Bank Marketing\_Decision Tree

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Importing the dataset

bank <- read.csv("bank.csv",header = TRUE, sep = ",")

summary(bank)

## age job marital education   
## Min. :19.00 management :969 divorced: 528 primary : 678   
## 1st Qu.:33.00 blue-collar:946 married :2797 secondary:2306   
## Median :39.00 technician :768 single :1196 tertiary :1350   
## Mean :41.17 admin. :478 unknown : 187   
## 3rd Qu.:49.00 services :417   
## Max. :87.00 retired :230   
## (Other) :713   
## default balance housing loan contact   
## no :4445 Min. :-3313 no :1962 no :3830 cellular :2896   
## yes: 76 1st Qu.: 69 yes:2559 yes: 691 telephone: 301   
## Median : 444 unknown :1324   
## Mean : 1423   
## 3rd Qu.: 1480   
## Max. :71188   
##   
## day month duration campaign   
## Min. : 1.00 may :1398 Min. : 4 Min. : 1.000   
## 1st Qu.: 9.00 jul : 706 1st Qu.: 104 1st Qu.: 1.000   
## Median :16.00 aug : 633 Median : 185 Median : 2.000   
## Mean :15.92 jun : 531 Mean : 264 Mean : 2.794   
## 3rd Qu.:21.00 nov : 389 3rd Qu.: 329 3rd Qu.: 3.000   
## Max. :31.00 apr : 293 Max. :3025 Max. :50.000   
## (Other): 571   
## pdays previous poutcome y   
## Min. : -1.00 Min. : 0.0000 failure: 490 no :4000   
## 1st Qu.: -1.00 1st Qu.: 0.0000 other : 197 yes: 521   
## Median : -1.00 Median : 0.0000 success: 129   
## Mean : 39.77 Mean : 0.5426 unknown:3705   
## 3rd Qu.: -1.00 3rd Qu.: 0.0000   
## Max. :871.00 Max. :25.0000   
##

str(bank)

## 'data.frame': 4521 obs. of 17 variables:  
## $ age : int 30 33 35 30 59 35 36 39 41 43 ...  
## $ job : Factor w/ 12 levels "admin.","blue-collar",..: 11 8 5 5 2 5 7 10 3 8 ...  
## $ marital : Factor w/ 3 levels "divorced","married",..: 2 2 3 2 2 3 2 2 2 2 ...  
## $ education: Factor w/ 4 levels "primary","secondary",..: 1 2 3 3 2 3 3 2 3 1 ...  
## $ default : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...  
## $ balance : int 1787 4789 1350 1476 0 747 307 147 221 -88 ...  
## $ housing : Factor w/ 2 levels "no","yes": 1 2 2 2 2 1 2 2 2 2 ...  
## $ loan : Factor w/ 2 levels "no","yes": 1 2 1 2 1 1 1 1 1 2 ...  
## $ contact : Factor w/ 3 levels "cellular","telephone",..: 1 1 1 3 3 1 1 1 3 1 ...  
## $ day : int 19 11 16 3 5 23 14 6 14 17 ...  
## $ month : Factor w/ 12 levels "apr","aug","dec",..: 11 9 1 7 9 4 9 9 9 1 ...  
## $ duration : int 79 220 185 199 226 141 341 151 57 313 ...  
## $ campaign : int 1 1 1 4 1 2 1 2 2 1 ...  
## $ pdays : int -1 339 330 -1 -1 176 330 -1 -1 147 ...  
## $ previous : int 0 4 1 0 0 3 2 0 0 2 ...  
## $ poutcome : Factor w/ 4 levels "failure","other",..: 4 1 1 4 4 1 2 4 4 1 ...  
## $ y : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...

bank$balance <- as.numeric(bank$balance)  
bank$day <- as.numeric(bank$day)  
bank$duration <- as.numeric(bank$duration)  
bank$campaign <- as.numeric(bank$campaign)  
bank$pdays <- as.numeric(bank$pdays)  
bank$previous <- as.numeric(bank$previous)

Splitting the dataset into training set and test set

# install.packages("caTools")  
library(caTools)  
set.seed(123)  
split = sample.split(bank$y, SplitRatio = 2/3)  
training\_set = subset(bank, split == TRUE)  
test\_set = subset(bank, split == FALSE)

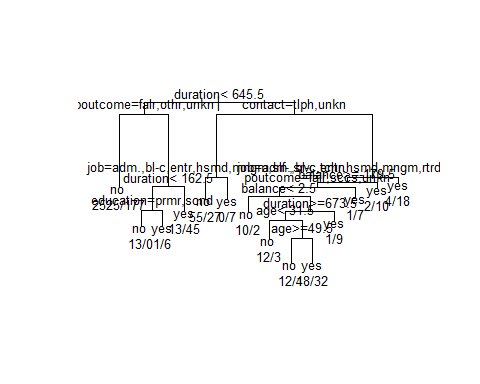
Fitting logistic regression to the training test

# install.packages("rpart")  
library(rpart)  
classifier = rpart(y ~ ., data = training\_set)  
classifier

## n= 3014   
##   
## node), split, n, loss, yval, (yprob)  
## \* denotes terminal node  
##   
## 1) root 3014 347 no (0.88487060 0.11512940)   
## 2) duration< 645.5 2780 228 no (0.91798561 0.08201439)   
## 4) poutcome=failure,other,unknown 2702 177 no (0.93449297 0.06550703) \*  
## 5) poutcome=success 78 27 yes (0.34615385 0.65384615)   
## 10) duration< 162.5 20 6 no (0.70000000 0.30000000)   
## 20) education=primary,secondary 13 0 no (1.00000000 0.00000000) \*  
## 21) education=tertiary 7 1 yes (0.14285714 0.85714286) \*  
## 11) duration>=162.5 58 13 yes (0.22413793 0.77586207) \*  
## 3) duration>=645.5 234 115 yes (0.49145299 0.50854701)   
## 6) contact=telephone,unknown 89 34 no (0.61797753 0.38202247)   
## 12) job=admin.,blue-collar,entrepreneur,housemaid,management,self-employed,services,technician 82 27 no (0.67073171 0.32926829) \*  
## 13) job=retired 7 0 yes (0.00000000 1.00000000) \*  
## 7) contact=cellular 145 60 yes (0.41379310 0.58620690)   
## 14) job=admin.,blue-collar,entrepreneur,housemaid,management,retired,technician,unemployed 123 56 yes (0.45528455 0.54471545)   
## 28) balance>=-179.5 111 54 yes (0.48648649 0.51351351)   
## 56) poutcome=failure,success,unknown 103 50 no (0.51456311 0.48543689)   
## 112) balance< 2.5 12 2 no (0.83333333 0.16666667) \*  
## 113) balance>=2.5 91 43 yes (0.47252747 0.52747253)   
## 226) duration>=673.5 81 39 no (0.51851852 0.48148148)   
## 452) age< 31.5 15 3 no (0.80000000 0.20000000) \*  
## 453) age>=31.5 66 30 yes (0.45454545 0.54545455)   
## 906) age>=49.5 16 4 no (0.75000000 0.25000000) \*  
## 907) age< 49.5 50 18 yes (0.36000000 0.64000000) \*  
## 227) duration< 673.5 10 1 yes (0.10000000 0.90000000) \*  
## 57) poutcome=other 8 1 yes (0.12500000 0.87500000) \*  
## 29) balance< -179.5 12 2 yes (0.16666667 0.83333333) \*  
## 15) job=self-employed,services,student,unknown 22 4 yes (0.18181818 0.81818182) \*

Plot the data

plot(classifier, margin = 0.1)  
text(classifier, use.n = TRUE, pretty = TRUE, cex = 0.8)



Predicting the test results

y\_pred = predict(classifier, newdata = test\_set, type = "class")  
y\_pred

## 2 4 5 8 11 13 17 21 22 23 25 27 31 33 35   
## no no no no no no no no no no no no yes no no   
## 36 39 40 41 44 54 60 63 68 69 76 78 79 80 81   
## no yes no no no no no no no no no no no no yes   
## 83 87 97 99 100 103 104 105 109 111 117 124 127 128 130   
## no no no yes no no no no no yes no yes no no no   
## 133 136 137 139 141 149 153 155 156 158 161 162 163 165 170   
## no no no no no no no no no no no no no no no   
## 175 176 190 193 199 200 201 202 208 210 212 216 219 220 223   
## no no no no no yes yes no no no no yes no no no   
## 225 232 236 244 246 249 250 252 253 254 259 260 261 269 271   
## no no no no no no no no no no no no no no no   
## 278 280 281 282 288 294 295 297 298 300 301 302 307 310 314   
## no no no no no no no no no no no no yes no no   
## 315 316 317 322 328 334 336 340 341 342 343 346 347 350 351   
## no no no no yes yes no no no no no no no no no   
## 361 364 365 368 370 373 376 378 380 384 385 388 391 398 404   
## no no no no yes no no no no no no no no yes no   
## 408 409 414 418 421 428 431 432 435 437 439 441 444 447 449   
## no no no no no no no no no no no no no no no   
## 450 451 458 459 461 466 469 471 477 486 491 492 498 504 508   
## no yes no no no no no no no no no no no no no   
## 510 511 512 513 514 517 524 525 526 529 536 538 540 543 549   
## no no yes no no no no no no no no no no no no   
## 551 553 554 557 561 562 565 567 570 571 580 585 587 590 592   
## no no no no no no no no no no no no no yes no   
## 598 599 601 603 605 607 609 611 615 617 619 620 621 623 624   
## no no no no no no no no no no no yes no no no   
## 629 631 635 639 640 641 648 650 652 655 656 661 664 665 667   
## no no no no no no no no no no no no yes no no   
## 671 675 678 681 684 685 689 691 697 699 700 701 702 703 706   
## no no no no no yes no no no no no no no yes no   
## 707 708 709 714 717 718 720 723 724 725 728 729 734 738 739   
## no no no no no no no no no no no no no no no   
## 740 742 744 745 746 747 751 752 754 758 768 772 773 774 775   
## no no no no no no no no no no no no no no no   
## 776 792 796 806 812 815 816 817 819 820 822 826 828 831 833   
## no no no no no no no no no no no no no no no   
## 836 843 845 846 855 856 868 870 871 873 874 880 883 886 892   
## no no no no no yes no no no no no no no no no   
## 894 898 906 910 913 914 916 918 919 925 928 929 934 936 938   
## no no no no no no no no no no no no no no no   
## 941 943 945 946 949 950 955 960 961 962 965 969 971 976 986   
## no no no no no no no yes no no no no no no no   
## 988 989 991 993 996 997 999 1000 1006 1007 1008 1010 1011 1015 1016   
## no no no no no no no no no no no no no no no   
## 1019 1024 1026 1027 1028 1029 1033 1036 1041 1042 1045 1046 1047 1049 1052   
## no no no no no no no no no no no no no yes no   
## 1053 1055 1058 1062 1063 1065 1071 1077 1078 1080 1083 1086 1093 1097 1099   
## no no no no no no no no no no no yes no no no   
## 1101 1105 1111 1113 1115 1116 1127 1130 1135 1137 1139 1149 1152 1156 1159   
## no no no no no no no no no no no no no no no   
## 1167 1177 1179 1182 1189 1190 1191 1192 1193 1194 1201 1202 1207 1208 1211   
## yes no no no no no no no yes no no no yes no no   
## 1214 1215 1216 1217 1228 1233 1235 1239 1240 1241 1245 1246 1248 1249 1253   
## no no no yes no no no no no no no no no no no   
## 1254 1257 1263 1267 1269 1274 1281 1282 1284 1285 1291 1294 1303 1304 1306   
## no no no no no no no no no no no no no yes no   
## 1307 1311 1313 1322 1325 1329 1332 1340 1341 1342 1345 1348 1349 1355 1358   
## no no yes no no no no no no no no no no no no   
## 1360 1362 1363 1365 1367 1370 1373 1377 1379 1382 1386 1387 1391 1401 1402   
## no no no no no no no no no no no no no no no   
## 1408 1414 1415 1421 1424 1431 1435 1439 1441 1442 1445 1448 1450 1454 1457   
## no no no no no no yes no no yes no no no no no   
## 1461 1466 1468 1471 1474 1475 1476 1481 1482 1487 1491 1493 1495 1499 1501   
## no no no no no no no no no no no no no no no   
## 1504 1510 1518 1519 1520 1527 1533 1541 1546 1558 1559 1560 1562 1567 1569   
## no yes no no no no no no no yes no no no no no   
## 1572 1576 1577 1581 1591 1593 1595 1596 1597 1601 1604 1607 1610 1615 1616   
## no no no no no no no no no no yes no no no no   
## 1618 1620 1621 1625 1632 1633 1640 1641 1644 1646 1647 1649 1650 1651 1655   
## no no no no no no no no no no no no no no no   
## 1656 1660 1662 1664 1666 1668 1669 1670 1671 1674 1675 1676 1678 1681 1682   
## no no no no no no no no no no no no no no no   
## 1685 1686 1688 1689 1693 1695 1696 1700 1706 1707 1711 1712 1721 1723 1725   
## no no no no no no yes no no yes no no no no no   
## 1733 1734 1735 1737 1743 1746 1749 1750 1757 1758 1761 1764 1766 1770 1771   
## no no no yes no no no no no no yes no no no no   
## 1773 1778 1780 1783 1786 1795 1801 1802 1809 1814 1815 1817 1822 1825 1826   
## no no yes no no no no no no no no no no no no   
## 1835 1837 1838 1842 1843 1849 1850 1854 1857 1858 1859 1861 1862 1864 1868   
## no no no no no no no no no no no no no yes no   
## 1875 1878 1879 1883 1885 1887 1894 1904 1912 1913 1921 1922 1923 1924 1929   
## no no no no no yes no no no no no no no no no   
## 1930 1933 1934 1936 1939 1945 1948 1951 1954 1959 1963 1965 1968 1969 1972   
## no no no no no no no no no no no no no no no   
## 1973 1974 1977 1982 1983 1984 1986 1987 1988 1990 1993 1997 1998 2003 2004   
## no no no no no no no no no no no no no no no   
## 2006 2007 2008 2023 2027 2029 2030 2034 2036 2041 2045 2047 2051 2052 2054   
## no no no no no no no yes no no no no no no no   
## 2059 2060 2067 2069 2071 2073 2076 2077 2078 2079 2080 2082 2088 2089 2090   
## no no no yes yes no no no no no no no no no no   
## 2094 2099 2101 2102 2107 2110 2112 2114 2115 2119 2122 2125 2129 2130 2134   
## no no no no yes no no no no no no no no no no   
## 2136 2141 2142 2150 2152 2153 2156 2160 2163 2167 2179 2180 2182 2185 2192   
## no no no no no no no no no no no no no no no   
## 2201 2203 2204 2208 2210 2214 2215 2218 2220 2226 2229 2238 2241 2244 2248   
## no no no no no no no no yes no yes no no no no   
## 2250 2252 2255 2264 2267 2269 2275 2283 2287 2289 2293 2295 2300 2303 2309   
## no yes no no yes no no no no no no no no no no   
## 2311 2312 2313 2317 2318 2319 2320 2327 2329 2332 2334 2335 2337 2339 2344   
## no no no no no no no no no no no no no no no   
## 2345 2347 2351 2354 2357 2358 2363 2364 2372 2373 2374 2383 2384 2385 2388   
## no no no no no no no no no yes no no yes no no   
## 2389 2390 2396 2398 2399 2400 2404 2406 2409 2414 2417 2418 2421 2422 2431   
## no no no no no no no no no no no no no no no   
## 2432 2435 2436 2437 2442 2446 2449 2452 2454 2455 2461 2464 2467 2468 2473   
## no no no no no no no no no no no no no no no   
## 2475 2476 2479 2480 2483 2484 2485 2486 2488 2492 2493 2494 2499 2504 2516   
## no no no no no no no no no no no no no no no   
## 2518 2522 2524 2525 2530 2533 2537 2538 2539 2543 2546 2547 2549 2556 2558   
## no no no no no no no no no no no no no no no   
## 2559 2562 2565 2573 2574 2575 2580 2591 2594 2597 2599 2601 2604 2607 2608   
## no no no no no no no no no no no no no no no   
## 2615 2621 2624 2634 2635 2636 2639 2640 2641 2644 2645 2646 2653 2654 2655   
## no no no no no no no no no no no yes no no no   
## 2656 2657 2659 2666 2668 2670 2675 2677 2679 2685 2686 2689 2692 2694 2699   
## no no no yes no no no no no no no no no no no   
## 2701 2706 2707 2708 2712 2713 2716 2717 2724 2726 2727 2730 2737 2743 2745   
## no no no no yes no no no no no no no no no no   
## 2746 2747 2751 2755 2757 2764 2769 2772 2773 2774 2775 2783 2797 2799 2803   
## no yes no no no no no no yes no no no no no no   
## 2805 2810 2811 2814 2815 2816 2817 2821 2824 2826 2830 2833 2834 2841 2849   
## no no no no no no no no yes no no no no no no   
## 2851 2855 2859 2860 2864 2868 2869 2870 2871 2874 2875 2877 2879 2886 2888   
## no no no no no no no no no no no no no no yes   
## 2895 2899 2903 2914 2915 2919 2923 2924 2925 2929 2933 2935 2936 2942 2951   
## no no yes no no no no no no no no no no no no   
## 2957 2961 2962 2963 2967 2968 2976 2977 2983 2989 2990 2993 2994 2995 2999   
## no no no no no no no no no no no yes no no no   
## 3000 3001 3008 3013 3017 3024 3031 3034 3038 3039 3042 3044 3046 3052 3054   
## no no no no no no no no no no no no no no no   
## 3056 3058 3062 3063 3064 3065 3066 3068 3073 3079 3081 3083 3084 3085 3086   
## no no no yes no no no no no no no no no no no   
## 3089 3092 3093 3095 3097 3098 3099 3100 3105 3106 3111 3118 3119 3120 3122   
## yes no no no no no no no no no no no no no no   
## 3127 3129 3131 3139 3140 3146 3148 3149 3151 3153 3154 3157 3159 3162 3166   
## no no no no no no no no no no no no no no no   
## 3171 3174 3176 3177 3178 3180 3181 3182 3185 3186 3189 3192 3193 3196 3202   
## no no no yes no no no no no no no no no no no   
## 3203 3204 3207 3209 3210 3213 3214 3216 3217 3222 3227 3231 3233 3234 3238   
## yes no no no no no no no no no no no no no no   
## 3239 3240 3242 3249 3250 3251 3252 3256 3259 3260 3263 3264 3267 3269 3275   
## no no no yes no yes no no no no no no no no no   
## 3279 3281 3283 3285 3288 3289 3291 3293 3295 3296 3299 3302 3303 3307 3309   
## no no no no no no no no no no no no no no no   
## 3310 3320 3328 3332 3333 3336 3341 3345 3347 3350 3352 3354 3356 3360 3361   
## no no no yes no no no no no yes no no no no yes   
## 3367 3368 3370 3372 3377 3378 3384 3386 3387 3389 3392 3394 3400 3402 3403   
## no no no no no no no yes no no no no no no no   
## 3404 3410 3412 3414 3416 3419 3420 3421 3422 3426 3427 3431 3437 3443 3447   
## no no no yes no no no no no no no no no no no   
## 3450 3453 3455 3456 3457 3458 3459 3464 3466 3469 3470 3474 3477 3478 3479   
## no no no no no no no yes no no no no no no no   
## 3485 3489 3490 3492 3494 3502 3503 3505 3508 3510 3511 3516 3518 3520 3521   
## no no no no no no yes no no no no no no no no   
## 3525 3526 3529 3530 3533 3535 3536 3545 3546 3553 3557 3558 3559 3560 3568   
## no no no no no no no no no no yes no no no no   
## 3571 3574 3575 3582 3585 3587 3588 3589 3590 3592 3600 3602 3604 3605 3606   
## no no no no no yes no no yes no no no no no no   
## 3609 3613 3623 3625 3627 3629 3631 3633 3635 3636 3638 3646 3648 3654 3656   
## no no no no no no no no no no no no no yes no   
## 3657 3658 3660 3661 3664 3668 3670 3673 3674 3675 3683 3684 3691 3692 3694   
## no no no no no no no no no no no no no no no   
## 3695 3705 3707 3709 3717 3720 3726 3728 3731 3734 3735 3736 3738 3742 3744   
## no no no no no no no no no no yes no no no no   
## 3747 3751 3754 3755 3760 3762 3764 3767 3770 3771 3772 3775 3777 3778 3782   
## no no no yes no no no no no no no no no no no   
## 3784 3790 3792 3796 3798 3801 3804 3805 3808 3810 3812 3815 3819 3824 3827   
## no no no no no no no no no no no no yes no no   
## 3828 3833 3844 3845 3847 3849 3855 3856 3858 3860 3864 3868 3871 3879 3880   
## no no yes no no no no no no no no no no no no   
## 3882 3885 3890 3907 3912 3913 3914 3922 3923 3929 3937 3940 3941 3942 3943   
## no no no no no no no no no no no no no no no   
## 3944 3948 3950 3951 3958 3962 3965 3966 3967 3969 3975 3985 3986 3987 3989   
## yes no no no no no no no no no no no no no no   
## 3991 3992 3994 4000 4003 4004 4005 4006 4008 4009 4012 4013 4015 4024 4026   
## no no no no no no no no no no no no no no no   
## 4028 4031 4032 4034 4035 4036 4037 4043 4046 4047 4050 4058 4059 4060 4069   
## no no no no no no no no no no no no no no no   
## 4070 4072 4083 4085 4089 4091 4094 4100 4101 4106 4109 4110 4113 4117 4119   
## no no no no no no yes no no no no no no no no   
## 4122 4123 4124 4125 4126 4130 4133 4135 4141 4142 4145 4146 4148 4149 4153   
## no no no no no no no no no no no no no no no   
## 4154 4156 4159 4162 4163 4172 4177 4178 4180 4181 4184 4188 4189 4190 4193   
## no no no no no yes no no no no no no no no no   
## 4198 4202 4209 4210 4217 4219 4222 4224 4225 4228 4234 4238 4240 4242 4244   
## no no no yes yes no no no yes no no no no no no   
## 4250 4254 4256 4257 4258 4259 4261 4262 4266 4267 4268 4271 4272 4275 4279   
## no no yes no no no no no no no no no no no no   
## 4281 4282 4287 4292 4294 4296 4297 4299 4305 4306 4311 4312 4318 4320 4321   
## no no no no no no no no no no no no no no no   
## 4324 4325 4331 4336 4340 4345 4348 4353 4354 4360 4363 4368 4369 4370 4373   
## yes no no no no no no no no no no no no no no   
## 4374 4375 4377 4380 4383 4385 4386 4390 4391 4398 4407 4408 4412 4417 4418   
## no no no no no no no no no no no yes no no yes   
## 4419 4420 4422 4424 4430 4431 4435 4439 4442 4443 4444 4445 4449 4453 4454   
## no no no no no no no no no no no no no no no   
## 4459 4461 4462 4465 4467 4468 4474 4477 4478 4479 4484 4488 4490 4491 4492   
## yes no no no no no no no no no no no no no no   
## 4494 4496 4500 4503 4515 4518 4519   
## no no no no no no no   
## Levels: no yes

Confusion matrix

cm = table(test\_set$y, y\_pred)  
cm

## y\_pred  
## no yes  
## no 1297 36  
## yes 116 58

Accuracy is the ratio of correctly predicted observation to the total observations

accuracy <- sum(diag(cm))/sum(cm)   
accuracy

## [1] 0.8991374

So we got 90% accuracy Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

precision <- cm[2,2]/(sum(cm[,2]))  
precision

## [1] 0.6170213

We got 62% precision Recall is the ratio of correctly predicted positive observations to the all observations in actual class

recall <- cm[2,2]/(sum(cm[2,]))  
recall

## [1] 0.3333333

We got 33% recall

F1 <- 2\*(recall \* precision) / (recall + precision)  
F1

## [1] 0.4328358

43% is the F1 score

We are removing age, job, previous, default, pdays, balance The new dataset becoomes

bank <- bank[c(3:4, 7:13, 16:17)]  
str(bank)

## 'data.frame': 4521 obs. of 11 variables:  
## $ marital : Factor w/ 3 levels "divorced","married",..: 2 2 3 2 2 3 2 2 2 2 ...  
## $ education: Factor w/ 4 levels "primary","secondary",..: 1 2 3 3 2 3 3 2 3 1 ...  
## $ housing : Factor w/ 2 levels "no","yes": 1 2 2 2 2 1 2 2 2 2 ...  
## $ loan : Factor w/ 2 levels "no","yes": 1 2 1 2 1 1 1 1 1 2 ...  
## $ contact : Factor w/ 3 levels "cellular","telephone",..: 1 1 1 3 3 1 1 1 3 1 ...  
## $ day : num 19 11 16 3 5 23 14 6 14 17 ...  
## $ month : Factor w/ 12 levels "apr","aug","dec",..: 11 9 1 7 9 4 9 9 9 1 ...  
## $ duration : num 79 220 185 199 226 141 341 151 57 313 ...  
## $ campaign : num 1 1 1 4 1 2 1 2 2 1 ...  
## $ poutcome : Factor w/ 4 levels "failure","other",..: 4 1 1 4 4 1 2 4 4 1 ...  
## $ y : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...

Splitting the dataset into training set and test set

set.seed(123)  
split = sample.split(bank$y, SplitRatio = 2/3)  
training\_set = subset(bank, split == TRUE)  
test\_set = subset(bank, split == FALSE)

Fitting logistic regression to the training test

classifier = glm(y ~ ., family = binomial, data = training\_set)  
summary(classifier)

##   
## Call:  
## glm(formula = y ~ ., family = binomial, data = training\_set)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.1663 -0.3861 -0.2639 -0.1694 3.0611   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.0678257 0.4531896 -6.769 1.29e-11 \*\*\*  
## maritalmarried -0.3018430 0.2169482 -1.391 0.164130   
## maritalsingle -0.1043632 0.2365918 -0.441 0.659133   
## educationsecondary 0.1276332 0.2230204 0.572 0.567123   
## educationtertiary 0.4185558 0.2325967 1.799 0.071941 .   
## educationunknown -0.1764658 0.3967326 -0.445 0.656466   
## housingyes -0.5031150 0.1610407 -3.124 0.001783 \*\*   
## loanyes -0.3526905 0.2266569 -1.556 0.119695   
## contacttelephone 0.1379167 0.2622240 0.526 0.598923   
## contactunknown -1.4830792 0.2707679 -5.477 4.32e-08 \*\*\*  
## day 0.0241640 0.0099253 2.435 0.014908 \*   
## monthaug -0.7132801 0.2957870 -2.411 0.015889 \*   
## monthdec 0.4861161 0.8652433 0.562 0.574235   
## monthfeb 0.1058598 0.3524892 0.300 0.763932   
## monthjan -1.0906635 0.4315884 -2.527 0.011501 \*   
## monthjul -1.0419580 0.2982600 -3.493 0.000477 \*\*\*  
## monthjun 0.4918584 0.3560914 1.381 0.167196   
## monthmar 0.9657465 0.4684404 2.062 0.039244 \*   
## monthmay -0.5130083 0.2760691 -1.858 0.063132 .   
## monthnov -0.8099755 0.3133784 -2.585 0.009748 \*\*   
## monthoct 1.1239597 0.4108586 2.736 0.006226 \*\*   
## monthsep 0.7419802 0.4712419 1.575 0.115367   
## duration 0.0040939 0.0002417 16.941 < 2e-16 \*\*\*  
## campaign -0.0725344 0.0334349 -2.169 0.030051 \*   
## poutcomeother 0.7694321 0.3174853 2.424 0.015371 \*   
## poutcomesuccess 2.6595806 0.3369931 7.892 2.97e-15 \*\*\*  
## poutcomeunknown 0.1750725 0.2354725 0.743 0.457182   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 2152.6 on 3013 degrees of freedom  
## Residual deviance: 1480.3 on 2987 degrees of freedom  
## AIC: 1534.3  
##   
## Number of Fisher Scoring iterations: 6

Predicting the test results

prob\_pred = predict(classifier, type = 'response', newdata = test\_set)  
y\_pred = ifelse(prob\_pred > 0.5, 1, 0)  
y\_pred

## 2 4 5 8 11 13 17 21 22 23 25 27 31 33 35   
## 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0   
## 36 39 40 41 44 54 60 63 68 69 76 78 79 80 81   
## 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0   
## 83 87 97 99 100 103 104 105 109 111 117 124 127 128 130   
## 0 0 0 1 1 0 0 0 0 1 0 0 0 0 0   
## 133 136 137 139 141 149 153 155 156 158 161 162 163 165 170   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   
## 175 176 190 193 199 200 201 202 208 210 212 216 219 220 223   
## 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0   
## 225 232 236 244 246 249 250 252 253 254 259 260 261 269 271   
## 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0   
## 278 280 281 282 288 294 295 297 298 300 301 302 307 310 314   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 315 316 317 322 328 334 336 340 341 342 343 346 347 350 351   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 361 364 365 368 370 373 376 378 380 384 385 388 391 398 404   
## 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0   
## 408 409 414 418 421 428 431 432 435 437 439 441 444 447 449   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 450 451 458 459 461 466 469 471 477 486 491 492 498 504 508   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 510 511 512 513 514 517 524 525 526 529 536 538 540 543 549   
## 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 551 553 554 557 561 562 565 567 570 571 580 585 587 590 592   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 598 599 601 603 605 607 609 611 615 617 619 620 621 623 624   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 629 631 635 639 640 641 648 650 652 655 656 661 664 665 667   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 671 675 678 681 684 685 689 691 697 699 700 701 702 703 706   
## 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0   
## 707 708 709 714 717 718 720 723 724 725 728 729 734 738 739   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 740 742 744 745 746 747 751 752 754 758 768 772 773 774 775   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 776 792 796 806 812 815 816 817 819 820 822 826 828 831 833   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 836 843 845 846 855 856 868 870 871 873 874 880 883 886 892   
## 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0   
## 894 898 906 910 913 914 916 918 919 925 928 929 934 936 938   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 941 943 945 946 949 950 955 960 961 962 965 969 971 976 986   
## 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0   
## 988 989 991 993 996 997 999 1000 1006 1007 1008 1010 1011 1015 1016   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1019 1024 1026 1027 1028 1029 1033 1036 1041 1042 1045 1046 1047 1049 1052   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   
## 1053 1055 1058 1062 1063 1065 1071 1077 1078 1080 1083 1086 1093 1097 1099   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1101 1105 1111 1113 1115 1116 1127 1130 1135 1137 1139 1149 1152 1156 1159   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1167 1177 1179 1182 1189 1190 1191 1192 1193 1194 1201 1202 1207 1208 1211   
## 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1214 1215 1216 1217 1228 1233 1235 1239 1240 1241 1245 1246 1248 1249 1253   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0   
## 1254 1257 1263 1267 1269 1274 1281 1282 1284 1285 1291 1294 1303 1304 1306   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   
## 1307 1311 1313 1322 1325 1329 1332 1340 1341 1342 1345 1348 1349 1355 1358   
## 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0   
## 1360 1362 1363 1365 1367 1370 1373 1377 1379 1382 1386 1387 1391 1401 1402   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1408 1414 1415 1421 1424 1431 1435 1439 1441 1442 1445 1448 1450 1454 1457   
## 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0   
## 1461 1466 1468 1471 1474 1475 1476 1481 1482 1487 1491 1493 1495 1499 1501   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1504 1510 1518 1519 1520 1527 1533 1541 1546 1558 1559 1560 1562 1567 1569   
## 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0   
## 1572 1576 1577 1581 1591 1593 1595 1596 1597 1601 1604 1607 1610 1615 1616   
## 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0   
## 1618 1620 1621 1625 1632 1633 1640 1641 1644 1646 1647 1649 1650 1651 1655   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1656 1660 1662 1664 1666 1668 1669 1670 1671 1674 1675 1676 1678 1681 1682   
## 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0   
## 1685 1686 1688 1689 1693 1695 1696 1700 1706 1707 1711 1712 1721 1723 1725   
## 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0   
## 1733 1734 1735 1737 1743 1746 1749 1750 1757 1758 1761 1764 1766 1770 1771   
## 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0   
## 1773 1778 1780 1783 1786 1795 1801 1802 1809 1814 1815 1817 1822 1825 1826   
## 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0   
## 1835 1837 1838 1842 1843 1849 1850 1854 1857 1858 1859 1861 1862 1864 1868   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   
## 1875 1878 1879 1883 1885 1887 1894 1904 1912 1913 1921 1922 1923 1924 1929   
## 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0   
## 1930 1933 1934 1936 1939 1945 1948 1951 1954 1959 1963 1965 1968 1969 1972   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 1973 1974 1977 1982 1983 1984 1986 1987 1988 1990 1993 1997 1998 2003 2004   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2006 2007 2008 2023 2027 2029 2030 2034 2036 2041 2045 2047 2051 2052 2054   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0   
## 2059 2060 2067 2069 2071 2073 2076 2077 2078 2079 2080 2082 2088 2089 2090   
## 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0   
## 2094 2099 2101 2102 2107 2110 2112 2114 2115 2119 2122 2125 2129 2130 2134   
## 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0   
## 2136 2141 2142 2150 2152 2153 2156 2160 2163 2167 2179 2180 2182 2185 2192   
## 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2201 2203 2204 2208 2210 2214 2215 2218 2220 2226 2229 2238 2241 2244 2248   
## 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0   
## 2250 2252 2255 2264 2267 2269 2275 2283 2287 2289 2293 2295 2300 2303 2309   
## 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0   
## 2311 2312 2313 2317 2318 2319 2320 2327 2329 2332 2334 2335 2337 2339 2344   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2345 2347 2351 2354 2357 2358 2363 2364 2372 2373 2374 2383 2384 2385 2388   
## 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0   
## 2389 2390 2396 2398 2399 2400 2404 2406 2409 2414 2417 2418 2421 2422 2431   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   
## 2432 2435 2436 2437 2442 2446 2449 2452 2454 2455 2461 2464 2467 2468 2473   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2475 2476 2479 2480 2483 2484 2485 2486 2488 2492 2493 2494 2499 2504 2516   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2518 2522 2524 2525 2530 2533 2537 2538 2539 2543 2546 2547 2549 2556 2558   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2559 2562 2565 2573 2574 2575 2580 2591 2594 2597 2599 2601 2604 2607 2608   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2615 2621 2624 2634 2635 2636 2639 2640 2641 2644 2645 2646 2653 2654 2655   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0   
## 2656 2657 2659 2666 2668 2670 2675 2677 2679 2685 2686 2689 2692 2694 2699   
## 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0   
## 2701 2706 2707 2708 2712 2713 2716 2717 2724 2726 2727 2730 2737 2743 2745   
## 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0   
## 2746 2747 2751 2755 2757 2764 2769 2772 2773 2774 2775 2783 2797 2799 2803   
## 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0   
## 2805 2810 2811 2814 2815 2816 2817 2821 2824 2826 2830 2833 2834 2841 2849   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2851 2855 2859 2860 2864 2868 2869 2870 2871 2874 2875 2877 2879 2886 2888   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2895 2899 2903 2914 2915 2919 2923 2924 2925 2929 2933 2935 2936 2942 2951   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 2957 2961 2962 2963 2967 2968 2976 2977 2983 2989 2990 2993 2994 2995 2999   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 3000 3001 3008 3013 3017 3024 3031 3034 3038 3039 3042 3044 3046 3052 3054   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 3056 3058 3062 3063 3064 3065 3066 3068 3073 3079 3081 3083 3084 3085 3086   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 3089 3092 3093 3095 3097 3098 3099 3100 3105 3106 3111 3118 3119 3120 3122   
## 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 3127 3129 3131 3139 3140 3146 3148 3149 3151 3153 3154 3157 3159 3162 3166   
## 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0   
## 3171 3174 3176 3177 3178 3180 3181 3182 3185 3186 3189 3192 3193 3196 3202   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0   
## 3203 3204 3207 3209 3210 3213 3214 3216 3217 3222 3227 3231 3233 3234 3238   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 3239 3240 3242 3249 3250 3251 3252 3256 3259 3260 3263 3264 3267 3269 3275   
## 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0   
## 3279 3281 3283 3285 3288 3289 3291 3293 3295 3296 3299 3302 3303 3307 3309   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 3310 3320 3328 3332 3333 3336 3341 3345 3347 3350 3352 3354 3356 3360 3361   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1   
## 3367 3368 3370 3372 3377 3378 3384 3386 3387 3389 3392 3394 3400 3402 3403   
## 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0   
## 3404 3410 3412 3414 3416 3419 3420 3421 3422 3426 3427 3431 3437 3443 3447   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0   
## 3450 3453 3455 3456 3457 3458 3459 3464 3466 3469 3470 3474 3477 3478 3479   
## 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0   
## 3485 3489 3490 3492 3494 3502 3503 3505 3508 3510 3511 3516 3518 3520 3521   
## 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0   
## 3525 3526 3529 3530 3533 3535 3536 3545 3546 3553 3557 3558 3559 3560 3568   
## 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0   
## 3571 3574 3575 3582 3585 3587 3588 3589 3590 3592 3600 3602 3604 3605 3606   
## 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0   
## 3609 3613 3623 3625 3627 3629 3631 3633 3635 3636 3638 3646 3648 3654 3656   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0   
## 3657 3658 3660 3661 3664 3668 3670 3673 3674 3675 3683 3684 3691 3692 3694   
## 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0   
## 3695 3705 3707 3709 3717 3720 3726 3728 3731 3734 3735 3736 3738 3742 3744   
## 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1   
## 3747 3751 3754 3755 3760 3762 3764 3767 3770 3771 3772 3775 3777 3778 3782   
## 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0   
## 3784 3790 3792 3796 3798 3801 3804 3805 3808 3810 3812 3815 3819 3824 3827   
## 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0   
## 3828 3833 3844 3845 3847 3849 3855 3856 3858 3860 3864 3868 3871 3879 3880   
## 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0   
## 3882 3885 3890 3907 3912 3913 3914 3922 3923 3929 3937 3940 3941 3942 3943   
## 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0   
## 3944 3948 3950 3951 3958 3962 3965 3966 3967 3969 3975 3985 3986 3987 3989   
## 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0   
## 3991 3992 3994 4000 4003 4004 4005 4006 4008 4009 4012 4013 4015 4024 4026   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4028 4031 4032 4034 4035 4036 4037 4043 4046 4047 4050 4058 4059 4060 4069   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1   
## 4070 4072 4083 4085 4089 4091 4094 4100 4101 4106 4109 4110 4113 4117 4119   
## 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0   
## 4122 4123 4124 4125 4126 4130 4133 4135 4141 4142 4145 4146 4148 4149 4153   
## 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0   
## 4154 4156 4159 4162 4163 4172 4177 4178 4180 4181 4184 4188 4189 4190 4193   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4198 4202 4209 4210 4217 4219 4222 4224 4225 4228 4234 4238 4240 4242 4244   
## 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0   
## 4250 4254 4256 4257 4258 4259 4261 4262 4266 4267 4268 4271 4272 4275 4279   
## 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0   
## 4281 4282 4287 4292 4294 4296 4297 4299 4305 4306 4311 4312 4318 4320 4321   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4324 4325 4331 4336 4340 4345 4348 4353 4354 4360 4363 4368 4369 4370 4373   
## 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4374 4375 4377 4380 4383 4385 4386 4390 4391 4398 4407 4408 4412 4417 4418   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4419 4420 4422 4424 4430 4431 4435 4439 4442 4443 4444 4445 4449 4453 4454   
## 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4459 4461 4462 4465 4467 4468 4474 4477 4478 4479 4484 4488 4490 4491 4492   
## 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0   
## 4494 4496 4500 4503 4515 4518 4519   
## 0 1 0 0 0 0 0

Confusion matrix

cm = table(test\_set$y, y\_pred)  
cm

## y\_pred  
## 0 1  
## no 1302 31  
## yes 115 59

Accuracy is the ratio of correctly predicted observation to the total observations

accuracy <- sum(diag(cm))/sum(cm)   
accuracy

## [1] 0.9031188

So we got 90% accuracy Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

precision <- cm[2,2]/(sum(cm[,2]))  
precision

## [1] 0.6555556

We got 66% precision Recall is the ratio of correctly predicted positive observations to the all observations in actual class

recall <- cm[2,2]/(sum(cm[2,]))  
recall

## [1] 0.3390805

We got 34% recall

F1 <- 2\*(recall \* precision) / (recall + precision)  
F1

## [1] 0.4469697

45% is the F1 score