Hybrid

Atmik

3/15/2022

Importing Libraries

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4  
## v tibble 3.1.6 v dplyr 1.0.8  
## v tidyr 1.2.0 v stringr 1.4.0  
## v readr 2.1.2 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(tseries)

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

library(forecast)  
library(ggplot2)  
library(caTools)

## Warning: package 'caTools' was built under R version 4.1.3

library(rpart)

## Warning: package 'rpart' was built under R version 4.1.3

library(syuzhet)

## Warning: package 'syuzhet' was built under R version 4.1.3

library(randomForest)

## Warning: package 'randomForest' was built under R version 4.1.3

## randomForest 4.7-1

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':  
##   
## combine

## The following object is masked from 'package:ggplot2':  
##   
## margin

Reading the file and Data Preprocessing

df <- read.csv(file.choose())  
start <- which(rownames(df) == "January-2012")  
end <- which(rownames(df) == "December-2019")   
df <- df[start:end,]  
df$MonthsNew <- rownames(df)  
df <- df[,c(4,6)]  
colnames(df) <- c("Close", "Month")  
df<- df[,c(2,1)]  
rownames(df) <- NULL  
df %>% separate(Month, c("Month","Year"), sep = "-" )

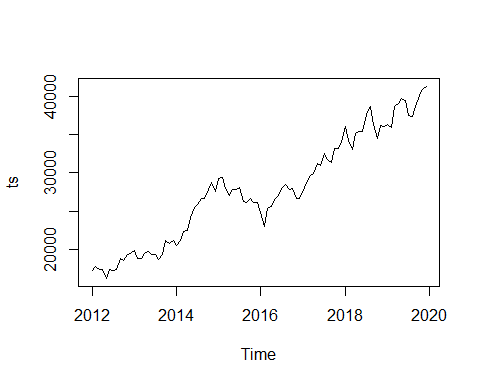
## Month Year Close  
## 1 January 2012 17193.55  
## 2 February 2012 17752.68  
## 3 March 2012 17404.20  
## 4 April 2012 17318.81  
## 5 May 2012 16218.53  
## 6 June 2012 17429.98  
## 7 July 2012 17236.18  
## 8 August 2012 17429.56  
## 9 September 2012 18762.74  
## 10 October 2012 18505.38  
## 11 November 2012 19339.90  
## 12 December 2012 19426.71  
## 13 January 2013 19894.98  
## 14 February 2013 18861.54  
## 15 March 2013 18835.77  
## 16 April 2013 19504.18  
## 17 May 2013 19760.30  
## 18 June 2013 19395.81  
## 19 July 2013 19345.70  
## 20 August 2013 18619.72  
## 21 September 2013 19379.77  
## 22 October 2013 21164.52  
## 23 November 2013 20791.93  
## 24 December 2013 21170.68  
## 25 January 2014 20513.85  
## 26 February 2014 21120.12  
## 27 March 2014 22386.27  
## 28 April 2014 22417.80  
## 29 May 2014 24217.34  
## 30 June 2014 25413.78  
## 31 July 2014 25894.97  
## 32 August 2014 26638.11  
## 33 September 2014 26630.51  
## 34 October 2014 27865.83  
## 35 November 2014 28693.99  
## 36 December 2014 27499.42  
## 37 January 2015 29182.95  
## 38 February 2015 29361.50  
## 39 March 2015 27957.49  
## 40 April 2015 27011.31  
## 41 May 2015 27828.44  
## 42 June 2015 27780.83  
## 43 July 2015 28114.56  
## 44 August 2015 26283.09  
## 45 September 2015 26154.83  
## 46 October 2015 26656.83  
## 47 November 2015 26145.67  
## 48 December 2015 26117.54  
## 49 January 2016 24870.69  
## 50 February 2016 23002.00  
## 51 March 2016 25341.86  
## 52 April 2016 25606.62  
## 53 May 2016 26667.96  
## 54 June 2016 26999.72  
## 55 July 2016 28051.86  
## 56 August 2016 28452.17  
## 57 September 2016 27865.96  
## 58 October 2016 27930.21  
## 59 November 2016 26652.81  
## 60 December 2016 26626.46  
## 61 January 2017 27655.96  
## 62 February 2017 28743.32  
## 63 March 2017 29620.50  
## 64 April 2017 29918.40  
## 65 May 2017 31145.80  
## 66 June 2017 30921.61  
## 67 July 2017 32514.94  
## 68 August 2017 31730.49  
## 69 September 2017 31283.72  
## 70 October 2017 33213.13  
## 71 November 2017 33149.35  
## 72 December 2017 34056.83  
## 73 January 2018 35965.02  
## 74 February 2018 34184.04  
## 75 March 2018 32968.68  
## 76 April 2018 35160.36  
## 77 May 2018 35322.38  
## 78 June 2018 35423.48  
## 79 July 2018 37606.58  
## 80 August 2018 38645.07  
## 81 September 2018 36227.14  
## 82 October 2018 34442.05  
## 83 November 2018 36194.30  
## 84 December 2018 36068.33  
## 85 January 2019 36256.69  
## 86 February 2019 35867.44  
## 87 March 2019 38672.91  
## 88 April 2019 39031.55  
## 89 May 2019 39714.20  
## 90 June 2019 39394.64  
## 91 July 2019 37481.12  
## 92 August 2019 37332.79  
## 93 September 2019 38667.33  
## 94 October 2019 40129.05  
## 95 November 2019 40793.81  
## 96 December 2019 41253.74

Converting into Time Series

ts <- ts(df$Close, freq = 12, start = c(2012, 1), end = c(2019, 12))  
ts

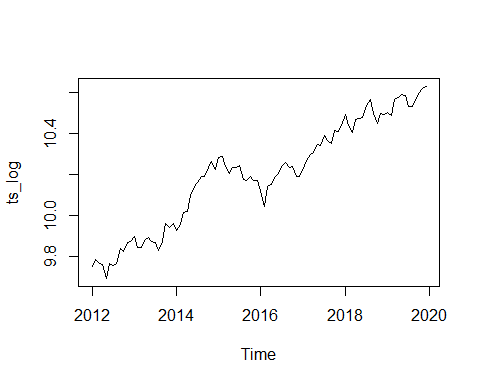
## Jan Feb Mar Apr May Jun Jul Aug  
## 2012 17193.55 17752.68 17404.20 17318.81 16218.53 17429.98 17236.18 17429.56  
## 2013 19894.98 18861.54 18835.77 19504.18 19760.30 19395.81 19345.70 18619.72  
## 2014 20513.85 21120.12 22386.27 22417.80 24217.34 25413.78 25894.97 26638.11  
## 2015 29182.95 29361.50 27957.49 27011.31 27828.44 27780.83 28114.56 26283.09  
## 2016 24870.69 23002.00 25341.86 25606.62 26667.96 26999.72 28051.86 28452.17  
## 2017 27655.96 28743.32 29620.50 29918.40 31145.80 30921.61 32514.94 31730.49  
## 2018 35965.02 34184.04 32968.68 35160.36 35322.38 35423.48 37606.58 38645.07  
## 2019 36256.69 35867.44 38672.91 39031.55 39714.20 39394.64 37481.12 37332.79  
## Sep Oct Nov Dec  
## 2012 18762.74 18505.38 19339.90 19426.71  
## 2013 19379.77 21164.52 20791.93 21170.68  
## 2014 26630.51 27865.83 28693.99 27499.42  
## 2015 26154.83 26656.83 26145.67 26117.54  
## 2016 27865.96 27930.21 26652.81 26626.46  
## 2017 31283.72 33213.13 33149.35 34056.83  
## 2018 36227.14 34442.05 36194.30 36068.33  
## 2019 38667.33 40129.05 40793.81 41253.74

plot(ts)

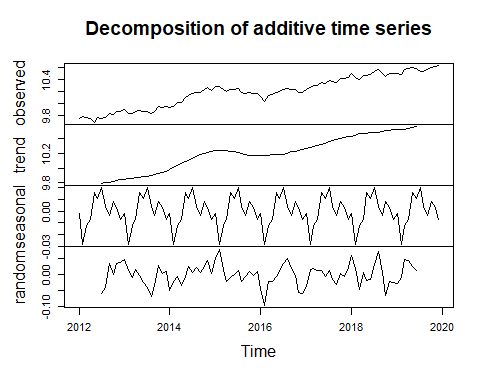


Making the Time Series Stationary and Plotting Different Components of Time Series

ts\_log <- log(ts)  
plot(ts\_log)



decomp <- decompose(ts\_log)  
plot(decomp)

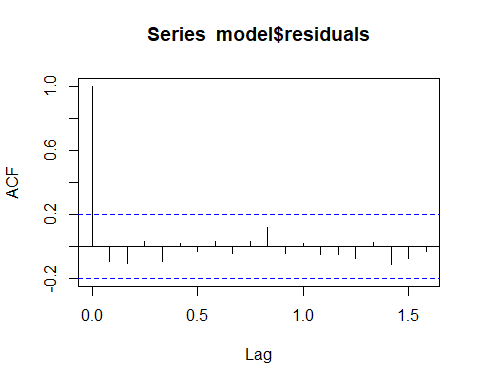


Fitting Arima Model

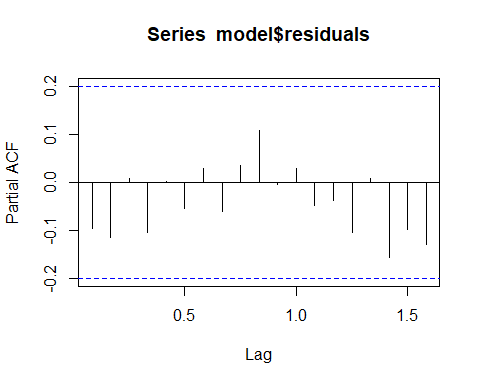
model <- auto.arima(ts\_log)  
model

## Series: ts\_log   
## ARIMA(0,1,0)(1,0,0)[12] with drift   
##   
## Coefficients:  
## sar1 drift  
## -0.2209 0.0091  
## s.e. 0.1038 0.0031  
##   
## sigma^2 = 0.001367: log likelihood = 179.21  
## AIC=-352.42 AICc=-352.16 BIC=-344.76

acf(model$residuals)



pacf(model$residuals)



Box.test(model$residuals, type = 'Ljung-Box')

##   
## Box-Ljung test  
##   
## data: model$residuals  
## X-squared = 0.91306, df = 1, p-value = 0.3393

adf.test(diff(ts\_log))

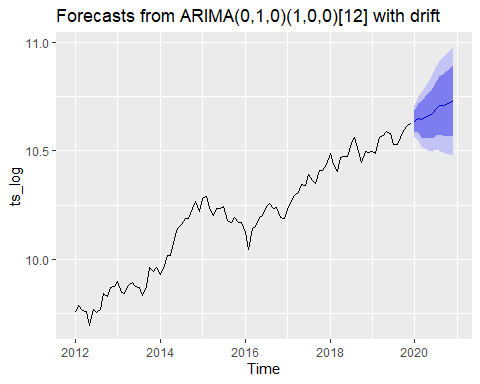
## Warning in adf.test(diff(ts\_log)): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: diff(ts\_log)  
## Dickey-Fuller = -4.4926, Lag order = 4, p-value = 0.01  
## alternative hypothesis: stationary

f <- forecast(model, 12)  
f

## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## Jan 2020 10.63745 10.59007 10.68484 10.56498 10.70992  
## Feb 2020 10.65094 10.58393 10.71795 10.54845 10.75342  
## Mar 2020 10.64541 10.56333 10.72748 10.51989 10.77093  
## Apr 2020 10.65447 10.55970 10.74924 10.50953 10.79941  
## May 2020 10.66175 10.55579 10.76770 10.49970 10.82379  
## Jun 2020 10.67463 10.55856 10.79070 10.49712 10.85215  
## Jul 2020 10.69673 10.57137 10.82210 10.50500 10.88847  
## Aug 2020 10.70871 10.57469 10.84274 10.50374 10.91369  
## Sep 2020 10.71206 10.56990 10.85421 10.49465 10.92947  
## Oct 2020 10.71497 10.56512 10.86481 10.48580 10.94414  
## Nov 2020 10.72244 10.56528 10.87960 10.48209 10.96280  
## Dec 2020 10.73107 10.56692 10.89522 10.48003 10.98211

autoplot(f)



NLP

news <- read.csv(file.choose(),stringsAsFactors = F)  
news2 <- news

news$publish\_date <- as.Date(as.character(news$publish\_date), "%Y%m%d")  
news$headline\_category <- NULL  
news <- news[news$publish\_date > as.Date("2012-01-01") & news$publish\_date < as.Date("2019-12-31"),]  
news <- separate(news, publish\_date, c("Year","Month","Day"), "-")  
news$Day <- NULL  
news$time <- paste(news$Month, "-", news$Year)   
news$Month <- NULL  
news$Year <- NULL  
news2 <- news

news3 <- news %>% group\_by(time) %>% summarise(headline\_text=paste(headline\_text, collapse = " "))

text <- as.character(news3$headline\_text)  
class(news3$headline\_text)

## [1] "character"

s <- get\_nrc\_sentiment(text)

## Warning: `spread\_()` was deprecated in tidyr 1.2.0.  
## Please use `spread()` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was generated.

head(s)

## anger anticipation disgust fear joy sadness surprise trust negative positive  
## 1 475 439 336 602 332 482 254 562 1168 1003  
## 2 497 441 356 590 346 461 251 586 1200 1070  
## 3 515 439 342 622 350 478 269 590 1172 1069  
## 4 519 463 359 637 371 492 276 614 1240 1099  
## 5 507 460 351 643 356 483 279 604 1230 1111  
## 6 509 458 361 627 364 490 265 631 1238 1107

final <- data.frame(df$Month, s, df$Close)  
final$PrevClose[2:96]<- final$df.Close[1:95]  
final$PrevClose[1] <- 15454.92  
final$Change <- (final$df.Close - final$PrevClose)\*100/final$PrevClose   
final$Change <- round(final$Change, 2)  
final$Class <- ifelse(final$Change > 0, 1 , 0)  
final

## df.Month anger anticipation disgust fear joy sadness surprise trust  
## 1 January-2012 475 439 336 602 332 482 254 562  
## 2 February-2012 497 441 356 590 346 461 251 586  
## 3 March-2012 515 439 342 622 350 478 269 590  
## 4 April-2012 519 463 359 637 371 492 276 614  
## 5 May-2012 507 460 351 643 356 483 279 604  
## 6 June-2012 509 458 361 627 364 490 265 631  
## 7 July-2012 539 463 395 666 367 499 271 609  
## 8 August-2012 440 398 324 539 308 422 234 531  
## 9 September-2012 490 430 340 596 350 470 250 587  
## 10 October-2012 485 434 353 606 347 478 267 581  
## 11 November-2012 486 403 335 577 321 445 255 586  
## 12 December-2012 482 434 346 575 352 465 258 597  
## 13 January-2013 491 457 356 604 351 469 263 620  
## 14 February-2013 525 444 364 646 357 495 267 609  
## 15 March-2013 495 440 359 639 350 487 264 584  
## 16 April-2013 432 370 314 533 285 424 216 509  
## 17 May-2013 495 426 343 614 335 499 251 572  
## 18 June-2013 486 439 353 604 342 487 254 588  
## 19 July-2013 475 430 330 570 344 434 250 569  
## 20 August-2013 516 472 366 640 363 502 266 613  
## 21 September-2013 520 457 391 666 374 508 290 617  
## 22 October-2013 534 454 371 659 347 528 257 622  
## 23 November-2013 516 450 374 645 362 510 270 606  
## 24 December-2013 456 393 331 549 297 440 223 519  
## 25 January-2014 496 435 351 621 336 482 261 576  
## 26 February-2014 506 455 360 638 344 494 266 578  
## 27 March-2014 456 431 323 555 337 439 250 558  
## 28 April-2014 529 453 364 644 364 499 278 611  
## 29 May-2014 527 465 371 628 369 495 269 622  
## 30 June-2014 533 470 379 669 367 510 277 626  
## 31 July-2014 515 452 384 637 352 510 254 589  
## 32 August-2014 462 386 334 573 308 447 223 534  
## 33 September-2014 496 426 352 611 324 477 247 583  
## 34 October-2014 492 417 347 626 330 490 252 584  
## 35 November-2014 503 441 354 628 353 476 266 576  
## 36 December-2014 518 454 366 636 354 505 271 603  
## 37 January-2015 523 463 368 653 355 517 277 593  
## 38 February-2015 549 456 388 669 357 527 270 603  
## 39 March-2015 525 471 373 650 364 515 266 603  
## 40 April-2015 466 410 340 577 315 453 241 543  
## 41 May-2015 489 438 353 602 334 479 250 564  
## 42 June-2015 478 410 343 603 322 476 254 578  
## 43 July-2015 490 442 360 618 336 511 251 576  
## 44 August-2015 512 459 373 643 355 520 260 609  
## 45 September-2015 494 437 353 620 338 497 258 598  
## 46 October-2015 525 457 367 655 353 501 269 603  
## 47 November-2015 529 434 391 655 339 521 268 599  
## 48 December-2015 430 375 314 558 308 442 226 513  
## 49 January-2016 496 424 359 616 332 497 253 583  
## 50 February-2016 507 416 372 633 308 488 262 584  
## 51 March-2016 511 453 361 628 359 486 282 605  
## 52 April-2016 515 451 386 646 361 515 271 592  
## 53 May-2016 504 445 369 635 352 504 273 613  
## 54 June-2016 529 465 375 656 367 530 276 617  
## 55 July-2016 532 451 391 674 351 518 266 614  
## 56 August-2016 440 404 335 566 303 448 246 523  
## 57 September-2016 512 422 376 641 333 511 264 595  
## 58 October-2016 486 428 355 624 332 488 246 583  
## 59 November-2016 496 450 354 618 369 475 258 606  
## 60 December-2016 521 461 375 645 358 514 267 592  
## 61 January-2017 501 434 364 634 358 512 265 616  
## 62 February-2017 517 469 366 656 363 515 277 623  
## 63 March-2017 500 457 373 635 363 519 259 611  
## 64 April-2017 463 404 334 590 304 449 248 521  
## 65 May-2017 502 442 353 621 324 474 250 565  
## 66 June-2017 493 433 352 631 331 478 269 598  
## 67 July-2017 495 438 352 626 357 488 269 585  
## 68 August-2017 524 464 376 654 363 510 280 619  
## 69 September-2017 503 454 351 632 362 485 256 612  
## 70 October-2017 532 462 361 656 382 505 291 640  
## 71 November-2017 517 428 368 628 347 506 257 597  
## 72 December-2017 450 418 330 579 328 445 245 546  
## 73 January-2018 499 457 366 625 358 486 256 611  
## 74 February-2018 509 455 368 628 340 500 277 592  
## 75 March-2018 527 448 358 647 368 507 278 594  
## 76 April-2018 524 460 374 649 374 509 275 637  
## 77 May-2018 533 439 380 642 366 494 268 621  
## 78 June-2018 524 454 362 665 351 512 271 609  
## 79 July-2018 539 460 383 652 347 506 272 609  
## 80 August-2018 471 409 341 561 325 464 259 524  
## 81 September-2018 485 434 340 608 334 489 238 584  
## 82 October-2018 488 445 363 630 346 480 260 597  
## 83 November-2018 515 444 363 639 368 511 269 608  
## 84 December-2018 547 449 372 654 352 530 277 608  
## 85 January-2019 511 459 372 643 373 529 274 604  
## 86 February-2019 518 455 360 657 350 508 269 624  
## 87 March-2019 507 446 363 631 359 495 265 594  
## 88 April-2019 453 394 334 563 319 455 245 538  
## 89 May-2019 510 443 362 639 332 490 258 598  
## 90 June-2019 504 418 345 608 354 499 268 586  
## 91 July-2019 510 477 361 647 372 499 289 612  
## 92 August-2019 506 443 368 627 349 498 263 593  
## 93 September-2019 533 459 362 650 369 531 269 621  
## 94 October-2019 522 440 375 642 372 521 269 616  
## 95 November-2019 502 457 376 629 362 503 275 612  
## 96 December-2019 453 409 320 557 308 421 229 544  
## negative positive df.Close PrevClose Change Class  
## 1 1168 1003 17193.55 15454.92 11.25 1  
## 2 1200 1070 17752.68 17193.55 3.25 1  
## 3 1172 1069 17404.20 17752.68 -1.96 0  
## 4 1240 1099 17318.81 17404.20 -0.49 0  
## 5 1230 1111 16218.53 17318.81 -6.35 0  
## 6 1238 1107 17429.98 16218.53 7.47 1  
## 7 1269 1096 17236.18 17429.98 -1.11 0  
## 8 1108 945 17429.56 17236.18 1.12 1  
## 9 1167 1034 18762.74 17429.56 7.65 1  
## 10 1186 1030 18505.38 18762.74 -1.37 0  
## 11 1149 1036 19339.90 18505.38 4.51 1  
## 12 1185 1055 19426.71 19339.90 0.45 1  
## 13 1181 1073 19894.98 19426.71 2.41 1  
## 14 1244 1069 18861.54 19894.98 -5.19 0  
## 15 1207 1053 18835.77 18861.54 -0.14 0  
## 16 1052 888 19504.18 18835.77 3.55 1  
## 17 1182 1024 19760.30 19504.18 1.31 1  
## 18 1172 1049 19395.81 19760.30 -1.84 0  
## 19 1135 1047 19345.70 19395.81 -0.26 0  
## 20 1282 1106 18619.72 19345.70 -3.75 0  
## 21 1267 1118 19379.77 18619.72 4.08 1  
## 22 1235 1086 21164.52 19379.77 9.21 1  
## 23 1242 1079 20791.93 21164.52 -1.76 0  
## 24 1099 928 21170.68 20791.93 1.82 1  
## 25 1224 1015 20513.85 21170.68 -3.10 0  
## 26 1221 1052 21120.12 20513.85 2.96 1  
## 27 1116 990 22386.27 21120.12 5.99 1  
## 28 1220 1101 22417.80 22386.27 0.14 1  
## 29 1239 1110 24217.34 22417.80 8.03 1  
## 30 1301 1116 25413.78 24217.34 4.94 1  
## 31 1254 1071 25894.97 25413.78 1.89 1  
## 32 1096 912 26638.11 25894.97 2.87 1  
## 33 1183 1015 26630.51 26638.11 -0.03 0  
## 34 1211 1024 27865.83 26630.51 4.64 1  
## 35 1185 1037 28693.99 27865.83 2.97 1  
## 36 1241 1115 27499.42 28693.99 -4.16 0  
## 37 1267 1070 29182.95 27499.42 6.12 1  
## 38 1305 1082 29361.50 29182.95 0.61 1  
## 39 1257 1097 27957.49 29361.50 -4.78 0  
## 40 1139 925 27011.31 27957.49 -3.38 0  
## 41 1179 1013 27828.44 27011.31 3.03 1  
## 42 1197 1027 27780.83 27828.44 -0.17 0  
## 43 1215 1042 28114.56 27780.83 1.20 1  
## 44 1255 1067 26283.09 28114.56 -6.51 0  
## 45 1225 1064 26154.83 26283.09 -0.49 0  
## 46 1213 1073 26656.83 26154.83 1.92 1  
## 47 1294 1057 26145.67 26656.83 -1.92 0  
## 48 1080 910 26117.54 26145.67 -0.11 0  
## 49 1224 1043 24870.69 26117.54 -4.77 0  
## 50 1225 1016 23002.00 24870.69 -7.51 0  
## 51 1219 1067 25341.86 23002.00 10.17 1  
## 52 1263 1099 25606.62 25341.86 1.04 1  
## 53 1260 1077 26667.96 25606.62 4.14 1  
## 54 1276 1116 26999.72 26667.96 1.24 1  
## 55 1312 1069 28051.86 26999.72 3.90 1  
## 56 1112 926 28452.17 28051.86 1.43 1  
## 57 1245 1044 27865.96 28452.17 -2.06 0  
## 58 1218 1064 27930.21 27865.96 0.23 1  
## 59 1226 1081 26652.81 27930.21 -4.57 0  
## 60 1309 1082 26626.46 26652.81 -0.10 0  
## 61 1239 1078 27655.96 26626.46 3.87 1  
## 62 1260 1120 28743.32 27655.96 3.93 1  
## 63 1255 1080 29620.50 28743.32 3.05 1  
## 64 1095 945 29918.40 29620.50 1.01 1  
## 65 1190 1004 31145.80 29918.40 4.10 1  
## 66 1191 1050 30921.61 31145.80 -0.72 0  
## 67 1205 1067 32514.94 30921.61 5.15 1  
## 68 1264 1110 31730.49 32514.94 -2.41 0  
## 69 1225 1076 31283.72 31730.49 -1.41 0  
## 70 1252 1125 33213.13 31283.72 6.17 1  
## 71 1271 1065 33149.35 33213.13 -0.19 0  
## 72 1106 968 34056.83 33149.35 2.74 1  
## 73 1199 1090 35965.02 34056.83 5.60 1  
## 74 1233 1066 34184.04 35965.02 -4.95 0  
## 75 1212 1085 32968.68 34184.04 -3.56 0  
## 76 1243 1126 35160.36 32968.68 6.65 1  
## 77 1256 1088 35322.38 35160.36 0.46 1  
## 78 1258 1082 35423.48 35322.38 0.29 1  
## 79 1264 1095 37606.58 35423.48 6.16 1  
## 80 1116 946 38645.07 37606.58 2.76 1  
## 81 1198 1043 36227.14 38645.07 -6.26 0  
## 82 1200 1052 34442.05 36227.14 -4.93 0  
## 83 1260 1117 36194.30 34442.05 5.09 1  
## 84 1268 1092 36068.33 36194.30 -0.35 0  
## 85 1249 1065 36256.69 36068.33 0.52 1  
## 86 1270 1089 35867.44 36256.69 -1.07 0  
## 87 1226 1082 38672.91 35867.44 7.82 1  
## 88 1100 941 39031.55 38672.91 0.93 1  
## 89 1235 1071 39714.20 39031.55 1.75 1  
## 90 1206 1046 39394.64 39714.20 -0.80 0  
## 91 1244 1107 37481.12 39394.64 -4.86 0  
## 92 1254 1070 37332.79 37481.12 -0.40 0  
## 93 1255 1122 38667.33 37332.79 3.57 1  
## 94 1274 1113 40129.05 38667.33 3.78 1  
## 95 1258 1084 40793.81 40129.05 1.66 1  
## 96 1063 929 41253.74 40793.81 1.13 1

final2 <- final[,c(-1,-12:-14)]  
final2[,1:10] <- scale(final2[,1:10])  
final2$Class <- factor(final2$Class)  
class(final2$Class)

## [1] "factor"

head(final2)

## anger anticipation disgust fear joy sadness  
## 1 -1.0313159 -0.05373090 -1.27113299 -0.65975196 -0.70551653 -0.299767546  
## 2 -0.1818076 0.03676325 -0.14018983 -1.03177069 -0.01237748 -1.099147668  
## 3 0.5132446 -0.05373090 -0.93185004 -0.03972075 0.18566224 -0.452030426  
## 4 0.6677006 1.03219894 0.02945164 0.42530266 1.22537081 0.080889655  
## 5 0.2043325 0.89645771 -0.42292562 0.61131202 0.48272184 -0.261701826  
## 6 0.2815605 0.80596356 0.14254596 0.11528705 0.87880129 0.004758215  
## surprise trust negative positive Class  
## 1 -0.5202548 -0.98731806 -0.7863307 -0.8998395 1  
## 2 -0.7274932 -0.16072620 -0.2350737 0.2947462 1  
## 3 0.5159373 -0.02296089 -0.7174235 0.2769166 0  
## 4 0.9994936 0.80363098 0.4539974 0.8118057 0  
## 5 1.2067320 0.45921770 0.2817296 1.0257613 0  
## 6 0.2396194 1.38913355 0.4195438 0.9544428 1

split <- sample.split(final2$Class, SplitRatio = 0.75)  
training <- subset(final2, split == T)  
test <- subset(final2, split == F)  
colnames(final2)

## [1] "anger" "anticipation" "disgust" "fear" "joy"   
## [6] "sadness" "surprise" "trust" "negative" "positive"   
## [11] "Class"

lr\_classifier <- glm(formula = Class ~ . , family = binomial, data = training )  
summary(lr\_classifier)

##   
## Call:  
## glm(formula = Class ~ ., family = binomial, data = training)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.1476 -1.1260 0.6614 0.9341 1.7611   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.40622 0.26429 1.537 0.1243   
## anger 0.05909 0.84575 0.070 0.9443   
## anticipation 0.68767 0.66416 1.035 0.3005   
## disgust 1.27827 0.69302 1.845 0.0651 .  
## fear -0.84708 0.97745 -0.867 0.3861   
## joy -0.24142 0.69605 -0.347 0.7287   
## sadness 0.82270 0.68289 1.205 0.2283   
## surprise -0.13474 0.61786 -0.218 0.8274   
## trust 0.19420 0.97762 0.199 0.8425   
## negative -2.47908 0.99674 -2.487 0.0129 \*  
## positive 0.43003 1.16818 0.368 0.7128   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 97.074 on 71 degrees of freedom  
## Residual deviance: 85.971 on 61 degrees of freedom  
## AIC: 107.97  
##   
## Number of Fisher Scoring iterations: 4

prob <- predict(lr\_classifier, type = 'response', newdata = test[-11])  
prob

## 3 12 13 19 26 31 32 33   
## 0.6567945 0.7305102 0.9044787 0.7145412 0.5755998 0.7891898 0.7710551 0.6827620   
## 38 42 47 51 53 57 58 61   
## 0.3305961 0.3047520 0.3649474 0.5965391 0.4087734 0.5198055 0.4552449 0.5310982   
## 67 70 74 77 82 86 89 91   
## 0.4903700 0.3750837 0.7259709 0.4991096 0.7304443 0.2092745 0.4583411 0.5272249

y <- ifelse(prob > 0.5, 1, 0)  
y

## 3 12 13 19 26 31 32 33 38 42 47 51 53 57 58 61 67 70 74 77 82 86 89 91   
## 1 1 1 1 1 1 1 1 0 0 0 1 0 1 0 1 0 0 1 0 1 0 0 1

lr\_cm <- table(test$Class, y)  
lr\_cm

## y  
## 0 1  
## 0 3 7  
## 1 7 7

dtclassifier <- rpart(formula = Class ~ . , data = training)  
  
dt\_y <- predict(dtclassifier, newdata = test[-11], type = 'class')  
dt\_y

## 3 12 13 19 26 31 32 33 38 42 47 51 53 57 58 61 67 70 74 77 82 86 89 91   
## 0 1 1 1 1 1 1 0 0 0 0 0 1 0 0 1 0 1 1 1 1 1 1 1   
## Levels: 0 1

cm <- table(test$Class, dt\_y)  
cm

## dt\_y  
## 0 1  
## 0 5 5  
## 1 4 10

rf\_classifier <- randomForest(x=training[-11], y=training$Class, ntree = 300)  
  
rf\_y <- predict(rf\_classifier, newdata = test[-11])  
rf\_y

## 3 12 13 19 26 31 32 33 38 42 47 51 53 57 58 61 67 70 74 77 82 86 89 91   
## 0 1 1 1 0 0 1 1 1 1 0 0 0 0 0 1 0 1 0 1 0 1 0 0   
## Levels: 0 1

cm <- table(test$Class, rf\_y)  
cm

## rf\_y  
## 0 1  
## 0 6 4  
## 1 7 7