 7  
## 660 1 1 1 1  
## 661 1 1 1 1  
## 662 4 1 1 1  
## 663 1 1 3 1  
## 664 1 1 3 1  
## 665 3 1 1 3  
## 666 1 1 1 1  
## 667 5 2 2 2  
## 668 3 1 1 1  
## 669 5 7 4 1  
## 670 5 10 10 8  
## 671 3 10 7 8  
## 672 3 2 1 2  
## 673 2 1 1 1  
## 674 5 3 2 1  
## 675 1 1 1 1  
## 676 4 1 4 1  
## 677 1 1 2 1  
## 678 5 1 1 1  
## 679 1 1 1 1  
## 680 2 1 1 1  
## 681 10 10 10 10  
## 682 5 10 10 10  
## 683 5 1 1 1  
## 684 1 1 1 1  
## 685 1 1 1 1  
## 686 1 1 1 1  
## 687 1 1 1 1  
## 688 3 1 1 1  
## 689 4 1 1 1  
## 690 1 1 1 1  
## 691 1 1 1 3  
## 692 5 10 10 5  
## 693 3 1 1 1  
## 694 3 1 1 1  
## 695 3 1 1 1  
## 696 2 1 1 1  
## 697 5 10 10 3  
## 698 4 8 6 4  
## 699 4 8 8 5  
## EpithelialCellSize BareNuclei BlandChromatin NormalNucleoli Mitoses  
## 1 2 1 3 1 1  
## 2 7 10 3 2 1  
## 3 2 2 3 1 1  
## 4 3 4 3 7 1  
## 5 2 1 3 1 1  
## 6 7 10 9 7 1  
## 7 2 10 3 1 1  
## 8 2 1 3 1 1  
## 9 2 1 1 1 5  
## 10 2 1 2 1 1  
## 11 1 1 3 1 1  
## 12 2 1 2 1 1  
## 13 2 3 4 4 1  
## 14 2 3 3 1 1  
## 15 7 9 5 5 4  
## 16 6 1 4 3 1  
## 17 2 1 2 1 1  
## 18 2 1 3 1 1  
## 19 4 10 4 1 2  
## 20 2 1 3 1 1  
## 21 5 10 5 4 4  
## 22 6 7 7 10 1  
## 23 2 1 2 1 1  
## 25 2 1 3 1 1  
## 26 2 7 3 6 1  
## 27 1 1 2 1 1  
## 28 2 1 2 1 1  
## 29 2 1 2 1 1  
## 30 2 1 1 1 1  
## 31 1 1 2 1 1  
## 32 2 1 3 1 1  
## 33 8 5 7 4 3  
## 34 2 1 3 1 1  
## 35 2 1 2 1 1  
## 36 2 1 2 1 1  
## 37 6 1 8 9 1  
## 38 1 1 7 1 1  
## 39 2 10 5 6 1  
## 40 6 7 7 5 1  
## 42 3 3 6 5 2  
## 43 8 10 7 3 3  
## 44 10 1 3 1 1  
## 45 8 1 8 10 1  
## 46 2 1 2 1 2  
## 47 4 9 4 8 1  
## 48 2 1 2 1 1  
## 49 2 1 3 1 1  
## 50 4 8 3 8 2  
## 51 2 3 2 1 5  
## 52 2 4 3 4 1  
## 53 3 5 4 10 2  
## 54 10 8 7 3 7  
## 55 8 8 7 1 1  
## 56 4 5 3 6 1  
## 57 3 6 3 9 1  
## 58 5 1 5 4 4  
## 59 6 10 5 1 1  
## 60 2 2 5 1 1  
## 61 3 3 4 10 1  
## 62 2 2 2 1 1  
## 63 10 8 3 3 1  
## 64 5 2 3 9 1  
## 65 2 1 2 1 1  
## 66 3 2 4 3 10  
## 67 2 1 3 1 1  
## 68 8 10 4 9 1  
## 69 4 9 8 9 8  
## 70 2 1 3 2 1  
## 71 2 1 2 1 1  
## 72 10 2 7 8 10  
## 73 2 1 7 2 1  
## 74 6 10 4 8 1  
## 75 3 4 3 2 3  
## 76 2 2 4 2 1  
## 77 2 1 2 1 1  
## 78 2 1 2 1 1  
## 79 2 3 3 1 1  
## 80 3 1 2 1 1  
## 81 1 1 7 1 1  
## 82 2 1 2 1 1  
## 83 2 1 3 1 1  
## 84 2 2 7 1 1  
## 85 8 9 7 10 7  
## 86 10 4 4 10 10  
## 87 5 8 4 4 1  
## 88 5 10 6 8 3  
## 89 2 1 3 1 1  
## 90 3 1 2 1 1  
## 91 2 1 3 1 1  
## 92 2 1 1 1 1  
## 93 2 1 3 1 1  
## 94 2 1 2 1 1  
## 95 2 1 3 1 1  
## 96 2 1 3 1 1  
## 97 2 1 1 1 1  
## 98 2 1 3 1 1  
## 99 10 6 2 9 10  
## 100 5 10 7 9 4  
## 101 10 5 3 10 2  
## 102 2 5 2 5 1  
## 103 2 1 3 1 1  
## 104 6 3 7 1 1  
## 105 10 1 8 8 8  
## 106 3 3 3 2 7  
## 107 2 10 4 1 1  
## 108 8 10 5 7 1  
## 109 2 1 2 3 1  
## 110 3 9 7 8 3  
## 111 2 2 5 3 2  
## 112 5 9 3 1 1  
## 113 2 10 7 3 3  
## 114 10 8 8 1 1  
## 115 2 3 3 1 1  
## 116 2 5 1 1 1  
## 117 2 2 3 2 1  
## 118 4 10 7 5 8  
## 119 4 3 1 1 1  
## 120 2 2 3 1 1  
## 121 2 1 3 1 1  
## 122 2 2 3 1 1  
## 123 10 10 5 3 3  
## 124 8 10 5 3 1  
## 125 9 7 8 10 1  
## 126 2 1 2 1 1  
## 127 4 10 7 5 5  
## 128 2 1 3 1 1  
## 129 5 10 1 6 2  
## 130 10 1 1 1 1  
## 131 2 1 2 1 1  
## 132 2 1 3 1 1  
## 133 8 10 3 6 3  
## 134 2 1 2 2 1  
## 135 3 1 2 1 1  
## 136 2 2 3 3 1  
## 137 2 1 2 1 1  
## 138 2 1 1 1 1  
## 139 2 1 2 1 1  
## 141 2 1 1 1 1  
## 142 2 1 1 1 1  
## 143 4 5 4 3 3  
## 144 2 5 1 1 1  
## 145 2 1 2 1 1  
## 147 6 8 4 1 1  
## 148 3 2 2 1 1  
## 149 8 1 5 8 1  
## 150 10 10 7 8 7  
## 151 1 1 3 1 1  
## 152 6 10 5 4 3  
## 153 4 5 8 10 1  
## 154 2 3 1 1 1  
## 155 2 1 1 1 1  
## 156 3 10 3 1 1  
## 157 2 1 2 1 1  
## 158 2 1 3 1 1  
## 160 6 10 7 10 6  
## 161 5 10 5 7 2  
## 162 2 1 3 2 1  
## 163 2 1 3 1 1  
## 164 1 3 1 1 7  
## 166 2 2 3 2 1  
## 167 8 10 3 10 3  
## 168 6 1 3 1 10  
## 169 2 1 3 1 1  
## 170 1 1 1 1 1  
## 171 2 1 1 1 1  
## 172 2 1 3 1 1  
## 173 2 1 2 1 1  
## 174 8 10 10 10 7  
## 175 3 10 6 1 1  
## 176 10 10 5 7 1  
## 177 2 1 3 1 1  
## 178 8 1 5 10 3  
## 179 2 1 3 1 1  
## 180 6 10 3 1 1  
## 181 1 1 3 1 1  
## 182 2 1 1 1 1  
## 183 2 1 3 1 1  
## 184 5 10 7 8 1  
## 185 4 10 5 1 1  
## 186 1 1 3 1 1  
## 187 5 8 7 10 1  
## 188 6 10 7 7 10  
## 189 5 8 9 10 1  
## 190 2 1 3 1 1  
## 191 6 8 7 10 1  
## 192 10 10 4 10 3  
## 193 2 1 2 1 1  
## 194 2 1 3 1 1  
## 195 2 1 3 1 1  
## 196 2 1 3 1 1  
## 197 4 7 7 8 2  
## 198 2 1 3 1 1  
## 199 2 1 1 1 1  
## 200 2 1 2 1 1  
## 201 5 10 7 8 3  
## 202 10 10 8 1 1  
## 203 2 1 3 1 1  
## 204 2 1 3 1 1  
## 205 2 1 3 1 1  
## 206 6 10 7 10 5  
## 207 7 5 3 5 1  
## 208 1 1 3 1 1  
## 209 1 1 3 1 1  
## 210 1 1 3 1 1  
## 211 5 10 8 10 6  
## 212 4 8 7 7 1  
## 213 2 1 3 1 1  
## 214 7 10 7 10 4  
## 215 3 10 10 6 1  
## 216 5 5 5 10 2  
## 217 2 1 2 1 1  
## 218 2 1 3 1 1  
## 219 6 4 8 10 2  
## 220 2 1 3 1 1  
## 221 2 1 3 1 1  
## 222 10 10 9 10 1  
## 223 1 5 2 1 1  
## 224 3 8 7 4 1  
## 225 3 10 7 9 2  
## 226 2 1 2 1 1  
## 227 4 10 8 9 1  
## 228 3 5 7 7 1  
## 229 1 1 3 1 1  
## 230 10 10 9 10 1  
## 231 3 7 7 6 1  
## 232 6 8 8 9 2  
## 233 3 1 4 3 1  
## 234 5 10 4 1 1  
## 235 3 1 3 6 1  
## 237 8 10 4 8 10  
## 238 6 2 4 10 4  
## 239 6 9 3 10 10  
## 240 3 10 5 3 2  
## 241 2 2 2 3 1  
## 242 1 1 3 1 1  
## 243 2 1 3 1 1  
## 244 2 5 5 1 1  
## 245 2 1 3 1 1  
## 246 2 2 3 1 1  
## 247 5 10 7 8 1  
## 248 2 9 3 3 1  
## 249 2 1 3 6 1  
## 251 2 1 1 1 1  
## 252 2 10 5 3 3  
## 253 3 10 3 5 3  
## 254 8 10 7 3 3  
## 255 10 8 3 3 1  
## 256 4 10 3 6 1  
## 257 2 1 1 1 1  
## 258 2 1 2 1 1  
## 259 2 1 3 1 1  
## 260 5 8 3 4 1  
## 261 3 10 5 1 3  
## 262 10 10 10 6 5  
## 263 5 10 7 8 1  
## 264 6 10 5 5 1  
## 265 10 3 5 3 3  
## 266 2 1 3 2 1  
## 267 3 10 4 3 2  
## 268 3 10 7 1 1  
## 269 3 4 8 7 8  
## 270 2 1 3 1 1  
## 271 3 10 3 9 2  
## 272 2 1 3 1 1  
## 273 3 10 7 1 1  
## 274 3 4 3 3 1  
## 275 2 1 3 2 1  
## 277 2 1 2 1 1  
## 278 2 1 2 1 1  
## 279 2 1 3 1 1  
## 280 3 7 3 3 8  
## 281 2 1 3 1 1  
## 282 2 1 3 1 1  
## 283 4 10 5 6 1  
## 284 2 10 5 3 1  
## 285 2 10 3 8 2  
## 286 8 10 10 7 3  
## 287 10 10 4 10 10  
## 288 3 1 2 1 1  
## 289 4 5 5 10 1  
## 290 6 10 4 10 4  
## 291 2 1 1 1 1  
## 292 2 1 3 1 1  
## 294 2 10 2 3 1  
## 296 6 10 7 4 1  
## 297 4 5 4 7 1  
## 299 5 1 1 1 1  
## 300 4 10 7 7 2  
## 301 4 4 7 10 1  
## 302 2 1 3 1 1  
## 303 9 10 7 10 10  
## 304 2 1 3 1 1  
## 305 3 10 3 3 1  
## 306 4 10 3 10 4  
## 307 2 1 3 1 1  
## 308 2 1 3 1 1  
## 309 4 3 8 8 4  
## 310 2 5 5 1 1  
## 311 3 1 2 1 1  
## 312 2 1 1 1 1  
## 313 10 1 3 5 1  
## 314 2 1 1 1 1  
## 315 1 1 2 1 1  
## 317 5 10 4 3 1  
## 318 6 8 8 9 1  
## 319 5 1 3 1 1  
## 320 6 5 7 3 1  
## 321 5 10 7 4 6  
## 323 2 1 3 1 1  
## 324 2 10 4 1 1  
## 325 2 1 3 1 1  
## 326 2 1 2 3 1  
## 327 2 10 5 4 1  
## 328 2 1 2 1 1  
## 329 6 4 3 10 1  
## 330 5 10 7 1 1  
## 331 2 8 6 1 1  
## 332 2 1 3 1 2  
## 333 2 1 2 2 1  
## 334 4 10 4 3 1  
## 335 3 10 3 4 2  
## 336 2 1 1 1 1  
## 337 4 10 3 4 1  
## 338 2 1 3 1 1  
## 339 1 1 2 1 1  
## 340 2 10 4 3 1  
## 341 2 10 7 6 1  
## 342 2 1 3 1 1  
## 343 2 1 1 1 1  
## 344 2 1 1 1 1  
## 345 10 10 9 5 3  
## 346 2 1 1 1 1  
## 347 3 1 1 3 1  
## 348 1 1 1 3 1  
## 349 5 1 3 3 1  
## 350 3 8 7 6 1  
## 351 2 1 1 1 1  
## 352 2 1 3 1 1  
## 353 7 3 4 6 1  
## 354 7 10 4 9 4  
## 355 2 1 2 1 1  
## 356 3 1 2 2 1  
## 357 3 3 3 3 3  
## 358 10 10 7 3 8  
## 359 8 4 4 10 3  
## 360 3 7 3 5 3  
## 361 10 10 8 10 10  
## 362 6 10 5 1 4  
## 363 4 3 2 1 1  
## 364 2 3 2 1 1  
## 365 2 1 3 1 1  
## 366 2 1 2 1 1  
## 367 8 10 7 10 7  
## 368 5 10 8 10 3  
## 369 2 1 1 1 1  
## 370 1 1 2 1 1  
## 371 3 1 2 1 1  
## 372 2 1 1 1 1  
## 373 2 1 2 1 1  
## 374 2 1 2 1 1  
## 375 2 1 2 1 1  
## 376 2 1 1 1 1  
## 377 2 1 2 1 1  
## 378 1 1 2 1 1  
## 379 3 1 2 2 1  
## 380 4 1 3 1 1  
## 381 2 1 1 1 1  
## 382 4 10 7 8 4  
## 383 2 1 3 2 1  
## 384 2 1 1 1 1  
## 385 2 1 1 1 1  
## 386 3 1 1 2 3  
## 387 2 10 7 1 1  
## 388 3 1 3 1 1  
## 389 2 1 2 2 1  
## 390 3 2 2 2 1  
## 391 2 1 2 1 1  
## 392 3 10 7 9 1  
## 393 2 1 2 1 1  
## 394 1 1 1 1 1  
## 395 2 1 2 1 1  
## 396 2 1 2 1 1  
## 397 2 1 3 1 1  
## 398 2 1 1 1 1  
## 399 2 1 2 2 1  
## 400 2 1 1 1 1  
## 401 6 9 9 3 8  
## 402 2 1 1 1 1  
## 403 2 1 2 1 1  
## 404 2 4 1 1 1  
## 405 2 1 1 2 1  
## 406 2 1 2 1 1  
## 407 2 1 2 1 1  
## 408 2 1 2 1 1  
## 409 2 2 3 1 1  
## 410 2 1 2 1 1  
## 411 2 1 2 1 1  
## 413 8 4 8 5 1  
## 414 2 1 3 1 1  
## 415 3 10 6 6 1  
## 416 3 3 3 5 1  
## 417 10 10 7 2 1  
## 418 2 1 2 1 1  
## 419 2 2 3 2 2  
## 420 5 1 1 1 1  
## 421 2 3 3 1 1  
## 422 10 10 8 2 1  
## 423 2 1 3 3 1  
## 424 2 1 2 1 1  
## 425 2 1 1 1 1  
## 426 10 10 10 10 1  
## 427 2 1 1 1 1  
## 428 4 2 5 10 1  
## 429 2 1 2 1 1  
## 430 2 1 2 1 1  
## 431 2 1 2 2 1  
## 432 4 1 3 2 1  
## 433 2 1 2 2 1  
## 434 2 1 1 1 1  
## 435 5 8 4 2 1  
## 436 3 10 5 1 1  
## 437 6 1 2 8 1  
## 438 2 1 1 1 1  
## 439 2 1 1 1 1  
## 440 2 1 1 1 1  
## 441 4 10 10 1 1  
## 442 2 4 1 1 1  
## 443 2 3 1 1 1  
## 444 2 2 1 1 1  
## 445 3 1 2 1 1  
## 446 2 1 1 1 1  
## 447 2 1 1 1 1  
## 448 2 1 1 1 1  
## 449 1 1 1 1 1  
## 450 6 10 8 10 1  
## 451 1 1 2 1 1  
## 452 2 1 1 1 1  
## 453 2 1 1 1 1  
## 454 6 10 10 7 1  
## 455 3 1 1 1 1  
## 456 2 6 1 1 2  
## 457 5 10 8 6 1  
## 458 6 3 10 10 1  
## 459 2 1 1 1 1  
## 460 2 1 1 1 1  
## 461 2 1 1 1 1  
## 462 2 5 1 1 1  
## 463 2 1 1 1 1  
## 464 2 1 1 2 1  
## 465 2 1 1 1 1  
## 466 6 4 7 10 3  
## 467 4 10 9 7 1  
## 468 4 10 7 6 2  
## 469 2 1 1 1 1  
## 470 2 1 2 1 1  
## 471 1 1 2 1 1  
## 472 2 1 1 1 1  
## 473 1 1 1 1 1  
## 474 2 1 1 1 1  
## 475 2 1 1 1 1  
## 476 2 1 1 1 1  
## 477 2 1 1 1 1  
## 478 2 1 1 1 1  
## 479 2 1 1 1 1  
## 480 4 10 7 5 1  
## 481 1 1 1 1 1  
## 482 2 1 1 1 1  
## 483 10 5 10 10 10  
## 484 5 10 9 10 1  
## 485 2 1 1 1 1  
## 486 1 3 1 1 1  
## 487 1 1 2 1 1  
## 488 6 10 8 1 5  
## 489 3 3 3 4 1  
## 490 3 4 4 1 1  
## 491 2 1 1 1 1  
## 492 3 10 7 1 1  
## 493 1 1 2 1 1  
## 494 6 10 6 5 2  
## 495 4 5 2 1 1  
## 496 1 1 2 1 1  
## 497 1 1 1 1 1  
## 498 2 1 1 1 1  
## 499 2 1 2 1 1  
## 500 2 1 2 1 1  
## 501 2 1 3 1 1  
## 502 2 1 2 1 1  
## 503 2 1 2 1 1  
## 504 2 1 3 1 1  
## 505 2 1 1 1 1  
## 506 2 1 1 1 1  
## 507 7 5 4 8 7  
## 508 2 4 1 1 1  
## 509 2 1 1 1 1  
## 510 2 1 1 1 1  
## 511 2 1 1 1 1  
## 512 2 1 2 1 1  
## 513 2 1 1 1 1  
## 514 1 1 2 1 1  
## 515 3 10 8 10 2  
## 516 3 10 9 10 1  
## 517 1 1 1 1 1  
## 518 1 1 2 1 1  
## 519 2 1 1 1 1  
## 520 4 10 9 1 1  
## 521 3 1 1 1 1  
## 522 3 1 1 1 1  
## 523 3 5 7 3 1  
## 524 4 10 5 3 1  
## 525 2 1 2 1 1  
## 526 2 1 1 1 1  
## 527 2 1 1 1 1  
## 528 2 1 3 1 1  
## 529 2 1 1 1 1  
## 530 1 1 2 1 1  
## 531 4 10 6 9 1  
## 532 2 1 2 1 1  
## 533 1 1 3 1 1  
## 534 2 1 2 1 1  
## 535 2 1 2 1 1  
## 536 2 1 3 1 1  
## 537 2 1 3 1 1  
## 538 2 1 3 1 1  
## 539 2 1 2 1 1  
## 540 2 1 2 1 1  
## 541 2 2 2 1 1  
## 542 2 1 1 1 1  
## 543 2 1 1 1 1  
## 544 2 1 2 1 1  
## 545 2 1 2 1 1  
## 546 2 1 2 1 1  
## 547 4 10 7 10 1  
## 548 1 1 1 1 1  
## 549 1 1 1 1 1  
## 550 4 5 7 8 2  
## 551 2 1 2 1 1  
## 552 2 1 3 1 1  
## 553 2 1 4 2 1  
## 554 2 5 2 1 2  
## 555 2 1 1 1 1  
## 556 2 1 4 8 1  
## 557 1 1 2 1 1  
## 558 2 1 1 1 1  
## 559 2 1 2 1 1  
## 560 2 1 2 1 1  
## 561 2 1 3 1 1  
## 562 2 1 3 1 1  
## 563 2 1 3 1 1  
## 564 2 1 2 1 1  
## 565 2 1 3 2 1  
## 566 5 10 10 10 1  
## 567 2 1 3 1 1  
## 568 2 3 2 1 1  
## 569 6 10 2 5 2  
## 570 6 5 10 3 1  
## 571 8 10 8 2 1  
## 572 3 10 9 10 2  
## 573 2 1 2 1 1  
## 574 2 1 2 1 1  
## 575 4 2 7 7 1  
## 576 2 1 3 1 1  
## 577 2 1 2 1 1  
## 578 2 1 2 1 1  
## 579 2 1 2 1 1  
## 580 2 1 3 1 1  
## 581 2 1 2 1 1  
## 582 5 10 7 5 1  
## 583 4 10 6 10 1  
## 584 2 1 1 1 1  
## 585 3 1 1 1 1  
## 586 2 1 1 1 1  
## 587 6 10 10 10 1  
## 588 2 1 2 2 1  
## 589 6 3 4 1 1  
## 590 2 1 1 1 1  
## 591 4 1 10 1 1  
## 592 4 10 7 6 1  
## 593 3 10 4 1 1  
## 594 2 1 1 1 1  
## 595 4 10 7 1 1  
## 596 2 1 2 1 1  
## 597 2 1 2 1 1  
## 598 2 1 3 1 1  
## 599 2 1 2 1 1  
## 600 1 1 1 1 1  
## 601 2 1 2 1 1  
## 602 1 1 2 1 1  
## 603 2 1 2 1 1  
## 604 4 1 8 10 1  
## 605 5 10 8 1 2  
## 606 5 8 7 8 3  
## 607 2 1 1 1 1  
## 608 2 1 1 1 1  
## 609 10 10 10 1 1  
## 610 2 1 1 1 1  
## 611 3 10 7 1 2  
## 612 5 2 8 5 1  
## 613 6 10 10 10 10  
## 614 2 1 2 1 1  
## 615 1 1 2 1 1  
## 616 2 1 2 1 1  
## 617 2 1 2 1 1  
## 619 2 1 2 1 1  
## 620 2 1 2 1 1  
## 621 2 1 2 1 1  
## 622 3 2 6 1 1  
## 623 2 1 2 1 1  
## 624 2 1 1 1 1  
## 625 1 1 2 1 1  
## 626 3 4 1 1 1  
## 627 7 6 7 7 3  
## 628 2 5 1 1 1  
## 629 2 1 1 1 1  
## 630 2 1 1 1 1  
## 631 2 1 1 1 1  
## 632 2 1 2 1 1  
## 633 2 1 1 1 1  
## 634 5 3 5 10 1  
## 635 2 1 1 1 1  
## 636 2 1 1 1 1  
## 637 7 1 10 10 3  
## 638 2 2 2 1 1  
## 639 2 1 1 1 1  
## 640 2 1 1 1 1  
## 641 2 1 1 1 1  
## 642 2 1 2 1 1  
## 643 2 1 2 1 1  
## 644 2 1 1 1 1  
## 645 2 1 1 1 1  
## 646 2 1 2 1 1  
## 647 2 1 1 1 1  
## 648 2 1 1 1 1  
## 649 10 2 10 10 10  
## 650 2 1 2 1 1  
## 651 3 4 1 1 1  
## 652 2 1 2 1 1  
## 653 2 1 2 2 1  
## 654 2 1 2 1 1  
## 655 2 1 3 1 1  
## 656 2 1 2 1 1  
## 657 2 1 2 1 1  
## 658 8 1 3 6 1  
## 659 3 10 7 2 3  
## 660 2 1 1 1 1  
## 661 2 1 2 1 1  
## 662 2 1 3 1 1  
## 663 2 1 2 1 1  
## 664 2 1 2 1 1  
## 665 2 1 2 1 1  
## 666 2 1 1 1 1  
## 667 2 1 1 1 2  
## 668 2 1 3 1 1  
## 669 6 1 7 10 3  
## 670 5 5 7 10 1  
## 671 5 8 7 4 1  
## 672 2 1 3 1 1  
## 673 2 1 3 1 1  
## 674 3 1 1 1 1  
## 675 2 1 2 1 1  
## 676 2 1 1 1 1  
## 677 2 1 2 1 1  
## 678 2 1 1 1 1  
## 679 2 1 1 1 1  
## 680 2 1 1 1 1  
## 681 5 10 10 10 7  
## 682 4 10 5 6 3  
## 683 2 1 3 2 1  
## 684 2 1 1 1 1  
## 685 2 1 1 1 1  
## 686 2 1 1 1 1  
## 687 2 1 1 1 1  
## 688 2 1 2 3 1  
## 689 2 1 1 1 1  
## 690 2 1 1 1 8  
## 691 2 1 1 1 1  
## 692 4 5 4 4 1  
## 693 2 1 1 1 1  
## 694 2 1 2 1 2  
## 695 3 2 1 1 1  
## 696 2 1 1 1 1  
## 697 7 3 8 10 2  
## 698 3 4 10 6 1  
## 699 4 5 10 4 1  
## Class  
## 1 benign  
## 2 benign  
## 3 benign  
## 4 benign  
## 5 benign  
## 6 malignant  
## 7 benign  
## 8 benign  
## 9 benign  
## 10 benign  
## 11 benign  
## 12 benign  
## 13 malignant  
## 14 benign  
## 15 malignant  
## 16 malignant  
## 17 benign  
## 18 benign  
## 19 malignant  
## 20 benign  
## 21 malignant  
## 22 malignant  
## 23 benign  
## 25 benign  
## 26 malignant  
## 27 benign  
## 28 benign  
## 29 benign  
## 30 benign  
## 31 benign  
## 32 benign  
## 33 malignant  
## 34 benign  
## 35 benign  
## 36 benign  
## 37 malignant  
## 38 benign  
## 39 malignant  
## 40 malignant  
## 42 malignant  
## 43 malignant  
## 44 malignant  
## 45 malignant  
## 46 benign  
## 47 malignant  
## 48 benign  
## 49 benign  
## 50 malignant  
## 51 malignant  
## 52 malignant  
## 53 malignant  
## 54 malignant  
## 55 malignant  
## 56 malignant  
## 57 malignant  
## 58 malignant  
## 59 malignant  
## 60 malignant  
## 61 malignant  
## 62 benign  
## 63 malignant  
## 64 malignant  
## 65 benign  
## 66 malignant  
## 67 benign  
## 68 malignant  
## 69 malignant  
## 70 benign  
## 71 benign  
## 72 malignant  
## 73 benign  
## 74 malignant  
## 75 malignant  
## 76 benign  
## 77 benign  
## 78 benign  
## 79 benign  
## 80 benign  
## 81 benign  
## 82 benign  
## 83 benign  
## 84 benign  
## 85 malignant  
## 86 malignant  
## 87 malignant  
## 88 malignant  
## 89 benign  
## 90 benign  
## 91 benign  
## 92 benign  
## 93 benign  
## 94 benign  
## 95 benign  
## 96 benign  
## 97 benign  
## 98 benign  
## 99 malignant  
## 100 malignant  
## 101 malignant  
## 102 malignant  
## 103 benign  
## 104 malignant  
## 105 malignant  
## 106 malignant  
## 107 malignant  
## 108 malignant  
## 109 benign  
## 110 malignant  
## 111 benign  
## 112 malignant  
## 113 malignant  
## 114 malignant  
## 115 benign  
## 116 benign  
## 117 benign  
## 118 malignant  
## 119 benign  
## 120 benign  
## 121 benign  
## 122 benign  
## 123 malignant  
## 124 malignant  
## 125 malignant  
## 126 benign  
## 127 malignant  
## 128 benign  
## 129 malignant  
## 130 benign  
## 131 benign  
## 132 benign  
## 133 malignant  
## 134 benign  
## 135 benign  
## 136 benign  
## 137 benign  
## 138 benign  
## 139 benign  
## 141 benign  
## 142 benign  
## 143 malignant  
## 144 benign  
## 145 benign  
## 147 malignant  
## 148 benign  
## 149 benign  
## 150 malignant  
## 151 benign  
## 152 malignant  
## 153 malignant  
## 154 benign  
## 155 benign  
## 156 malignant  
## 157 benign  
## 158 benign  
## 160 malignant  
## 161 malignant  
## 162 benign  
## 163 benign  
## 164 benign  
## 166 benign  
## 167 malignant  
## 168 malignant  
## 169 benign  
## 170 benign  
## 171 benign  
## 172 benign  
## 173 benign  
## 174 malignant  
## 175 malignant  
## 176 malignant  
## 177 benign  
## 178 malignant  
## 179 benign  
## 180 malignant  
## 181 benign  
## 182 benign  
## 183 benign  
## 184 malignant  
## 185 malignant  
## 186 benign  
## 187 malignant  
## 188 malignant  
## 189 malignant  
## 190 benign  
## 191 malignant  
## 192 malignant  
## 193 benign  
## 194 benign  
## 195 benign  
## 196 benign  
## 197 benign  
## 198 benign  
## 199 benign  
## 200 benign  
## 201 malignant  
## 202 malignant  
## 203 benign  
## 204 benign  
## 205 benign  
## 206 malignant  
## 207 malignant  
## 208 benign  
## 209 benign  
## 210 benign  
## 211 malignant  
## 212 malignant  
## 213 benign  
## 214 malignant  
## 215 malignant  
## 216 malignant  
## 217 benign  
## 218 benign  
## 219 malignant  
## 220 benign  
## 221 benign  
## 222 malignant  
## 223 malignant  
## 224 malignant  
## 225 malignant  
## 226 benign  
## 227 malignant  
## 228 malignant  
## 229 benign  
## 230 malignant  
## 231 malignant  
## 232 malignant  
## 233 benign  
## 234 malignant  
## 235 benign  
## 237 malignant  
## 238 malignant  
## 239 malignant  
## 240 malignant  
## 241 benign  
## 242 benign  
## 243 benign  
## 244 benign  
## 245 benign  
## 246 benign  
## 247 malignant  
## 248 malignant  
## 249 benign  
## 251 benign  
## 252 malignant  
## 253 benign  
## 254 malignant  
## 255 malignant  
## 256 malignant  
## 257 benign  
## 258 benign  
## 259 benign  
## 260 benign  
## 261 malignant  
## 262 malignant  
## 263 malignant  
## 264 malignant  
## 265 malignant  
## 266 benign  
## 267 malignant  
## 268 malignant  
## 269 malignant  
## 270 benign  
## 271 malignant  
## 272 benign  
## 273 malignant  
## 274 malignant  
## 275 benign  
## 277 benign  
## 278 benign  
## 279 benign  
## 280 malignant  
## 281 benign  
## 282 benign  
## 283 malignant  
## 284 malignant  
## 285 malignant  
## 286 malignant  
## 287 malignant  
## 288 benign  
## 289 malignant  
## 290 malignant  
## 291 benign  
## 292 benign  
## 294 malignant  
## 296 malignant  
## 297 benign  
## 299 benign  
## 300 malignant  
## 301 malignant  
## 302 benign  
## 303 malignant  
## 304 benign  
## 305 malignant  
## 306 malignant  
## 307 benign  
## 308 benign  
## 309 malignant  
## 310 benign  
## 311 benign  
## 312 benign  
## 313 malignant  
## 314 benign  
## 315 benign  
## 317 malignant  
## 318 malignant  
## 319 benign  
## 320 benign  
## 321 malignant  
## 323 benign  
## 324 malignant  
## 325 benign  
## 326 benign  
## 327 malignant  
## 328 benign  
## 329 malignant  
## 330 malignant  
## 331 malignant  
## 332 benign  
## 333 benign  
## 334 malignant  
## 335 malignant  
## 336 benign  
## 337 malignant  
## 338 benign  
## 339 benign  
## 340 malignant  
## 341 malignant  
## 342 benign  
## 343 benign  
## 344 benign  
## 345 malignant  
## 346 benign  
## 347 benign  
## 348 benign  
## 349 malignant  
## 350 malignant  
## 351 benign  
## 352 benign  
## 353 benign  
## 354 malignant  
## 355 benign  
## 356 benign  
## 357 malignant  
## 358 malignant  
## 359 malignant  
## 360 malignant  
## 361 malignant  
## 362 malignant  
## 363 benign  
## 364 benign  
## 365 benign  
## 366 benign  
## 367 malignant  
## 368 malignant  
## 369 benign  
## 370 benign  
## 371 benign  
## 372 benign  
## 373 benign  
## 374 benign  
## 375 benign  
## 376 benign  
## 377 benign  
## 378 benign  
## 379 benign  
## 380 benign  
## 381 benign  
## 382 malignant  
## 383 benign  
## 384 benign  
## 385 benign  
## 386 benign  
## 387 malignant  
## 388 benign  
## 389 benign  
## 390 benign  
## 391 benign  
## 392 malignant  
## 393 benign  
## 394 benign  
## 395 benign  
## 396 benign  
## 397 benign  
## 398 benign  
## 399 benign  
## 400 benign  
## 401 malignant  
## 402 benign  
## 403 benign  
## 404 benign  
## 405 benign  
## 406 benign  
## 407 benign  
## 408 benign  
## 409 benign  
## 410 benign  
## 411 benign  
## 413 malignant  
## 414 benign  
## 415 malignant  
## 416 benign  
## 417 malignant  
## 418 benign  
## 419 benign  
## 420 benign  
## 421 benign  
## 422 malignant  
## 423 benign  
## 424 benign  
## 425 benign  
## 426 malignant  
## 427 benign  
## 428 malignant  
## 429 benign  
## 430 benign  
## 431 benign  
## 432 benign  
## 433 benign  
## 434 benign  
## 435 benign  
## 436 malignant  
## 437 malignant  
## 438 benign  
## 439 benign  
## 440 benign  
## 441 malignant  
## 442 benign  
## 443 benign  
## 444 benign  
## 445 benign  
## 446 benign  
## 447 benign  
## 448 benign  
## 449 benign  
## 450 malignant  
## 451 benign  
## 452 benign  
## 453 benign  
## 454 malignant  
## 455 benign  
## 456 malignant  
## 457 malignant  
## 458 malignant  
## 459 benign  
## 460 benign  
## 461 benign  
## 462 benign  
## 463 benign  
## 464 benign  
## 465 benign  
## 466 malignant  
## 467 malignant  
## 468 malignant  
## 469 benign  
## 470 benign  
## 471 benign  
## 472 benign  
## 473 benign  
## 474 benign  
## 475 benign  
## 476 benign  
## 477 benign  
## 478 benign  
## 479 benign  
## 480 malignant  
## 481 benign  
## 482 benign  
## 483 malignant  
## 484 malignant  
## 485 benign  
## 486 benign  
## 487 benign  
## 488 malignant  
## 489 malignant  
## 490 malignant  
## 491 benign  
## 492 malignant  
## 493 benign  
## 494 malignant  
## 495 benign  
## 496 benign  
## 497 benign  
## 498 benign  
## 499 benign  
## 500 benign  
## 501 benign  
## 502 benign  
## 503 benign  
## 504 benign  
## 505 benign  
## 506 benign  
## 507 malignant  
## 508 benign  
## 509 benign  
## 510 benign  
## 511 benign  
## 512 benign  
## 513 benign  
## 514 benign  
## 515 malignant  
## 516 malignant  
## 517 benign  
## 518 benign  
## 519 benign  
## 520 malignant  
## 521 benign  
## 522 benign  
## 523 malignant  
## 524 malignant  
## 525 benign  
## 526 benign  
## 527 benign  
## 528 benign  
## 529 benign  
## 530 benign  
## 531 malignant  
## 532 benign  
## 533 benign  
## 534 benign  
## 535 benign  
## 536 benign  
## 537 benign  
## 538 benign  
## 539 benign  
## 540 benign  
## 541 benign  
## 542 benign  
## 543 benign  
## 544 benign  
## 545 benign  
## 546 benign  
## 547 malignant  
## 548 benign  
## 549 benign  
## 550 malignant  
## 551 benign  
## 552 benign  
## 553 benign  
## 554 benign  
## 555 benign  
## 556 benign  
## 557 benign  
## 558 benign  
## 559 benign  
## 560 benign  
## 561 benign  
## 562 benign  
## 563 benign  
## 564 benign  
## 565 benign  
## 566 malignant  
## 567 benign  
## 568 benign  
## 569 malignant  
## 570 malignant  
## 571 malignant  
## 572 malignant  
## 573 benign  
## 574 benign  
## 575 malignant  
## 576 benign  
## 577 benign  
## 578 benign  
## 579 benign  
## 580 benign  
## 581 benign  
## 582 malignant  
## 583 malignant  
## 584 benign  
## 585 benign  
## 586 benign  
## 587 malignant  
## 588 benign  
## 589 malignant  
## 590 benign  
## 591 malignant  
## 592 malignant  
## 593 malignant  
## 594 benign  
## 595 malignant  
## 596 benign  
## 597 benign  
## 598 benign  
## 599 benign  
## 600 benign  
## 601 benign  
## 602 benign  
## 603 benign  
## 604 malignant  
## 605 malignant  
## 606 malignant  
## 607 benign  
## 608 benign  
## 609 malignant  
## 610 benign  
## 611 malignant  
## 612 malignant  
## 613 malignant  
## 614 benign  
## 615 benign  
## 616 benign  
## 617 benign  
## 619 benign  
## 620 benign  
## 621 benign  
## 622 benign  
## 623 benign  
## 624 benign  
## 625 benign  
## 626 benign  
## 627 malignant  
## 628 benign  
## 629 benign  
## 630 benign  
## 631 benign  
## 632 benign  
## 633 benign  
## 634 malignant  
## 635 benign  
## 636 benign  
## 637 malignant  
## 638 benign  
## 639 benign  
## 640 benign  
## 641 benign  
## 642 benign  
## 643 benign  
## 644 benign  
## 645 benign  
## 646 benign  
## 647 benign  
## 648 benign  
## 649 malignant  
## 650 benign  
## 651 benign  
## 652 benign  
## 653 benign  
## 654 benign  
## 655 benign  
## 656 benign  
## 657 benign  
## 658 benign  
## 659 malignant  
## 660 benign  
## 661 benign  
## 662 benign  
## 663 benign  
## 664 benign  
## 665 benign  
## 666 benign  
## 667 benign  
## 668 benign  
## 669 malignant  
## 670 malignant  
## 671 malignant  
## 672 benign  
## 673 benign  
## 674 benign  
## 675 benign  
## 676 benign  
## 677 benign  
## 678 benign  
## 679 benign  
## 680 benign  
## 681 malignant  
## 682 malignant  
## 683 benign  
## 684 benign  
## 685 benign  
## 686 benign  
## 687 benign  
## 688 benign  
## 689 benign  
## 690 benign  
## 691 benign  
## 692 malignant  
## 693 benign  
## 694 benign  
## 695 benign  
## 696 benign  
## 697 malignant  
## 698 malignant  
## 699 malignant

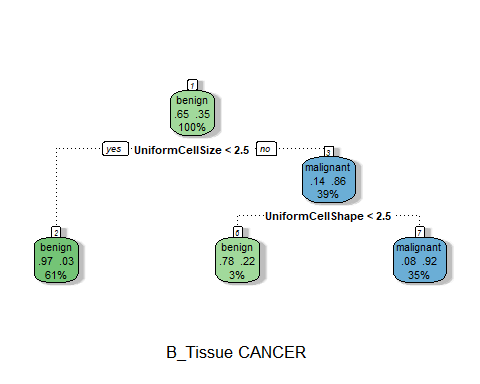
# 1a. Apply decision tree learning (use rpart) to the data to predict breast cancer malignancy (Class) and report the accuracy using 10-fold cross validation.

train\_control = trainControl(method = "cv", number = 10)  
tree1 <- train(Class ~., data = cncr\_IDNA, method = "rpart", trControl = train\_control)  
# Evaluate fit  
tree1

## CART   
##   
## 683 samples  
## 9 predictor  
## 2 classes: 'benign', 'malignant'   
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 615, 615, 615, 614, 615, 616, ...   
## Resampling results across tuning parameters:  
##   
## cp Accuracy Kappa   
## 0.02510460 0.9400237 0.8691296  
## 0.05439331 0.9297295 0.8484586  
## 0.79079498 0.8199883 0.5341749  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final value used for the model was cp = 0.0251046.

# 1b. Generate a visualization of the decision tree.

fancyRpartPlot(tree1$finalModel, caption = "B\_Tissue CANCER")



**1c. Generate the full set of rules using IF-THEN statements.**

We can create one rule per leaf as below;

IF UniformCellSize = “<2.5” THEN Class = “benign” IF UniformCellSize = “>=2.5” AND UniformCellShape = “<2.5” THEN Class = “benign” IF UniformCellSize = “>=2.5” AND UniformCellShape = “>=2.5” THEN Class = “malignant”

# Problem 2 (15 points):

# In this problem you will generate decision trees with a set of parameters. You will be using the storms data, a subset of the NOAA Atlantic hurricane database2, which includes the positions and attributes of 198 tropical storms (potential hurricanes), measured every six hours during the lifetime of a storm. It is part of the dplyr library, so load the library and you will be able to access it.

library(dplyr)  
data("storms")  
head(storms)

## # A tibble: 6 × 13  
## name year month day hour lat long status categ…¹ wind press…² tropi…³  
## <chr> <dbl> <dbl> <int> <dbl> <dbl> <dbl> <chr> <ord> <int> <int> <int>  
## 1 Amy 1975 6 27 0 27.5 -79 tropi… -1 25 1013 NA  
## 2 Amy 1975 6 27 6 28.5 -79 tropi… -1 25 1013 NA  
## 3 Amy 1975 6 27 12 29.5 -79 tropi… -1 25 1013 NA  
## 4 Amy 1975 6 27 18 30.5 -79 tropi… -1 25 1013 NA  
## 5 Amy 1975 6 28 0 31.5 -78.8 tropi… -1 25 1012 NA  
## 6 Amy 1975 6 28 6 32.4 -78.7 tropi… -1 25 1012 NA  
## # … with 1 more variable: hurricane\_force\_diameter <int>, and abbreviated  
## # variable names ¹​category, ²​pressure, ³​tropicalstorm\_force\_diameter

# As a preprocessing step, view the data and makesure the target variable (category) is converted to a factor (as opposed to character string).

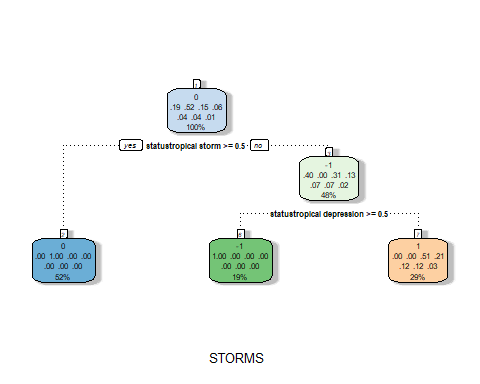
# Remove NA  
stormsNA <- na.omit(storms)  
# Converting the target variable (category) to a factor (as opposed to character string)  
 stormsNA$category <- as.factor(stormsNA$category)

# 2a. Build a decision tree using the following hyperparameters, maxdepth=2, minsplit=5 and minbucket=3. Be careful to use the right method of training so that you are not automatically tuning the cp parameter, but you are controlling the aforementioned parameters specifically. Use cross validation to report your accuracy score. These parameters will result in a relatively small tree.

train\_control = trainControl(method = "cv", number = 10)  
# Setting hyper parameters  
hypers = rpart.control(minsplit = 5, maxdepth = 2, minbucket = 3)  
# Fitting the model  
treeA <- train(category ~., data = stormsNA, control = hypers, trControl = train\_control, method = "rpart1SE")  
treeA

## CART   
##   
## 5350 samples  
## 12 predictor  
## 7 classes: '-1', '0', '1', '2', '3', '4', '5'   
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 4815, 4817, 4816, 4812, 4814, 4815, ...   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.8594419 0.7842303

# Visualize with Decision Tree  
fancyRpartPlot(treeA$finalModel, caption = "STORMS")



**b. To see how this performed with respect to the individual classes, we could use a confusion matrix.**

# Confusion Matrix for fit evaluation  
pred\_treeA <- predict(treeA, stormsNA)  
# Confusion Matrix  
confusionMatrix(stormsNA$category, pred\_treeA)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction -1 0 1 2 3 4 5  
## -1 1031 0 0 0 0 0 0  
## 0 0 2771 0 0 0 0 0  
## 1 0 1 796 0 0 0 0  
## 2 0 0 323 0 0 0 0  
## 3 0 0 190 0 0 0 0  
## 4 0 0 191 0 0 0 0  
## 5 0 0 47 0 0 0 0  
##   
## Overall Statistics  
##   
## Accuracy : 0.8594   
## 95% CI : (0.8498, 0.8686)  
## No Information Rate : 0.5181   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.7842   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: -1 Class: 0 Class: 1 Class: 2 Class: 3 Class: 4  
## Sensitivity 1.0000 0.9996 0.5145 NA NA NA  
## Specificity 1.0000 1.0000 0.9997 0.93963 0.96449 0.9643  
## Pos Pred Value 1.0000 1.0000 0.9987 NA NA NA  
## Neg Pred Value 1.0000 0.9996 0.8351 NA NA NA  
## Prevalence 0.1927 0.5181 0.2892 0.00000 0.00000 0.0000  
## Detection Rate 0.1927 0.5179 0.1488 0.00000 0.00000 0.0000  
## Detection Prevalence 0.1927 0.5179 0.1490 0.06037 0.03551 0.0357  
## Balanced Accuracy 1.0000 0.9998 0.7571 NA NA NA  
## Class: 5  
## Sensitivity NA  
## Specificity 0.991215  
## Pos Pred Value NA  
## Neg Pred Value NA  
## Prevalence 0.000000  
## Detection Rate 0.000000  
## Detection Prevalence 0.008785  
## Balanced Accuracy NA

# We also want to see if that aspect of performance is different on the train versus the test set. Create a train/test partition. Train on the training set. By making predictions with that model on the train set and on the test set separately, use the outputs to create two separate confusion matrices, one for each partition. Remember, we are testing if the model built with the training data performs differently on data used to train it (train set) as opposed to new data (test set).

**Q) Compare the confusion matrices and report which classes it has problem classifying. Do you think that both are performing similarly and what does that suggest about overfitting for the model?**

# Test vs Train Performances

# Partitioning the data  
index = createDataPartition( y= stormsNA$category, p=0.7, list=FALSE)  
# Everything in the generated index list  
train\_set = stormsNA[index,]  
# Everything except the generated indices  
test\_set = stormsNA[-index,]

# Training Set

# Fit the model  
tree1 <- train(category ~., data = train\_set, method = "rpart1SE", trControl = train\_control)  
# Evaluate the fit with a confusion matrix  
pred\_tree\_train <- predict(tree1, train\_set)  
# Confusion Matrix  
confusionMatrix(train\_set$category, pred\_tree\_train)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction -1 0 1 2 3 4 5  
## -1 722 0 0 0 0 0 0  
## 0 0 1940 0 0 0 0 0  
## 1 0 1 557 0 0 0 0  
## 2 0 0 0 227 0 0 0  
## 3 0 0 0 0 133 0 0  
## 4 0 0 0 0 0 134 0  
## 5 0 0 0 0 0 0 33  
##   
## Overall Statistics  
##   
## Accuracy : 0.9997   
## 95% CI : (0.9985, 1)  
## No Information Rate : 0.518   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9996   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: -1 Class: 0 Class: 1 Class: 2 Class: 3 Class: 4  
## Sensitivity 1.0000 0.9995 1.0000 1.00000 1.0000 1.00000  
## Specificity 1.0000 1.0000 0.9997 1.00000 1.0000 1.00000  
## Pos Pred Value 1.0000 1.0000 0.9982 1.00000 1.0000 1.00000  
## Neg Pred Value 1.0000 0.9994 1.0000 1.00000 1.0000 1.00000  
## Prevalence 0.1927 0.5180 0.1487 0.06058 0.0355 0.03576  
## Detection Rate 0.1927 0.5177 0.1487 0.06058 0.0355 0.03576  
## Detection Prevalence 0.1927 0.5177 0.1489 0.06058 0.0355 0.03576  
## Balanced Accuracy 1.0000 0.9997 0.9998 1.00000 1.0000 1.00000  
## Class: 5  
## Sensitivity 1.000000  
## Specificity 1.000000  
## Pos Pred Value 1.000000  
## Neg Pred Value 1.000000  
## Prevalence 0.008807  
## Detection Rate 0.008807  
## Detection Prevalence 0.008807  
## Balanced Accuracy 1.000000

# Test Set

# Evaluate the fit with a confusion matrix  
pred\_tree\_test <- predict(tree1, test\_set)  
# Confusion Matrix  
confusionMatrix(test\_set$category, pred\_tree\_test)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction -1 0 1 2 3 4 5  
## -1 309 0 0 0 0 0 0  
## 0 0 831 0 0 0 0 0  
## 1 0 0 239 0 0 0 0  
## 2 0 0 0 96 0 0 0  
## 3 0 0 0 0 57 0 0  
## 4 0 0 0 0 0 57 0  
## 5 0 0 0 0 0 0 14  
##   
## Overall Statistics  
##   
## Accuracy : 1   
## 95% CI : (0.9977, 1)  
## No Information Rate : 0.5184   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 1   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: -1 Class: 0 Class: 1 Class: 2 Class: 3 Class: 4  
## Sensitivity 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Specificity 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Pos Pred Value 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Neg Pred Value 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Prevalence 0.1928 0.5184 0.1491 0.05989 0.03556 0.03556  
## Detection Rate 0.1928 0.5184 0.1491 0.05989 0.03556 0.03556  
## Detection Prevalence 0.1928 0.5184 0.1491 0.05989 0.03556 0.03556  
## Balanced Accuracy 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Class: 5  
## Sensitivity 1.000000  
## Specificity 1.000000  
## Pos Pred Value 1.000000  
## Neg Pred Value 1.000000  
## Prevalence 0.008734  
## Detection Rate 0.008734  
## Detection Prevalence 0.008734  
## Balanced Accuracy 1.000000

# Inference 2b)

Ccomparing the confusion matrices of training and testing data, the accuracy of both the models are almost same (training Accuracy : 0.9997) , (test Accuracy : 1) However, classes 2-5 have the problem classifying themselves in the both matrices. As the accuracy of both training and testing data are almost the same Therefore ,model will not be overfitting.

# Problem 3 (15 points):

# This is will be an extension of Problem 2, using the same data and class. Here you will build many decision trees, manually tuning the parameters to gain intuition about the tradeoffs and how these tree parameters affect the complexity and quality of the model. The goal is to find the best tree model, which means it should be accurate but not too complex that the model overfits the training data. We will achieve this by using multiple sets of parameters and creating a graph of accuracy versus complexity for the training and the test sets (refer # to the tutorial). This problem may require a significant amount of effort because you will need to train a # substantial number of trees (at least 10).

# a. Partition your data into 80% for training and 20% for the test data set.

# Partition the data  
index3 = createDataPartition(y=stormsNA$category, p=0.8, list=FALSE)  
# Everything in the generated index list  
train\_set3 = stormsNA[index3,]  
# Everything except the generated indices  
test\_set3 = stormsNA[-index3,]

# b. Train at least 10 trees using different sets of parameters, through you made need more.

# Create the graph described above such that you can identify the inflection point where the tree is overfitting and pick a high-quality decision tree.

# Initialize cross validation  
train\_control = trainControl(method = "cv", number = 10)

# Tree 01

hypers = rpart.control(minsplit = 2, maxdepth = 1, minbucket = 2)  
tree01 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr01 <- predict(tree01, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr01)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst01 <- predict(tree01, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst01)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree01$finalModel$frame)  
  
# Form the table  
comp\_tbl <- data.frame("Nodes" = nodes, "TrainAccuracy" = a\_train, "TestAccuracy" = a\_test,  
 "MaxDepth" = 1, "Minsplit" = 2, "Minbucket" = 2)  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.710435 0.7111666 1 2 2

# Tree 2

hypers = rpart.control(minsplit = 5, maxdepth = 2, minbucket = 5)  
tree02 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr02 <- predict(tree02, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr02)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst02 <- predict(tree02, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst02)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree02$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 2, 5, 5))  
 comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5

# Tree 3

hypers = rpart.control(minsplit = 10, maxdepth = 3, minbucket = 10)  
tree03 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr03 <- predict(tree03, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr03)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst03 <- predict(tree03, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst03)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree03$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 3, 10, 10))  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10

# Tree 4

hypers = rpart.control(minsplit = 50, maxdepth = 4, minbucket = 50)  
tree04 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr04 <- predict(tree04, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr04)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst04 <- predict(tree04, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst04)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree04$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 4, 50, 50))  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50

# Tree 5

hypers = rpart.control(minsplit = 100, maxdepth = 6, minbucket = 100)  
tree05 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr05 <- predict(tree05, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr05)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst05 <- predict(tree05, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst05)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree05$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 6, 100, 100))  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100

# Tree 6

hypers = rpart.control(minsplit = 500, maxdepth = 8, minbucket = 500)  
tree06 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr06 <- predict(tree06, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr06)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst06 <- predict(tree06, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst06)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree06$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 8, 500, 500))  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100  
## 14 7 0.9196691 0.9201497 8 500 500

# Tree 7

hypers = rpart.control(minsplit = 1000, maxdepth = 4, minbucket = 1000)  
tree07 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr07 <- predict(tree07, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr07)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst07 <- predict(tree07, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst07)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree07$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 4, 1000, 1000))  
 comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100  
## 14 7 0.9196691 0.9201497 8 500 500  
## 15 3 0.7104350 0.7111666 4 1000 1000

# Tree 8

hypers = rpart.control(minsplit = 2500, maxdepth = 10, minbucket = 2500)  
tree08 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr08 <- predict(tree08, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr08)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst08 <- predict(tree08, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst08)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree08$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 10, 2500, 2500))  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100  
## 14 7 0.9196691 0.9201497 8 500 500  
## 15 3 0.7104350 0.7111666 4 1000 1000  
## 16 1 0.5177475 0.5184030 10 2500 2500

# Tree 9

hypers = rpart.control(minsplit = 5000, maxdepth = 25, minbucket = 5000)  
tree09 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr09 <- predict(tree09, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr09)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst09 <- predict(tree09, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst09)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree09$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 25, 5000, 5000))  
 comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100  
## 14 7 0.9196691 0.9201497 8 500 500  
## 15 3 0.7104350 0.7111666 4 1000 1000  
## 16 1 0.5177475 0.5184030 10 2500 2500  
## 17 1 0.5177475 0.5184030 25 5000 5000

# Tree 10

hypers = rpart.control(minsplit = 10000, maxdepth = 20, minbucket = 10000)  
tree10 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr10 <- predict(tree10, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr10)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst10 <- predict(tree10, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst10)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree10$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 20, 10000, 10000))  
comp\_tbl

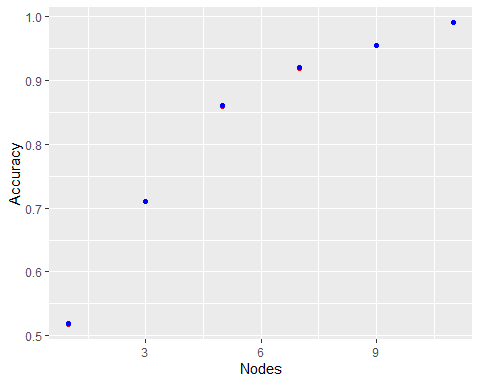
## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100  
## 14 7 0.9196691 0.9201497 8 500 500  
## 15 3 0.7104350 0.7111666 4 1000 1000  
## 16 1 0.5177475 0.5184030 10 2500 2500  
## 17 1 0.5177475 0.5184030 25 5000 5000  
## 18 1 0.5177475 0.5184030 20 10000 10000

# Tree 11

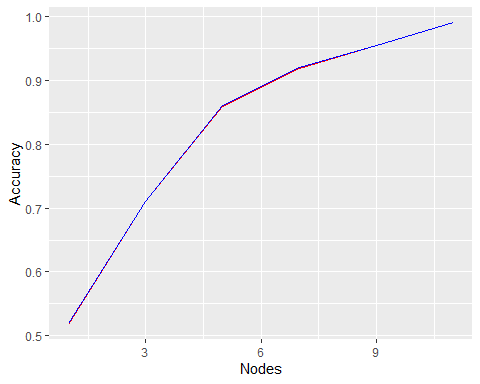
hypers = rpart.control(minsplit = 500, maxdepth = 5, minbucket = 500)  
tree11 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")  
  
# Training Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tr11 <- predict(tree11, train\_set)  
# Confusion Matrix  
cfm\_train <- confusionMatrix(train\_set$category, pred\_tree\_tr11)  
  
# Test Set  
# Evaluate the fit with a confusion matrix  
pred\_tree\_tst11 <- predict(tree11, test\_set)  
# Confusion Matrix  
cfm\_test <- confusionMatrix(test\_set$category, pred\_tree\_tst11)  
  
# Get training accuracy  
a\_train <- cfm\_train$overall[1]  
# Get testing accuracy  
a\_test <- cfm\_test$overall[1]  
# Get number of nodes  
nodes <- nrow(tree11$finalModel$frame)  
  
# Add rows to the table - Make sure the order is correct  
comp\_tbl <- comp\_tbl %>% rbind(list(nodes, a\_train, a\_test, 5, 500, 500))  
comp\_tbl

## Nodes TrainAccuracy TestAccuracy MaxDepth Minsplit Minbucket  
## Accuracy 3 0.7104350 0.7111666 1 2 2  
## 1 5 0.8590873 0.8602620 2 5 5  
## 11 7 0.9196691 0.9201497 3 10 10  
## 12 9 0.9554310 0.9557080 4 50 50  
## 13 11 0.9909261 0.9912664 6 100 100  
## 14 7 0.9196691 0.9201497 8 500 500  
## 15 3 0.7104350 0.7111666 4 1000 1000  
## 16 1 0.5177475 0.5184030 10 2500 2500  
## 17 1 0.5177475 0.5184030 25 5000 5000  
## 18 1 0.5177475 0.5184030 20 10000 10000  
## 19 7 0.9196691 0.9201497 5 500 500

# Visualize with scatter plot  
# dev.off()  
ggplot(comp\_tbl, aes(x=Nodes)) +   
 geom\_point(aes(y = TrainAccuracy), color = "red") +   
 geom\_point(aes(y = TestAccuracy), color="blue") +  
 ylab("Accuracy")



# Visualize with line plot  
# dev.off()  
ggplot(comp\_tbl, aes(x=Nodes)) +   
 geom\_line(aes(y = TrainAccuracy), color = "red") +   
 geom\_line(aes(y = TestAccuracy), color="blue") +  
 ylab("Accuracy")



# 3.c)c. Identify the final choice of model, list it parameters and evaluate with a the confusion matrix to make sure that it gets balanced performance over classes. Also get a better accuracy estimate for this tree using cross validation.

# Identifying and evaluating the final choice of model

# Tree 5

hypers = rpart.control(minsplit = 100, maxdepth = 6, minbucket = 100)  
tree05 <- train(category ~., data = train\_set, control = hypers, trControl = train\_control, method = "rpart1SE")

# Evaluating the fit with a confusion matrix for Tree-5

# Evaluate the fit with a confusion matrix for tree-5  
pred\_tree <- predict(tree05, stormsNA)  
# Confusion Matrix  
confusionMatrix(stormsNA$category, pred\_tree)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction -1 0 1 2 3 4 5  
## -1 1031 0 0 0 0 0 0  
## 0 0 2771 0 0 0 0 0  
## 1 0 1 796 0 0 0 0  
## 2 0 0 0 323 0 0 0  
## 3 0 0 0 0 190 0 0  
## 4 0 0 0 0 0 191 0  
## 5 0 0 0 0 0 47 0  
##   
## Overall Statistics  
##   
## Accuracy : 0.991   
## 95% CI : (0.9881, 0.9934)  
## No Information Rate : 0.5181   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9865   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: -1 Class: 0 Class: 1 Class: 2 Class: 3 Class: 4  
## Sensitivity 1.0000 0.9996 1.0000 1.00000 1.00000 0.80252  
## Specificity 1.0000 1.0000 0.9998 1.00000 1.00000 1.00000  
## Pos Pred Value 1.0000 1.0000 0.9987 1.00000 1.00000 1.00000  
## Neg Pred Value 1.0000 0.9996 1.0000 1.00000 1.00000 0.99089  
## Prevalence 0.1927 0.5181 0.1488 0.06037 0.03551 0.04449  
## Detection Rate 0.1927 0.5179 0.1488 0.06037 0.03551 0.03570  
## Detection Prevalence 0.1927 0.5179 0.1490 0.06037 0.03551 0.03570  
## Balanced Accuracy 1.0000 0.9998 0.9999 1.00000 1.00000 0.90126  
## Class: 5  
## Sensitivity NA  
## Specificity 0.991215  
## Pos Pred Value NA  
## Neg Pred Value NA  
## Prevalence 0.000000  
## Detection Rate 0.000000  
## Detection Prevalence 0.008785  
## Balanced Accuracy NA

# Training Set (for Tree-5)

# Evaluate the fit for training set with a confusion matrix for tree-5  
pred\_tree\_trcm05 <- predict(tree05, train\_set)  
# Confusion Matrix  
confusionMatrix(train\_set$category, pred\_tree\_trcm05)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction -1 0 1 2 3 4 5  
## -1 722 0 0 0 0 0 0  
## 0 0 1940 0 0 0 0 0  
## 1 0 1 557 0 0 0 0  
## 2 0 0 0 227 0 0 0  
## 3 0 0 0 0 133 0 0  
## 4 0 0 0 0 0 134 0  
## 5 0 0 0 0 0 33 0  
##   
## Overall Statistics  
##   
## Accuracy : 0.9909   
## 95% CI : (0.9873, 0.9937)  
## No Information Rate : 0.518   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9864   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: -1 Class: 0 Class: 1 Class: 2 Class: 3 Class: 4  
## Sensitivity 1.0000 0.9995 1.0000 1.00000 1.0000 0.80240  
## Specificity 1.0000 1.0000 0.9997 1.00000 1.0000 1.00000  
## Pos Pred Value 1.0000 1.0000 0.9982 1.00000 1.0000 1.00000  
## Neg Pred Value 1.0000 0.9994 1.0000 1.00000 1.0000 0.99087  
## Prevalence 0.1927 0.5180 0.1487 0.06058 0.0355 0.04457  
## Detection Rate 0.1927 0.5177 0.1487 0.06058 0.0355 0.03576  
## Detection Prevalence 0.1927 0.5177 0.1489 0.06058 0.0355 0.03576  
## Balanced Accuracy 1.0000 0.9997 0.9998 1.00000 1.0000 0.90120  
## Class: 5  
## Sensitivity NA  
## Specificity 0.991193  
## Pos Pred Value NA  
## Neg Pred Value NA  
## Prevalence 0.000000  
## Detection Rate 0.000000  
## Detection Prevalence 0.008807  
## Balanced Accuracy NA

# Test Set (for Tree-5)

# Evaluate the fit testing set with a confusion matrix for tree-5  
pred\_tree\_tstcm05 <- predict(tree05, test\_set)  
# Confusion Matrix  
confusionMatrix(test\_set$category, pred\_tree\_tstcm05)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction -1 0 1 2 3 4 5  
## -1 309 0 0 0 0 0 0  
## 0 0 831 0 0 0 0 0  
## 1 0 0 239 0 0 0 0  
## 2 0 0 0 96 0 0 0  
## 3 0 0 0 0 57 0 0  
## 4 0 0 0 0 0 57 0  
## 5 0 0 0 0 0 14 0  
##   
## Overall Statistics  
##   
## Accuracy : 0.9913   
## 95% CI : (0.9854, 0.9952)  
## No Information Rate : 0.5184   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.9869   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: -1 Class: 0 Class: 1 Class: 2 Class: 3 Class: 4  
## Sensitivity 1.0000 1.0000 1.0000 1.00000 1.00000 0.80282  
## Specificity 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Pos Pred Value 1.0000 1.0000 1.0000 1.00000 1.00000 1.00000  
## Neg Pred Value 1.0000 1.0000 1.0000 1.00000 1.00000 0.99094  
## Prevalence 0.1928 0.5184 0.1491 0.05989 0.03556 0.04429  
## Detection Rate 0.1928 0.5184 0.1491 0.05989 0.03556 0.03556  
## Detection Prevalence 0.1928 0.5184 0.1491 0.05989 0.03556 0.03556  
## Balanced Accuracy 1.0000 1.0000 1.0000 1.00000 1.00000 0.90141  
## Class: 5  
## Sensitivity NA  
## Specificity 0.991266  
## Pos Pred Value NA  
## Neg Pred Value NA  
## Prevalence 0.000000  
## Detection Rate 0.000000  
## Detection Prevalence 0.008734  
## Balanced Accuracy NA

# Using Cross Validation to get a better accuracy estimate for tree-5.

train\_control = trainControl(method = "cv", number = 10)  
Tree05 <- train(category ~., data = train\_set, trControl = train\_control, method = "rpart1SE")  
Tree05

## CART   
##   
## 3747 samples  
## 12 predictor  
## 7 classes: '-1', '0', '1', '2', '3', '4', '5'   
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 3372, 3372, 3374, 3374, 3373, 3372, ...   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.9997319 0.9995957