Classify whether application accepted or not using Logistic regression

card

Factor. Was the application for a credit card accepted?

reports

Number of major derogatory reports.

age

Age in years plus twelfths of a year.

income

Yearly income (in USD 10,000).

share

Ratio of monthly credit card expenditure to yearly income.

expenditure

Average monthly credit card expenditure.

owner

Factor. Does the individual own their home?

selfemp

Factor. Is the individual self-employed?

dependents

Number of dependents.

months

Months living at current address.

majorcards

Number of major credit cards held.

active

Number of active credit accounts.

**Ans:**

> view(credit\_card)

> summary(credit\_card)

X reports age income

Min. : 1.0 Min. : 0.0000 Min. : 0.1667 Min. : 0.210

1st Qu.: 330.5 1st Qu.: 0.0000 1st Qu.:25.4167 1st Qu.: 2.244

Median : 660.0 Median : 0.0000 Median :31.2500 Median : 2.900

Mean : 660.0 Mean : 0.4564 Mean :33.2131 Mean : 3.365

3rd Qu.: 989.5 3rd Qu.: 0.0000 3rd Qu.:39.4167 3rd Qu.: 4.000

Max. :1319.0 Max. :14.0000 Max. :83.5000 Max. :13.500

share expenditure dependents months

Min. :0.0001091 Min. : 0.000 Min. :0.0000 Min. : 0.00

1st Qu.:0.0023159 1st Qu.: 4.583 1st Qu.:0.0000 1st Qu.: 12.00

Median :0.0388272 Median : 101.298 Median :1.0000 Median : 30.00

Mean :0.0687322 Mean : 185.057 Mean :0.9939 Mean : 55.27

3rd Qu.:0.0936168 3rd Qu.: 249.036 3rd Qu.:2.0000 3rd Qu.: 72.00

Max. :0.9063205 Max. :3099.505 Max. :6.0000 Max. :540.00

majorcards active

Min. :0.0000 Min. : 0.000

1st Qu.:1.0000 1st Qu.: 2.000

Median :1.0000 Median : 6.000

Mean :0.8173 Mean : 6.997

3rd Qu.:1.0000 3rd Qu.:11.000

Max. :1.0000 Max. :46.000

> sum(is.na(credit\_card))

[1] 0

# NA values are not present in the dataset.

> str(affairs)

'data.frame': 1319 obs. of 10 variables:

$ X : int 1 2 3 4 5 6 7 8 9 10 ...

$ reports : int 0 0 0 0 0 0 0 0 0 0 ...

$ age : num 37.7 33.2 33.7 30.5 32.2 ...

$ income : num 4.52 2.42 4.5 2.54 9.79 ...

$ share : num 0.03327 0.00522 0.00416 0.06521 0.06705 ...

$ expenditure: num 124.98 9.85 15 137.87 546.5 ...

$ dependents : int 3 3 4 0 2 0 2 0 0 0 ...

$ months : int 54 34 58 25 64 54 7 77 97 65 ...

$ majorcards : int 1 1 1 1 1 1 1 1 1 1 ...

$ active : int 12 13 5 7 5 1 5 3 6 18 .**..**

> credit\_card <- na.omit(credit\_card)

> dim(credit\_card)

[1] 1319 10

> colnames(credit\_card)

[1] "X" "reports" "age" "income" "share"

[6] "expenditure" "dependents" "months" "majorcards" "active"

> credit\_card <- credit\_card[,-1]

> mod\_lm <- lm(majorcards ~.,data = credit\_card)

> pred1 <- predict(mod\_lm,credit\_card)

> pred1

1 2 3 4 5 6 7 8

0.8602863 0.8159779 0.7966009 0.8184194 0.9516240 0.7622921 0.8199860 0.7706497

9 10 11 12 13 14 15 16

0.8482946 0.8908038 0.9399808 0.7157949 0.8046105 0.7820543 0.7286296 0.8926248

17 18 19 20 21 22 23 24

0.7620040 0.7529440 0.9482062 0.8247953 0.7122173 0.7362957 0.7739261 0.9368213

25 26 27 28 29 30 31 32

0.9292550 0.8567142 0.7946490 0.7730771 0.7484557 0.9597542 0.7904356 0.8438037

33 34 35 36 37 38 39 40

0.7882971 0.8815787 0.8268080 0.8398421 0.7201798 0.9532983 0.9110352 0.8845882

41 42 43 44 45 46 47 48

0.8706749 0.7761103 0.7449467 0.8351510 0.7663034 0.7328871 0.7664372 0.7449237

49 50 51 52 53 54 55 56

0.7643795 0.9845004 0.7644646 0.8429087 0.7880428 0.7987042 0.7678171 0.7433017

57 58 59 60 61 62 63 64

0.8725903 0.8217106 0.7113043 0.7664759 0.7983273 0.7585844 0.8919402 0.7762708

65 66 67 68 69 70 71 72

0.8928762 0.7571590 0.8930856 0.7885387 0.8246372 0.8281855 0.8727372 0.8081498

73 74 75 76 77 78 79 80

0.7328398 0.8565356 0.7882093 0.8184895 0.8861693 0.7796550 0.8008188 0.8824401

81 82 83 84 85 86 87 88

0.7755799 0.8589501 0.8061131 0.7833229 0.8295051 0.9021285 0.7882011 0.8821164

89 90 91 92 93 94 95 96

0.7464435 0.6998192 0.7316186 0.7226099 0.7589259 0.7636590 0.9490282 0.7500965

97 98 99 100 101 102 103 104

0.8853070 0.8087418 0.9576083 0.8214304 1.1354128 0.8841453 0.7844203 0.8322651

105 106 107 108 109 110 111 112

0.7964428 0.8366594 0.8172136 0.7856282 1.0121245 1.0729899 0.9005418 0.8754596

113 114 115 116 117 118 119 120

0.9914841 1.0507849 0.8081217 0.8181709 0.7364694 0.7844412 0.7755248 0.8035464

121 122 123 124 125 126 127 128

0.9475300 0.7763601 0.8250063 0.7658656 0.7476567 0.8130582 0.9351546 0.8665757

129 130 131 132 133 134 135 136

0.7601012 0.7247102 0.8689458 0.7883809 0.7745571 0.7954157 0.7778947 0.7483231

137 138 139 140 141 142 143 144

0.7278952 0.8482802 0.8424462 0.8765273 0.8169806 0.8134865 0.8107400 0.7444765

145 146 147 148 149 150 151 152

0.8921097 0.7984485 0.7777056 0.8426034 0.8747550 1.0115905 0.8050134 0.8048206

153 154 155 156 157 158 159 160

0.7929299 0.8096132 0.7540262 0.7603165 0.8857157 0.7709534 0.8074350 0.9390187

161 162 163 164 165 166 167 168

0.7824363 0.8134459 0.7686133 0.7950053 0.8924483 0.8003598 0.8433491 0.9131390

169 170 171 172 173 174 175 176

0.8299406 0.7761433 0.7789213 0.7942817 0.7324385 0.8106440 0.8659597 0.8091023

177 178 179 180 181 182 183 184

0.8409401 0.7925083 0.7555895 0.8563815 1.0066184 0.7722945 0.9866926 0.7403981

185 186 187 188 189 190 191 192

0.7977749 0.8411369 0.7588436 0.7639465 0.7131392 0.8771322 0.7866016 0.8251581

193 194 195 196 197 198 199 200

0.8912623 0.7613788 0.8038506 0.7959952 0.7753636 0.7618574 0.8291723 0.8014184

201 202 203 204 205 206 207 208

0.8031154 0.8781806 0.7669782 0.7765943 0.8690486 0.7898283 0.8090328 0.7866200

209 210 211 212 213 214 215 216

0.8570937 0.8002790 0.8460226 1.0118710 0.7793610 0.7581258 0.8416692 0.9085929

217 218 219 220 221 222 223 224

0.7116398 0.9263420 0.8313171 0.8023174 0.8493070 0.7686479 0.8267209 0.8330528

225 226 227 228 229 230 231 232

0.7851001 0.7436484 0.8155425 0.8084159 0.9405673 0.8527395 0.8102314 0.8091812

233 234 235 236 237 238 239 240

0.8772496 0.8547559 0.8163894 0.8032367 0.7265477 0.7737177 0.7264313 0.7417856

241 242 243 244 245 246 247 248

0.8366161 0.7886028 0.7520040 0.7576815 0.7605301 0.7927396 0.8928242 0.8040318

249 250 251 252 253 254 255 256

0.7821320 0.8225110 0.7882545 0.9142092 0.8296100 0.8113805 0.8208557 0.8293508

257 258 259 260 261 262 263 264

0.9510799 1.1258159 0.8789713 0.8555633 0.7838080 0.8289455 0.9110003 0.8523853

265 266 267 268 269 270 271 272

0.9013139 0.7879488 0.7966820 0.7535409 0.8008159 0.7787187 0.8493124 0.7781560

273 274 275 276 277 278 279 280

0.8332785 0.7999278 0.8178026 0.8245031 0.7833646 0.8324316 0.9203243 0.8134836

281 282 283 284 285 286 287 288

0.8494640 0.8444610 0.7957827 0.7402178 0.8143157 0.9346640 0.8079627 0.8104145

289 290 291 292 293 294 295 296

0.7878204 0.7602417 0.9345694 0.8137643 0.8232920 0.8006988 0.7793983 0.9027881

297 298 299 300 301 302 303 304

0.7528014 0.9577601 0.8297243 0.8611238 0.8227557 0.8275326 0.7872426 0.8519648

305 306 307 308 309 310 311 312

0.7585678 0.6979644 0.7707022 0.7938021 1.0484567 0.7291301 0.8246719 0.8714739

313 314 315 316 317 318 319 320

0.8244801 0.7801114 0.8502898 0.8148027 0.8030035 0.9137078 0.7284034 0.7247008

321 322 323 324 325 326 327 328

0.7472991 0.7342617 0.9174718 0.9072286 0.7347814 0.6756811 0.8346383 0.8721152

329 330 331 332 333 334 335 336

0.8299447 0.8162308 0.8865051 0.8244255 0.8233679 0.7788785 0.9373114 0.7677246

337 338 339 340 341 342 343 344

0.7481746 0.7963251 0.7945928 0.7707704 0.8105865 0.7864336 0.8194031 0.8422232

345 346 347 348 349 350 351 352

0.7759833 0.7842824 0.7672568 0.7624634 0.8268554 0.9371847 0.7381517 1.1070936

353 354 355 356 357 358 359 360

0.8174896 0.7550905 0.8679936 0.8592714 0.8344723 0.7963306 1.1322788 0.9075864

361 362 363 364 365 366 367 368

0.8467403 0.8106359 0.8449498 0.8266624 0.7313779 0.7415921 0.7707292 0.8251782

369 370 371 372 373 374 375 376

0.8312690 0.7701931 0.9241025 0.7933315 0.7680805 0.8243888 0.7795789 0.8224771

377 378 379 380 381 382 383 384

0.8045337 0.8611748 0.7904336 0.8827088 0.7963448 0.7233879 0.7526073 0.7923572

385 386 387 388 389 390 391 392

0.7837796 0.8685610 0.8377504 0.7925897 0.7999453 0.7715292 0.8452606 0.8261363

393 394 395 396 397 398 399 400

0.8941795 0.8418006 0.8139762 0.8321779 0.7654491 0.8139069 0.9029637 0.7845957

401 402 403 404 405 406 407 408

0.8206881 0.8022755 0.7804471 0.9515666 0.8162132 0.8579936 0.9228290 0.7900259

409 410 411 412 413 414 415 416

0.8258832 0.7564692 0.8576235 0.9916569 0.8360513 0.6493345 0.7973803 0.8090020

417 418 419 420 421 422 423 424

0.8352007 0.7971128 0.8162135 0.9314037 0.7679209 0.8987393 0.8814755 0.8123593

425 426 427 428 429 430 431 432

0.8623385 0.7784433 0.7822272 0.7619998 0.8405247 0.8479681 0.8663651 0.8366275

433 434 435 436 437 438 439 440

0.8808072 0.8617657 0.7728108 0.8350168 0.8162810 0.7518396 0.7864827 0.7347365

441 442 443 444 445 446 447 448

0.7884104 0.7822167 0.7862930 0.7579571 0.7380796 0.8681733 1.1016458 0.8476543

449 450 451 452 453 454 455 456

0.8663458 0.7842519 0.8305501 0.7268324 0.8106632 0.8548684 0.7219436 0.8257955

457 458 459 460 461 462 463 464

0.8249066 0.8269191 0.9789482 0.8030384 0.9835865 0.7921511 0.8043476 0.7647725

465 466 467 468 469 470 471 472

0.7702363 0.7454612 0.9728989 0.8968026 0.8306988 0.7254587 0.7571439 0.8974681

473 474 475 476 477 478 479 480

0.7812431 0.6550326 0.8651375 0.7986273 0.7898451 0.9658842 0.8144454 0.7264319

481 482 483 484 485 486 487 488

0.8004877 0.8201797 0.8794673 0.7961751 0.7575240 0.8987712 0.8157910 0.8113509

489 490 491 492 493 494 495 496

0.7708518 0.7823925 0.8390468 0.7388866 0.7451959 0.8512866 0.7798460 0.7834422

497 498 499 500 501 502 503 504

0.7749677 0.8110563 0.8161506 0.7750858 0.8252452 0.8842492 0.8712238 0.7860024

505 506 507 508 509 510 511 512

0.7704995 0.8320492 0.8439619 0.7788943 0.8225471 0.7881231 0.8342396 0.8023351

513 514 515 516 517 518 519 520

0.7526312 0.8926230 1.0162231 0.8603287 0.8142043 0.8574449 0.9504248 0.8749490

521 522 523 524 525 526 527 528

0.8798078 0.8251634 0.9216521 0.8794459 0.7802163 0.8108042 0.8282413 0.7489555

529 530 531 532 533 534 535 536

0.8482129 0.7367131 0.8593234 0.8539584 0.7593162 0.8246758 0.8346416 0.7946483

537 538 539 540 541 542 543 544

0.7420116 0.7841324 0.7289600 0.9942025 0.8422576 0.9918383 0.9946691 0.7989447

545 546 547 548 549 550 551 552

0.7491481 0.8394382 0.8765990 0.9034092 0.7231302 0.7914414 0.7936663 0.8108812

553 554 555 556 557 558 559 560

0.8300934 0.7444578 0.8206943 0.7617193 0.7650341 0.7828462 0.8367512 0.8810706

561 562 563 564 565 566 567 568

0.7836014 0.7358130 0.8969881 0.7816494 0.8181072 0.6940948 0.8894036 0.8026620

569 570 571 572 573 574 575 576

0.7759452 0.8528377 0.8348728 0.8202393 0.8238122 0.7719382 0.7669896 0.7848299

577 578 579 580 581 582 583 584

0.7341784 0.7047853 0.8409686 0.8883474 0.7577740 0.8482812 0.8187763 0.7389082

585 586 587 588 589 590 591 592

0.7671587 0.7657494 0.9395305 0.7799740 0.7538995 0.7449754 0.7284004 0.8773001

593 594 595 596 597 598 599 600

0.8036066 0.7793697 0.7772036 0.8650039 0.8399684 0.6934599 0.8312691 0.7811146

601 602 603 604 605 606 607 608

0.7673097 0.7673181 0.8425249 0.8334057 0.8063906 0.8419336 0.9772204 0.7808619

609 610 611 612 613 614 615 616

0.8853561 0.7736490 0.7370296 0.7900390 0.8671670 0.7754950 0.7954534 0.7663126

617 618 619 620 621 622 623 624

0.7747254 0.7213009 0.7799228 0.8064854 0.7790705 0.7956035 0.7987300 1.0032170

625 626 627 628 629 630 631 632

0.9197377 0.8211929 0.7885313 0.9208584 0.9109464 0.7675600 0.8138230 0.8502073

633 634 635 636 637 638 639 640

0.8204321 0.8351830 0.7389323 0.8062643 0.7615947 0.8941575 0.8347830 0.7632629

641 642 643 644 645 646 647 648

0.7977641 0.8972152 0.7937294 0.7432774 0.8612517 0.8060722 0.7932887 0.7878666

649 650 651 652 653 654 655 656

0.8390218 0.7849413 0.8161604 0.7816266 0.8326950 0.7910285 0.9162396 0.8183767

657 658 659 660 661 662 663 664

0.8289182 1.0333967 0.7698664 0.7748257 0.8322965 0.7403547 0.8045628 0.8098005

665 666 667 668 669 670 671 672

0.8757050 0.8252826 0.7657787 0.8390642 0.7932884 0.7611717 0.8875815 0.9450765

673 674 675 676 677 678 679 680

0.8814502 0.8192408 0.7976751 0.8229708 0.8070724 0.9097680 0.8039295 0.7123525

681 682 683 684 685 686 687 688

0.8955676 0.7746468 0.7567914 0.8294238 0.8383696 0.8933332 0.7893743 0.7733596

689 690 691 692 693 694 695 696

0.8905206 0.7612512 0.7628721 0.8603822 0.8499141 0.9195219 0.8702205 0.7350023

697 698 699 700 701 702 703 704

0.8341559 0.6552895 0.9054170 0.8500065 0.8042283 0.8074556 0.8724157 0.7549245

705 706 707 708 709 710 711 712

0.9056261 0.8376783 0.7044400 0.8918697 0.7616335 0.7279227 0.8688666 0.7282805

713 714 715 716 717 718 719 720

0.7521308 0.8854430 0.7701563 0.8059044 0.8278885 0.7898039 0.8330945 0.8059270

721 722 723 724 725 726 727 728

0.8201750 0.8045745 0.8534858 0.8444058 0.8415428 0.7517323 0.8318462 0.7878344

729 730 731 732 733 734 735 736

0.8292562 0.8317395 0.8607517 0.8841847 0.7506376 0.8370608 0.7287699 0.7684597

737 738 739 740 741 742 743 744

0.8868786 0.8220562 0.7996838 0.8190330 0.6969947 0.7144040 0.8049190 0.7408093

745 746 747 748 749 750 751 752

0.7803706 0.7634683 0.7663096 0.7563531 0.8673712 0.7504288 0.7677992 0.8084842

753 754 755 756 757 758 759 760

0.7279628 0.8386502 0.7269723 0.8351903 0.8519327 0.7978040 0.8194533 0.8850113

761 762 763 764 765 766 767 768

0.8355875 0.8956157 0.8881268 0.9395420 0.9708667 0.8148153 0.7706618 0.7509388

769 770 771 772 773 774 775 776

0.8212417 0.9335206 0.7601460 0.7833496 0.8712137 0.9805962 0.7402701 0.7802737

777 778 779 780 781 782 783 784

0.7853677 0.8707605 0.8997962 0.8396370 0.7717904 0.8417404 0.7952103 0.7941663

785 786 787 788 789 790 791 792

0.7579851 0.7605485 0.8077527 0.6943535 0.8223062 0.8731705 0.8337631 0.8294423

793 794 795 796 797 798 799 800

0.8236643 0.7724293 0.8322927 0.7997955 0.8090677 0.9218594 0.7256405 0.7469453

801 802 803 804 805 806 807 808

0.6653729 0.7779949 0.7586471 0.8022288 0.7431521 0.8524246 0.8615056 0.8414305

809 810 811 812 813 814 815 816

0.7828859 0.8324008 0.7780918 0.8436868 0.8273145 0.8609606 0.8187250 0.7730889

817 818 819 820 821 822 823 824

0.7723570 0.7795272 0.7792833 0.8609198 0.9333431 0.7567080 0.8529425 0.8896338

825 826 827 828 829 830 831 832

0.7513273 0.7766576 0.9042520 0.8432182 0.7314508 0.8194274 0.8310124 0.8079860

833 834 835 836 837 838 839 840

0.9816203 0.7512703 0.9702992 0.7812398 0.9261359 0.8232194 0.7728261 0.7817252

841 842 843 844 845 846 847 848

0.7069413 0.6406243 0.8207868 0.7774458 0.7987274 0.8858372 0.9071736 0.7754975

849 850 851 852 853 854 855 856

0.8176367 0.8114911 0.8304841 0.8322165 0.8303750 0.8135621 0.7573323 0.8089805

857 858 859 860 861 862 863 864

0.7306505 0.7509973 0.8430850 0.7932433 0.7502251 0.8003251 0.8209949 0.6946783

865 866 867 868 869 870 871 872

0.7147452 0.8569187 0.8418598 0.7134662 0.8056525 0.7777798 0.7529117 0.8816570

873 874 875 876 877 878 879 880

0.7839144 0.8887965 0.8290878 0.7332318 0.7401383 0.7834209 0.8413704 0.8194846

881 882 883 884 885 886 887 888

0.7565048 0.7675807 0.7119787 0.7505087 0.9125625 0.7140122 0.8255397 0.8098912

889 890 891 892 893 894 895 896

0.7718860 0.7289898 0.8567324 0.8485337 0.8546169 0.6682103 0.8227901 0.7664120

897 898 899 900 901 902 903 904

0.7962895 0.8338102 0.8008513 0.8484939 0.7748094 0.6774190 0.7848154 0.7724935

905 906 907 908 909 910 911 912

0.7671817 0.9449712 0.7899213 0.7482940 0.7078107 0.7473826 0.7290410 0.8106812

913 914 915 916 917 918 919 920

0.7528192 0.7382854 0.7868930 0.7787693 0.8193008 0.8436889 0.7516554 0.7368721

921 922 923 924 925 926 927 928

0.8231959 0.8410249 0.7695170 0.7433256 0.7828228 0.9651776 0.7109405 0.8000248

929 930 931 932 933 934 935 936

0.8167437 0.8015940 0.7685796 0.7343975 0.7122246 0.8107081 0.8421722 0.7672473

937 938 939 940 941 942 943 944

0.7358614 0.8226287 0.7603635 0.8472418 0.8570026 0.8338751 0.7751841 0.8551722

945 946 947 948 949 950 951 952

0.7526796 0.8049328 0.8202226 0.8310803 0.7163947 0.7358166 0.7710955 0.7852980

953 954 955 956 957 958 959 960

0.7645889 0.7932490 0.8413711 0.8704788 0.8468365 0.7892793 0.6995069 0.7464807

961 962 963 964 965 966 967 968

0.9150813 0.8432918 0.9009923 0.8307709 0.7262089 0.8030494 0.7599036 0.9880570

969 970 971 972 973 974 975 976

0.8076826 0.8179035 0.7915403 0.8174293 0.8139559 0.7099183 0.9311882 0.8506759

977 978 979 980 981 982 983 984

0.8059087 0.8116096 0.7755137 0.9655697 0.7717819 0.7233939 0.7405844 0.8507101

985 986 987 988 989 990 991 992

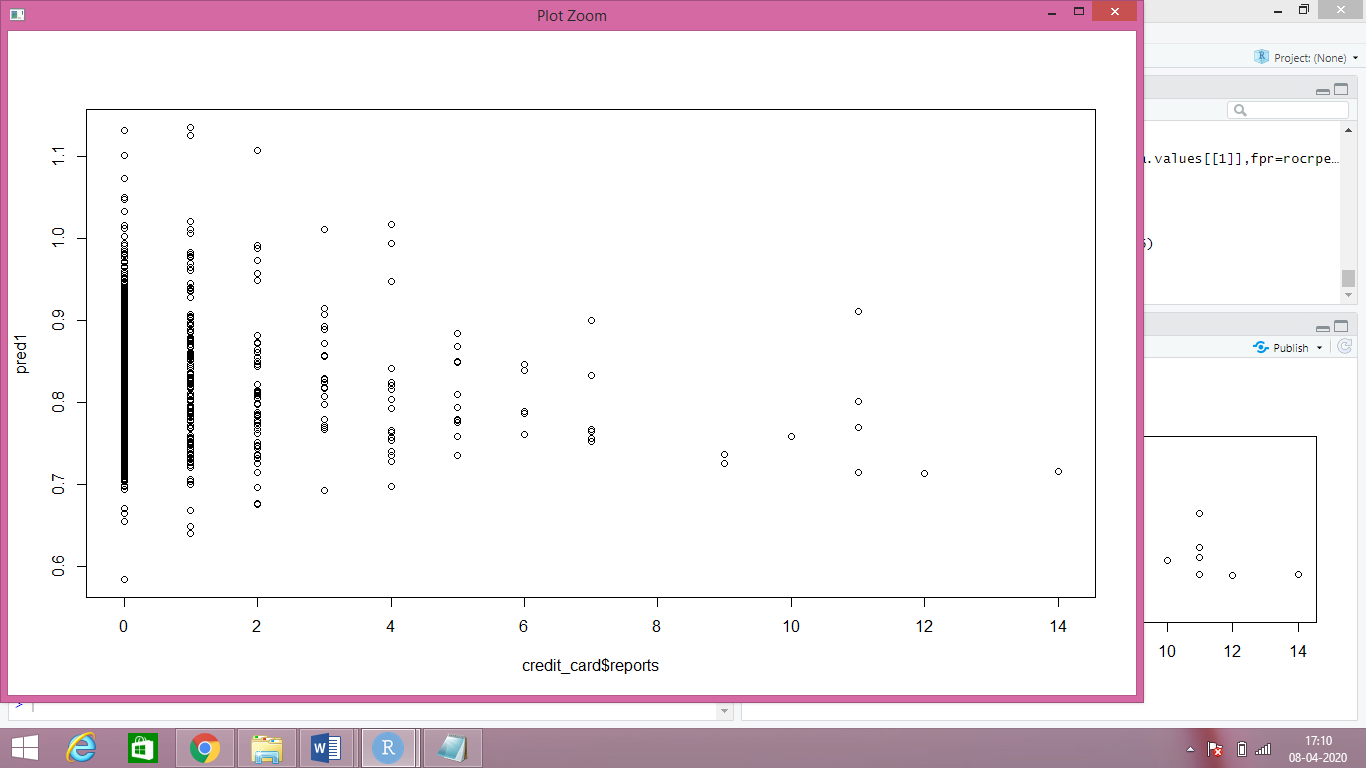
0.7743201 0.8678539 0.7127118 0.7064936 0.8487200 0.7567944 0.6984673 0.9270533

993 994 995 996 997 998 999 1000

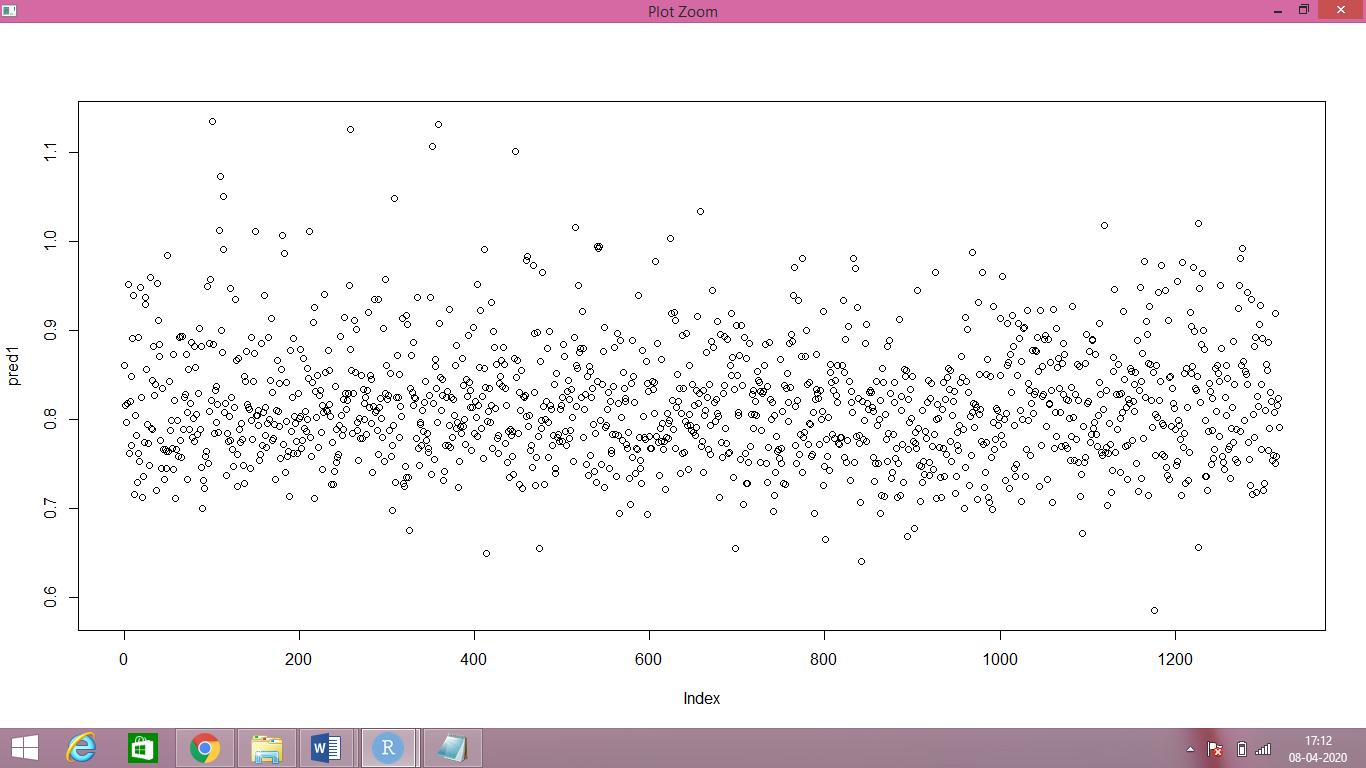
0.7966780 0.7657423 0.7777618 0.8120430 0.8099844 0.7928905 0.8021690 0.9130459

[ reached getOption("max.print") -- omitted 319 entries ]

> plot(credit\_card$reports,pred1)



> plot(pred1)



> model<-glm(majorcards ~.,data = credit\_card,family = “binomial”)

> exp(coef(model))

(Intercept) reports age income share expenditure dependents

1.9777267 0.9578252 0.9976582 1.2262390 3.2842253 1.0001530 0.9326763

months active

0.9977143 1.0588142

> prob <- predict(model,credit\_card,type=”response”)

> summary(model)

Call:

glm(formula = majorcards ~ ., family = "binomial", data = credit\_card)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.5649 0.4089 0.5812 0.6923 1.0434

Coefficients:

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

(Intercept) 0.6819480 0.3038002 2.245 0.02479 \*

reports -0.0430900 0.0534971 -0.805 0.42055

age -0.0023446 0.0083305 -0.281 0.77837

income 0.2039518 0.0691697 2.949 0.00319 \*\*

share 1.1891308 2.2149337 0.537 0.59136

expenditure 0.0001530 0.0009021 0.170 0.86533

dependents -0.0696971 0.0628329 -1.109 0.26732

months -0.0022884 0.0011444 -2.000 0.04554 \*

active 0.0571496 0.0141979 4.025 5.69e-05 \*\*\*

---

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1254.3 on 1318 degrees of freedom

Residual deviance: 1209.5 on 1310 degrees of freedom

AIC: 1227.5

Number of Fisher Scoring iterations: 5

> confusion <- table(prob>0.5,credit\_card$reports)

> confusion

0 1 2 3 4 5 6 7 9 10 11 12 14

FALSE 1 0 0 0 0 0 0 0 0 0 0 0 0

TRUE 1059 137 50 24 17 11 5 6 2 1 4 1 1

> Accuracy <- sum(confusion)/sum(confusion))

> Accuracy

[1] 0.1046247

> pred\_values <- NULL

> yes\_no <- NULL

> pred\_values <- ifelse(prob>=0.5,1,0)

> yes\_no <- ifelse(prob>=0.5,"yes","no")

> credit\_card[,"prob"] <- prob

> credit\_card[,"pred\_values"] <- pred\_values

> credit\_card[,"yes\_no"] <- yes\_no

> View(credit\_card[,c(1,7:9)])

> table(credit\_card$reports,credit\_card$pred\_values)

0 1

0 1 1059

1 0 137

2 0 50

3 0 24

4 0 17

5 0 11

6 0 5

7 0 6

9 0 2

10 0 1

11 0 4

12 0 1

14 0 1

# ROC Curve => used to evaluate the betterness of the logistic model

# more area under ROC curve better is the model

# We will use ROC curve for any classification technique not only for logistic

# To plot ROC curve:

> library(ROCR)

> rocrpred <- prediction(prob,credit\_card$majorcards)

> rocrperf <- performance(rocrpred,'tpr','fpr')

> str(rocrperf)

Formal class 'performance' [package "ROCR"] with 6 slots

..@ x.name : chr "False positive rate"

..@ y.name : chr "True positive rate"

..@ alpha.name : chr "Cutoff"

..@ x.values :List of 1

.. ..$ : num [1:1320] 0 0 0 0 0 0 0 0 0 0 ...

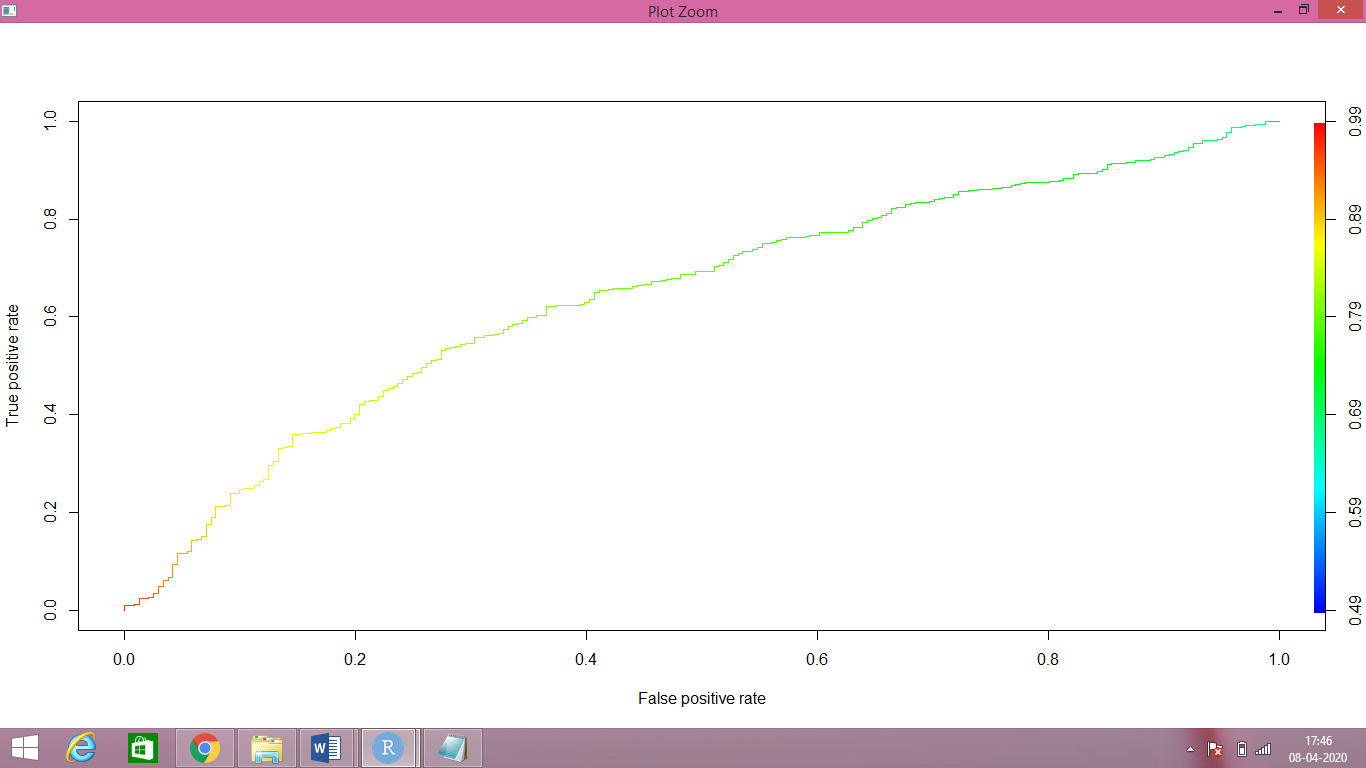
..@ y.values :List of 1

.. ..$ : num [1:1320] 0 0.000928 0.001855 0.002783 0.003711 ...

..@ alpha.values:List of 1

.. ..$ : num [1:1320] Inf 0.992 0.987 0.986 0.985 ...

> plot(rocrperf,colorize=T,text.adj=c(-0.2,1.7))



> rocr\_cutoff <- data.frame(cut\_off = rocrperf@alpha.values[[1]],fpr=rocrperf@x.values,tpr=rocrperf@y.values

> colnames(rocr\_cutoff) <- c("cut\_off","FPR","TPR")

> View(rocr\_cutoff)

> head(rocr\_cutoff)

cut\_off FPR TPR

1 0.580240 0.9917012 1.0000000

2 0.578216 0.9958506 1.0000000

3 0.490954 1.0000000 1.0000000

4 0.608130 0.9875519 0.9990724

5 0.592822 0.9917012 0.9990724

6 0.614419 0.9875519 0.9981447

# Sorting data frame with respect to tpr in decreasing order

> library(dplyr)

> rocr\_cutoff$cut\_off <- round(rocr\_cutoff$cut\_off,6)

> rocr\_cutoff <- arrange(rocr\_cutoff,desc(TPR))

> View(rocr\_cutoff)

> head(rocr\_cutoff,n = 15)

cut\_off FPR TPR

1 0.580240 0.9917012 1.0000000

2 0.578216 0.9958506 1.0000000

3 0.490954 1.0000000 1.0000000

4 0.608130 0.9875519 0.9990724

5 0.592822 0.9917012 0.9990724

6 0.614419 0.9875519 0.9981447

7 0.618438 0.9875519 0.9972171

8 0.624482 0.9875519 0.9962894

9 0.627658 0.9875519 0.9953618

10 0.648433 0.9834025 0.9944341

11 0.645143 0.9875519 0.9944341

12 0.656732 0.9792531 0.9935065

13 0.650713 0.9834025 0.9935065

14 0.656735 0.9792531 0.9925788

15 0.663077 0.9751037 0.9916512