> print(sprintf("model: %s formula: %s", model.name, deparse(model.formula)))

[1] "model: LDA - Feature Selection formula: factor(appetency) ~ ."

>

> #########################################

[1] "model: LDA - Feature Selection formula: factor(appetency) ~ ."

>

> #########################################

> ############## Drive Config #############

> ############## Drive Config #############

> #########################################

> #########################################

> if(isServerRun){

> if(isServerRun){

+ setwd('/host/dsm1/fmare001/stats/svm/deliverables')

+ setwd('/host/dsm1/fmare001/stats/svm/deliverables')

+ }else{

+ }else{

+ #setwd('C:/Users/audrey.ekuban/dev/goldsmiths/mlsdm/assignment3')

+ #setwd('C:/Users/audrey.ekuban/dev/goldsmiths/mlsdm/assignment3')

+ #setwd('C:/Users/john/dev/goldsmiths/mlsdm/assignment3')

+ #setwd('C:/Users/john/dev/goldsmiths/mlsdm/assignment3')

+ setwd('C:/Users/Fred/Desktop/Studies/MSc-DataScience/Statistical Learning/Assignments/Assignment3/deliverables')

+ setwd('C:/Users/Fred/Desktop/Studies/MSc-DataScience/Statistical Learning/Assignments/Assignment3/deliverables')

+ }

+ }

>

> #########################################

>

> #########################################

> ########### Load Dependencies ###########

> ########### Load Dependencies ###########

> #########################################

> #########################################

> source("init\_data.r")

> source("init\_data.r")

> source("exploratory\_functions.r")

> source("exploratory\_functions.r")

> source("pre\_processing\_functions.r")

> source("pre\_processing\_functions.r")

> source("feat\_selection.r")

> source("feat\_selection.r")

> require(randomForest)

> require(randomForest)

>

> forceReloadPreCanned1 = TRUE

>

> forceReloadPreCanned1 = TRUE

> if (forceReloadPreCanned1) {

> if (forceReloadPreCanned1) {

+ #List the pre-processing functions

+ #List the pre-processing functions

+ model.preProcessingFunctions <- c(

+ model.preProcessingFunctions <- c(

+ convert\_to\_factors,

+ convert\_to\_factors,

+ drop\_na\_cols,

+ drop\_na\_cols,

+ remove\_correlated\_predictors,

+ remove\_correlated\_predictors,

+ convert\_NAs\_to\_level,

+ convert\_NAs\_to\_level,

+ remove\_linear\_dependencies,

+ remove\_linear\_dependencies,

+ bin\_negative\_levels\_appetency,

+ bin\_negative\_levels\_appetency,

+ keep\_top\_10\_levels,

+ keep\_top\_10\_levels,

+ impute\_data

+ impute\_data

+ )

+ )

+ # Need to do these up-front, otherwise we might end up with mis-matched

+ # Need to do these up-front, otherwise we might end up with mis-matched

+ # levels between training and test folds in the cross-validation loop.

+ # levels between training and test folds in the cross-validation loop.

+ # Shouldn't introduce any bias since nothing is being imputed.

+ # Shouldn't introduce any bias since nothing is being imputed.

+ model.data <- apply\_pre\_processing(train, model.preProcessingFunctions)

+ model.data <- apply\_pre\_processing(train, model.preProcessingFunctions)

+ write.csv(model.data, file = "train.bin.neg.top10.app.csv", row.names = FALSE)

+ write.csv(model.data, file = "train.bin.neg.top10.app.csv", row.names = FALSE)

+ } else {

+ } else {

+ #reload from file

+ #reload from file

+ model.data <- read.csv("train.bin.neg.top10.app.csv", stringsAsFactors = FALSE)

+ model.data <- read.csv("train.bin.neg.top10.app.csv", stringsAsFactors = FALSE)

+ model.data <- convert\_to\_factors(model.data)

+ model.data <- convert\_to\_factors(model.data)

+ }

+ }

[1] "Number of removed linearly dependent col(s): 0"

[1] "Number of removed linearly dependent col(s): 0"

>

>

> random\_forest\_feature\_selection(model.formula, model.data,500)

>

>

> random\_forest\_feature\_selection(model.formula, model.data,500)

-1 1 MeanDecreaseAccuracy MeanDecreaseGini

V84 13.66575199 45.55443006 35.0086726 91.8769876

V118 -7.59461885 48.40969724 11.4862690 68.2614555

V165 9.37564269 49.63197733 37.3648416 57.9682287

V224 -21.99289706 38.82361446 0.6118401 55.2046874

V116 10.25874751 23.63932855 17.3139752 35.2540849

V155 12.29133518 -4.91118188 9.7523000 30.5370201

V14 5.82544566 1.71527779 6.4065350 25.8992885

V90 12.09120673 15.54896682 16.8851757 25.7503586

V156 -0.13194249 5.04943116 2.4518366 25.6281656

V101 5.84063066 0.83177942 6.2850891 25.2844451

V93 7.31838422 13.93813512 9.9064478 24.6572119

V187 7.01657258 13.61513297 8.9089478 24.2806829

V111 0.62249219 -0.60498506 0.3165445 23.7582023

V70 3.45208750 -1.46378623 2.2833329 23.3319818

V222 9.06112800 1.48605789 9.5176451 23.0957738

V192 7.74995105 0.25286644 8.3326742 22.9640100

V141 9.35149247 -5.46028823 7.7162666 21.1381328

V134 10.82411150 -1.98091440 10.2908976 20.8474401

V161 13.01423798 -4.14716864 11.7394092 19.8010092

V55 12.07758167 -4.14556338 11.5133642 19.7275242

V189 12.75924840 -3.14058829 11.8108122 19.1310087

V191 7.57922656 -1.67618491 7.0711103 18.6688470

V166 13.15664055 -3.65067853 12.7164509 17.5424871

V37 6.96502905 -1.22482046 6.7127792 17.5090359

V42 11.95953217 -3.12117118 11.8594412 17.4508936

V91 6.33324556 -1.01673774 6.0938049 16.9497247

V124 11.58333771 -3.79463215 10.6612057 16.8044589

V151 14.23009009 -4.49862002 13.7773350 16.7404863

V119 13.12550392 -6.18747830 10.8942176 16.3586021

V202 12.73900232 -5.18216271 11.5903073 15.7114873

V96 8.88859049 -2.57529366 8.4833733 15.2980564

V81 5.65106681 -1.51556460 5.2639427 14.8143670

V1 6.87055038 -0.05890440 6.7871470 14.5733733

V75 9.15713468 -5.03808928 7.7715382 14.1830018

V197 11.51446835 -4.24961835 10.5263931 13.9428348

V121 6.95953588 -2.61094308 5.7737932 13.8027887

V206 6.33583874 -4.76891848 5.7329623 13.6615697

V67 5.23796232 -0.72728212 5.1500969 13.1566410

V71 9.98220373 -1.10753936 9.3420749 12.3176280

V150 8.69048760 -2.35412063 7.9543841 11.3888629

V15 9.61373448 -2.01318672 8.6422665 10.5166053

V149 17.15497950 -11.16953372 16.1327903 10.4823560

V21 7.45477721 -0.57655500 7.4007579 10.3217752

V29 10.87243503 -7.96961519 10.2631775 8.2678712

V58 3.40061526 -0.95134741 2.8918676 8.2217859

V190 13.75925112 -9.09683746 13.4111830 8.2194751

V204 11.46947050 -4.24687100 11.3308355 7.7816825

V95 4.02300796 -0.43925896 3.6010782 7.7479986

V112 12.30167093 -3.80970189 12.5129113 7.2374451

V114 3.63077345 -1.06504024 2.8269005 7.0644965

V115 5.52374251 -1.35058303 5.3570041 6.9375723

V107 4.40143599 -1.28739549 3.7421412 6.8509529

V138 -3.08821739 11.59901000 2.7040270 6.7242576

upselling 9.42351780 14.75006408 13.3065649 6.6225902

V212 11.34403002 -5.37549927 11.1382063 6.4179283

V143 2.19902700 -0.43548252 1.8048761 6.3131821

V79 1.29700665 -1.05675036 0.7907959 5.8522469

V171 5.19601934 3.23756293 6.1414578 5.6450387

V154 5.94565393 -3.42096711 5.4536370 5.3708211

V30 0.68901089 0.04880968 0.6558780 5.0201906

churn 9.20138837 15.38078003 14.7323173 4.7782742

V168 6.24773686 -1.42014727 6.2726251 4.2873608

V196 -0.06277142 -1.15889830 -0.5418918 2.5524178

V230 3.93895197 -0.19972217 3.9755644 2.4224946

V11 -1.85627667 0.46845329 -1.6952522 1.9166825

V94 3.58905363 -2.56350965 3.1973507 1.7673644

V125 -0.22369505 -0.73416602 -0.5602741 1.0861307

V219 2.49401833 -1.32826446 2.3207894 0.8733158

V113 1.66352978 -1.05607629 1.6249390 0.6619434

V128 -3.11792067 -0.94314047 -3.3408927 0.6269780

V73 -1.22933250 -0.86372762 -1.3437638 0.4084785

-1 1 MeanDecreaseAccuracy MeanDecreaseGini

V84 13.66575199 45.55443006 35.0086726 91.8769876

V118 -7.59461885 48.40969724 11.4862690 68.2614555

V165 9.37564269 49.63197733 37.3648416 57.9682287

V224 -21.99289706 38.82361446 0.6118401 55.2046874

V116 10.25874751 23.63932855 17.3139752 35.2540849

V155 12.29133518 -4.91118188 9.7523000 30.5370201

V14 5.82544566 1.71527779 6.4065350 25.8992885

V90 12.09120673 15.54896682 16.8851757 25.7503586

V156 -0.13194249 5.04943116 2.4518366 25.6281656

V101 5.84063066 0.83177942 6.2850891 25.2844451

V93 7.31838422 13.93813512 9.9064478 24.6572119

V187 7.01657258 13.61513297 8.9089478 24.2806829

V111 0.62249219 -0.60498506 0.3165445 23.7582023

V70 3.45208750 -1.46378623 2.2833329 23.3319818

V222 9.06112800 1.48605789 9.5176451 23.0957738

V192 7.74995105 0.25286644 8.3326742 22.9640100

V141 9.35149247 -5.46028823 7.7162666 21.1381328

V134 10.82411150 -1.98091440 10.2908976 20.8474401

V161 13.01423798 -4.14716864 11.7394092 19.8010092

V55 12.07758167 -4.14556338 11.5133642 19.7275242

V189 12.75924840 -3.14058829 11.8108122 19.1310087

V191 7.57922656 -1.67618491 7.0711103 18.6688470

V166 13.15664055 -3.65067853 12.7164509 17.5424871

V37 6.96502905 -1.22482046 6.7127792 17.5090359

V42 11.95953217 -3.12117118 11.8594412 17.4508936

V91 6.33324556 -1.01673774 6.0938049 16.9497247

V124 11.58333771 -3.79463215 10.6612057 16.8044589

V151 14.23009009 -4.49862002 13.7773350 16.7404863

V119 13.12550392 -6.18747830 10.8942176 16.3586021

V202 12.73900232 -5.18216271 11.5903073 15.7114873

V96 8.88859049 -2.57529366 8.4833733 15.2980564

V81 5.65106681 -1.51556460 5.2639427 14.8143670

V1 6.87055038 -0.05890440 6.7871470 14.5733733

V75 9.15713468 -5.03808928 7.7715382 14.1830018

V197 11.51446835 -4.24961835 10.5263931 13.9428348

V121 6.95953588 -2.61094308 5.7737932 13.8027887

V206 6.33583874 -4.76891848 5.7329623 13.6615697

V67 5.23796232 -0.72728212 5.1500969 13.1566410

V71 9.98220373 -1.10753936 9.3420749 12.3176280

V150 8.69048760 -2.35412063 7.9543841 11.3888629

V15 9.61373448 -2.01318672 8.6422665 10.5166053

V149 17.15497950 -11.16953372 16.1327903 10.4823560

V21 7.45477721 -0.57655500 7.4007579 10.3217752

V29 10.87243503 -7.96961519 10.2631775 8.2678712

V58 3.40061526 -0.95134741 2.8918676 8.2217859

V190 13.75925112 -9.09683746 13.4111830 8.2194751

V204 11.46947050 -4.24687100 11.3308355 7.7816825

V95 4.02300796 -0.43925896 3.6010782 7.7479986

V112 12.30167093 -3.80970189 12.5129113 7.2374451

V114 3.63077345 -1.06504024 2.8269005 7.0644965

V115 5.52374251 -1.35058303 5.3570041 6.9375723

V107 4.40143599 -1.28739549 3.7421412 6.8509529

V138 -3.08821739 11.59901000 2.7040270 6.7242576

upselling 9.42351780 14.75006408 13.3065649 6.6225902

V212 11.34403002 -5.37549927 11.1382063 6.4179283

V143 2.19902700 -0.43548252 1.8048761 6.3131821

V79 1.29700665 -1.05675036 0.7907959 5.8522469

V171 5.19601934 3.23756293 6.1414578 5.6450387

V154 5.94565393 -3.42096711 5.4536370 5.3708211

V30 0.68901089 0.04880968 0.6558780 5.0201906

churn 9.20138837 15.38078003 14.7323173 4.7782742

V168 6.24773686 -1.42014727 6.2726251 4.2873608

V196 -0.06277142 -1.15889830 -0.5418918 2.5524178

V230 3.93895197 -0.19972217 3.9755644 2.4224946

V11 -1.85627667 0.46845329 -1.6952522 1.9166825

V94 3.58905363 -2.56350965 3.1973507 1.7673644

V125 -0.22369505 -0.73416602 -0.5602741 1.0861307

V219 2.49401833 -1.32826446 2.3207894 0.8733158

V113 1.66352978 -1.05607629 1.6249390 0.6619434

V128 -3.11792067 -0.94314047 -3.3408927 0.6269780

V73 -1.22933250 -0.86372762 -1.3437638 0.4084785

