MLHW6\_JF

February 23, 2022

## Q1: Restrict the NHANES data to the list of 11 variables below. Partition the data into training and testing using a 70/30 split. REMINDER: Look at the frequency of your outcome variable to check for balance

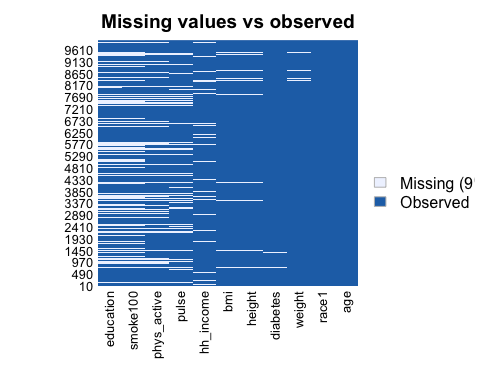
data(NHANES)  
  
NH = NHANES %>%  
 janitor::clean\_names() %>%  
 select(age, race1, education, hh\_income, weight, height, pulse, diabetes, bmi, phys\_active, smoke100)  
  
NH$diabetes <- factor(NH$diabetes, levels = c("No", "Yes"))  
  
#Check distributions, missing data etc, omitting the NAs  
summary(NH)

## age race1 education hh\_income   
## Min. : 0.00 Black :1197 8th Grade : 451 more 99999 :2220   
## 1st Qu.:17.00 Hispanic: 610 9 - 11th Grade: 888 75000-99999:1084   
## Median :36.00 Mexican :1015 High School :1517 25000-34999: 958   
## Mean :36.74 White :6372 Some College :2267 35000-44999: 863   
## 3rd Qu.:54.00 Other : 806 College Grad :2098 45000-54999: 784   
## Max. :80.00 NA's :2779 (Other) :3280   
## NA's : 811   
## weight height pulse diabetes bmi   
## Min. : 2.80 Min. : 83.6 Min. : 40.00 No :9098 Min. :12.88   
## 1st Qu.: 56.10 1st Qu.:156.8 1st Qu.: 64.00 Yes : 760 1st Qu.:21.58   
## Median : 72.70 Median :166.0 Median : 72.00 NA's: 142 Median :25.98   
## Mean : 70.98 Mean :161.9 Mean : 73.56 Mean :26.66   
## 3rd Qu.: 88.90 3rd Qu.:174.5 3rd Qu.: 82.00 3rd Qu.:30.89   
## Max. :230.70 Max. :200.4 Max. :136.00 Max. :81.25   
## NA's :78 NA's :353 NA's :1437 NA's :366   
## phys\_active smoke100   
## No :3677 No :4024   
## Yes :4649 Yes :3211   
## NA's:1674 NA's:2765   
##   
##   
##   
##

missmap(NH, main = "Missing values vs observed")

## Warning: Unknown or uninitialised column: `arguments`.  
  
## Warning: Unknown or uninitialised column: `arguments`.

## Warning: Unknown or uninitialised column: `imputations`.



NH <- na.omit(NH)  
summary(NH$diabetes) #Notice that the data is unbalanced so we will downsize

## No Yes   
## 5697 659

#tidyverse way to create data partition  
train\_indices <- createDataPartition(y = NH$diabetes,p = 0.7,list = FALSE)  
train\_data <- NH[train\_indices, ]  
test\_data <- NH[-train\_indices, ]

## Q2:Construct three prediction models to predict diabetes using the 11 features from NHANES. You will use the following three algorithms to create your prediction models: Q3: You will optimize each model using cross-validation to choose hyperparameters in the training data and then compare performance across models.

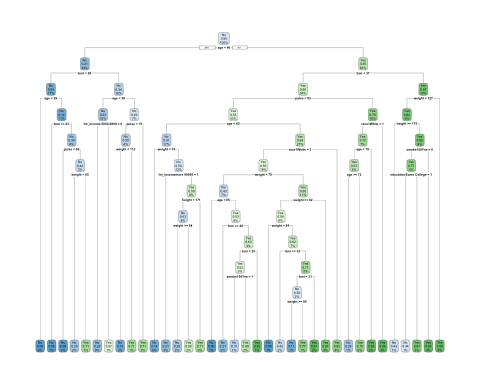
#The diabetes data is unbalanced so we will downsize.  
  
set.seed(100)  
#Creating 10-fold cross-validation and using down-sampling because of imbalance in data  
train\_control\_ct <- trainControl(method = "cv", number = 10, sampling = "down")  
  
#Create sequence of cp parameters to try   
grid\_ct <- expand.grid(cp = seq(0.001, 0.3, by = 0.01))  
  
#Train model  
ct\_diabetes <- train(diabetes~., data = train\_data, method = "rpart",trControl = train\_control\_ct, tuneGrid = grid\_ct)  
  
ct\_diabetes$bestTune #cp:0.001

## cp  
## 1 0.001

ct\_diabetes

## CART   
##   
## 4450 samples  
## 10 predictor  
## 2 classes: 'No', 'Yes'   
##   
## No pre-processing  
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 4004, 4005, 4005, 4005, 4006, 4006, ...   
## Addtional sampling using down-sampling  
##   
## Resampling results across tuning parameters:  
##   
## cp Accuracy Kappa   
## 0.001 0.7116863 0.2090109  
## 0.011 0.6723250 0.2099558  
## 0.021 0.6660193 0.1996902  
## 0.031 0.6350539 0.1845990  
## 0.041 0.6242987 0.1651977  
## 0.051 0.6112650 0.1579774  
## 0.061 0.6112650 0.1579774  
## 0.071 0.6112650 0.1579774  
## 0.081 0.6112650 0.1579774  
## 0.091 0.6112650 0.1579774  
## 0.101 0.6112650 0.1579774  
## 0.111 0.6112650 0.1579774  
## 0.121 0.6112650 0.1579774  
## 0.131 0.6112650 0.1579774  
## 0.141 0.6112650 0.1579774  
## 0.151 0.6112650 0.1579774  
## 0.161 0.6112650 0.1579774  
## 0.171 0.6112650 0.1579774  
## 0.181 0.6112650 0.1579774  
## 0.191 0.6112650 0.1579774  
## 0.201 0.6112650 0.1579774  
## 0.211 0.6112650 0.1579774  
## 0.221 0.6112650 0.1579774  
## 0.231 0.6112650 0.1579774  
## 0.241 0.6112650 0.1579774  
## 0.251 0.6112650 0.1579774  
## 0.261 0.6112650 0.1579774  
## 0.271 0.6112650 0.1579774  
## 0.281 0.6112650 0.1579774  
## 0.291 0.6112650 0.1579774  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final value used for the model was cp = 0.001.

rpart.plot(ct\_diabetes$finalModel)



#variable importance on the final model within training data  
varImp(ct\_diabetes) # top three most important variables were age(100), bmi(81.99), weight(65.090) and the least important variable was income.

## rpart variable importance  
##   
## only 20 most important variables shown (out of 38)  
##   
## Overall  
## age 100.000  
## bmi 85.183  
## weight 74.236  
## pulse 39.963  
## hh\_incomemore 99999 33.585  
## height 27.701  
## race1White 24.231  
## phys\_activeYes 18.996  
## hh\_income55000-64999 7.876  
## smoke100Yes 6.071  
## educationCollege Grad 5.160  
## educationSome College 4.224  
## hh\_income45000-54999 3.322  
## hh\_income10000-14999 2.697  
## race1Mexican 2.575  
## educationHigh School 2.031  
## hh\_income35000-44999 1.706  
## hh\_income75000-99999 1.601  
## hh\_income25000-34999 1.414  
## hh\_income 5000-9999 1.339

#accuracy metric and confusion matrix from training.  
confusionMatrix(ct\_diabetes) #Accuracy(average): 0.7171

## Cross-Validated (10 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction No Yes  
## No 64.0 3.2  
## Yes 25.6 7.2  
##   
## Accuracy (average) : 0.7117

#if this the measure you choose then I can predictions on the test data as probabilities and/or produce a ROC curve.

modelLookup("svmLinear")

## model parameter label forReg forClass probModel  
## 1 svmLinear C Cost TRUE TRUE TRUE

set.seed(100)  
  
#Set 10-fold cross-validation. Note if you want predicted probabilities, you need to set class Probs=True  
train\_controlSVC <- trainControl(method = "cv", number = 10, classProbs = T)  
  
svm <- train(diabetes ~ ., data = train\_data, method = "svmLinear", trControl = train\_controlSVC, preProcess = c("center", "scale"))

## maximum number of iterations reached 0.009904756 0.009094279maximum number of iterations reached 0.008646789 0.007975517maximum number of iterations reached 0.006107872 0.00576851maximum number of iterations reached 0.0115749 0.01067183maximum number of iterations reached 0.007301763 0.006817109maximum number of iterations reached 0.007557303 0.007044807maximum number of iterations reached 0.008235431 0.00758133maximum number of iterations reached 0.009440526 0.008638458maximum number of iterations reached 0.00584573 0.005574221maximum number of iterations reached 0.005487902 0.005176938maximum number of iterations reached 0.006812653 0.006431533

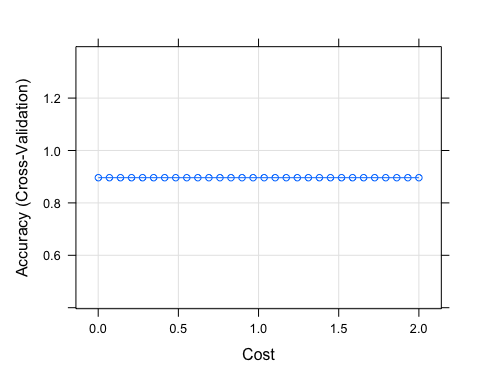
svm #accuracy: 0.8961

## Support Vector Machines with Linear Kernel   
##   
## 4450 samples  
## 10 predictor  
## 2 classes: 'No', 'Yes'   
##   
## Pre-processing: centered (26), scaled (26)   
## Resampling: Cross-Validated (10 fold)   
## Summary of sample sizes: 4004, 4005, 4005, 4005, 4006, 4006, ...   
## Resampling results:  
##   
## Accuracy Kappa  
## 0.8961806 0   
##   
## Tuning parameter 'C' was held constant at a value of 1

#Incorporate different values for cost paramter(cp) bc this method won't tune the hyperparameters for us. The CP is how much misclassification the support vector will allow  
svm\_caret <- train(diabetes ~ ., data = train\_data, method = "svmLinear", trControl = train\_controlSVC, preProcess = c("center", "scale"), tuneGrid = expand.grid(C = seq(0.001,2, length = 30)))

## maximum number of iterations reached 0.00953775 0.008576621maximum number of iterations reached 0.01085459 0.009768192maximum number of iterations reached 0.00387818 0.003775983maximum number of iterations reached 0.006843363 0.006334838maximum number of iterations reached 0.004042363 0.003900125maximum number of iterations reached 0.006927224 0.006415399maximum number of iterations reached 0.008871119 0.008232275maximum number of iterations reached 0.004885015 0.004636722maximum number of iterations reached 0.00807131 0.007512031maximum number of iterations reached 0.0096223 0.00882762maximum number of iterations reached 0.005962699 0.005603407maximum number of iterations reached 0.002554365 0.002499112maximum number of iterations reached 0.007053665 0.006703063maximum number of iterations reached 0.005191927 0.004990852maximum number of iterations reached 0.005763083 0.005486019maximum number of iterations reached 0.005936902 0.00562298maximum number of iterations reached 0.006278137 0.005946422maximum number of iterations reached 0.01045621 0.009465822maximum number of iterations reached 0.007600048 0.007085706maximum number of iterations reached 0.007768462 0.007301714maximum number of iterations reached 0.007724149 0.007148346maximum number of iterations reached 0.003049413 0.00297504maximum number of iterations reached 0.01095286 0.009742282maximum number of iterations reached 0.005000481 0.004789618maximum number of iterations reached 0.004734818 0.004587354maximum number of iterations reached 0.006025287 0.005693541maximum number of iterations reached 0.005770304 0.005525551maximum number of iterations reached 0.007397995 0.006885786maximum number of iterations reached 0.008843072 0.00815328maximum number of iterations reached 0.01006341 0.009133714maximum number of iterations reached 0.01141534 0.009701783maximum number of iterations reached 0.008918197 0.008167353maximum number of iterations reached 0.007607776 0.007094438maximum number of iterations reached 0.009289986 0.008353115maximum number of iterations reached 0.00629808 0.005951278maximum number of iterations reached 0.002476424 0.002422565maximum number of iterations reached 0.006572201 0.00613995maximum number of iterations reached 0.006525931 0.006156031maximum number of iterations reached 0.008206533 0.007546654maximum number of iterations reached 0.006494004 0.006047266maximum number of iterations reached 0.006808999 0.006415821maximum number of iterations reached 0.00613104 0.0057354maximum number of iterations reached 0.006626872 0.006217924maximum number of iterations reached 0.008842479 0.00827568maximum number of iterations reached 0.007410126 0.006807959maximum number of iterations reached 0.004492205 0.004326342maximum number of iterations reached 0.006380058 0.005986507maximum number of iterations reached 0.00778533 0.007295936maximum number of iterations reached 0.008619547 0.007956483maximum number of iterations reached 0.006871545 0.006528753maximum number of iterations reached 0.006526052 0.006247644maximum number of iterations reached 0.009490073 0.008581205maximum number of iterations reached 0.006044862 0.005738994maximum number of iterations reached 0.009387844 0.008695005maximum number of iterations reached 0.009503907 0.008628325maximum number of iterations reached 0.003855676 0.003766624maximum number of iterations reached 0.007041976 0.006611667maximum number of iterations reached 0.005009967 0.004786215maximum number of iterations reached 0.006905243 0.006467136maximum number of iterations reached 0.002749423 0.002694487maximum number of iterations reached 0.007616287 0.007178133maximum number of iterations reached 0.009355235 0.008416306maximum number of iterations reached 0.005572633 0.005259669maximum number of iterations reached 0.008512003 0.0077836maximum number of iterations reached 0.006348758 0.005947933maximum number of iterations reached 0.005806968 0.005523019maximum number of iterations reached 0.008896373 0.007985722maximum number of iterations reached 0.008906243 0.008104149maximum number of iterations reached 0.007850618 0.007291454maximum number of iterations reached 0.009461784 0.008672634maximum number of iterations reached 0.007646552 0.007135261maximum number of iterations reached 0.008200234 0.007672951maximum number of iterations reached 0.009187852 0.008518306maximum number of iterations reached 0.007963821 0.007454204maximum number of iterations reached 0.006624034 0.00624348maximum number of iterations reached 0.005797219 0.005518152maximum number of iterations reached 0.007245461 0.006866275maximum number of iterations reached 0.004970793 0.004775775maximum number of iterations reached 0.007908937 0.0074517maximum number of iterations reached 0.009686967 0.008987356maximum number of iterations reached 0.008993549 0.008285364maximum number of iterations reached 0.00680969 0.006375787maximum number of iterations reached 0.005291844 0.005074705maximum number of iterations reached 0.007780412 0.007304593maximum number of iterations reached 0.006142227 0.005825221maximum number of iterations reached 0.01066366 0.009646353maximum number of iterations reached 0.007169328 0.006693707maximum number of iterations reached 0.004906866 0.004688067maximum number of iterations reached 0.00665156 0.006315069maximum number of iterations reached 0.006144449 0.005823864maximum number of iterations reached 0.01041013 0.009275385maximum number of iterations reached 0.01068642 0.009861286maximum number of iterations reached 0.009536771 0.008523741maximum number of iterations reached 0.003453196 0.003358466maximum number of iterations reached 0.009091133 0.008406158maximum number of iterations reached 0.006545985 0.006179444maximum number of iterations reached 0.005496922 0.005229202maximum number of iterations reached 0.01059578 0.00966858maximum number of iterations reached 0.003908954 0.003770394maximum number of iterations reached 7.722006e-05 7.71809e-05maximum number of iterations reached 0.004487149 0.004323019maximum number of iterations reached 0.00305012 0.002981404maximum number of iterations reached 0.007263097 0.006841925maximum number of iterations reached 0.005969559 0.005664149maximum number of iterations reached 0.006547808 0.006155405maximum number of iterations reached 0.008186604 0.007587246maximum number of iterations reached 0.00987853 0.009105118maximum number of iterations reached 0.01016017 0.009195479maximum number of iterations reached 0.01035813 0.009536773maximum number of iterations reached 0.01017891 0.00936094maximum number of iterations reached 0.006520547 0.006206868maximum number of iterations reached 0.007994733 0.007453143maximum number of iterations reached 0.007695612 0.007238077maximum number of iterations reached 0.007660082 0.007232457maximum number of iterations reached 0.005548111 0.005284432maximum number of iterations reached 0.008088806 0.007519024maximum number of iterations reached 0.007041598 0.006693264maximum number of iterations reached 0.008906149 0.008183656maximum number of iterations reached 0.01107289 0.009909816maximum number of iterations reached 0.008203626 0.007669968maximum number of iterations reached 0.01189524 0.01004288maximum number of iterations reached 0.005316375 0.005055406maximum number of iterations reached 0.01019728 0.009347226maximum number of iterations reached 0.008293881 0.007719462maximum number of iterations reached 0.008655374 0.007893936maximum number of iterations reached 0.01055439 0.009612668maximum number of iterations reached 0.007402217 0.006953423maximum number of iterations reached 0.007601164 0.007041645maximum number of iterations reached 0.007849448 0.007315659maximum number of iterations reached 0.009086698 0.00836041maximum number of iterations reached 0.009209698 0.00829359maximum number of iterations reached 0.008487354 0.007829441maximum number of iterations reached 0.007866194 0.007285013maximum number of iterations reached 0.006755137 0.00630216maximum number of iterations reached 0.009557644 0.008726522maximum number of iterations reached 0.01005979 0.00926496maximum number of iterations reached 0.009547647 0.008696315maximum number of iterations reached 0.007951984 0.007510425maximum number of iterations reached 0.007556711 0.007040612maximum number of iterations reached 0.007133657 0.006647217maximum number of iterations reached 0.009723993 0.008915248maximum number of iterations reached 0.01212529 0.01064787maximum number of iterations reached 0.009501096 0.008752864maximum number of iterations reached 0.007621032 0.006978054maximum number of iterations reached 0.01117486 0.01009247maximum number of iterations reached 0.009281559 0.008540712maximum number of iterations reached 0.005813001 0.005550537maximum number of iterations reached 0.00918818 0.008488183maximum number of iterations reached 0.008046639 0.007441792maximum number of iterations reached 0.007956979 0.007387373maximum number of iterations reached 0.01302384 0.01090584maximum number of iterations reached 0.007146073 0.006659867maximum number of iterations reached 0.007636998 0.007052933maximum number of iterations reached 0.01061727 0.009780098maximum number of iterations reached 0.008331494 0.007712571maximum number of iterations reached 0.004533819 0.004356629maximum number of iterations reached 0.007016603 0.006652224maximum number of iterations reached 0.005820649 0.005571186maximum number of iterations reached 0.009672078 0.008843731maximum number of iterations reached 0.004167502 0.004037749maximum number of iterations reached 0.008613087 0.007870838maximum number of iterations reached 0.005394182 0.005145836maximum number of iterations reached 0.008419479 0.007741666maximum number of iterations reached 0.006008649 0.00567559maximum number of iterations reached 0.009403152 0.008510159maximum number of iterations reached 0.00505281 0.004882454maximum number of iterations reached 0.009360492 0.00848647maximum number of iterations reached 0.008439392 0.007902537maximum number of iterations reached 0.01042039 0.009528248maximum number of iterations reached 0.0072036 0.006729732maximum number of iterations reached 0.008283207 0.007723036maximum number of iterations reached 0.007932159 0.007254903maximum number of iterations reached 0.01160963 0.01058749maximum number of iterations reached 0.009424503 0.008546027maximum number of iterations reached 0.006837356 0.00653814maximum number of iterations reached 0.008990108 0.008316216maximum number of iterations reached 0.001681624 0.001660565maximum number of iterations reached 0.007971264 0.00752138maximum number of iterations reached 0.008204246 0.007553366maximum number of iterations reached 0.005099038 0.004867307maximum number of iterations reached 0.01179082 0.0104083maximum number of iterations reached 0.001961056 0.001922044maximum number of iterations reached 0.005654005 0.00532282maximum number of iterations reached 0.01050295 0.009525181maximum number of iterations reached 0.008031268 0.007542211maximum number of iterations reached 0.009849185 0.008870948maximum number of iterations reached 0.007522559 0.007073769maximum number of iterations reached 0.00782154 0.007312203maximum number of iterations reached 0.007784302 0.007306955maximum number of iterations reached 0.008467426 0.007810836maximum number of iterations reached 0.009147874 0.008452188maximum number of iterations reached 0.01014914 0.009326764maximum number of iterations reached 0.003785303 0.003662272maximum number of iterations reached 0.006698874 0.00630381maximum number of iterations reached 0.007428215 0.007005746maximum number of iterations reached 0.0101989 0.009294877maximum number of iterations reached 0.007644995 0.007149046maximum number of iterations reached 0.004764716 0.004599155maximum number of iterations reached 0.01029011 0.009565304maximum number of iterations reached 0.01081522 0.009715005maximum number of iterations reached 0.00827162 0.007694158maximum number of iterations reached 0.008643777 0.007889311maximum number of iterations reached 0.004343302 0.004186102maximum number of iterations reached 0.00691575 0.006527273maximum number of iterations reached 0.01188834 0.01062207maximum number of iterations reached 0.007418056 0.006954314maximum number of iterations reached 0.007727427 0.007185562maximum number of iterations reached 0.006738309 0.006332906maximum number of iterations reached 0.008112972 0.007533769maximum number of iterations reached 0.005905536 0.00563284maximum number of iterations reached 0.007261964 0.006609385maximum number of iterations reached 0.008167347 0.007474239maximum number of iterations reached 0.008225279 0.007475091maximum number of iterations reached 0.008609922 0.007907642maximum number of iterations reached 0.008208941 0.007558426maximum number of iterations reached 0.008223376 0.007702633maximum number of iterations reached 0.009448289 0.008734022maximum number of iterations reached 0.005656801 0.005337177maximum number of iterations reached 0.007377935 0.006948295maximum number of iterations reached 0.006937145 0.006436158maximum number of iterations reached 0.007857989 0.007322397maximum number of iterations reached 0.005516562 0.005281135maximum number of iterations reached 0.008727937 0.007995915maximum number of iterations reached 0.01179718 0.01050233maximum number of iterations reached 0.006634001 0.006322914maximum number of iterations reached 0.007179805 0.006755301maximum number of iterations reached 0.01202111 0.01055683maximum number of iterations reached 0.008627127 0.00795716maximum number of iterations reached 0.006709604 0.00626222maximum number of iterations reached 0.008213057 0.007670249maximum number of iterations reached 0.007447011 0.006938395maximum number of iterations reached 0.008282308 0.007735285maximum number of iterations reached 0.007214276 0.006774192maximum number of iterations reached 0.006674679 0.006281461maximum number of iterations reached 0.007508201 0.006998545maximum number of iterations reached 0.009924582 0.008977722maximum number of iterations reached 0.006321399 0.005967618maximum number of iterations reached 0.007318713 0.006958845maximum number of iterations reached 0.008609703 0.007983048maximum number of iterations reached 0.007984186 0.007427589maximum number of iterations reached 0.01077296 0.009845041maximum number of iterations reached 0.005149958 0.004879259maximum number of iterations reached 0.006267895 0.005931677maximum number of iterations reached 0.008720198 0.007992687maximum number of iterations reached 0.00891727 0.008206773maximum number of iterations reached 0.01004839 0.008898136maximum number of iterations reached 0.01137704 0.01028986maximum number of iterations reached 0.005304742 0.005052577maximum number of iterations reached 0.007742609 0.007228222maximum number of iterations reached 0.005453858 0.005205171maximum number of iterations reached 0.006462036 0.006134757maximum number of iterations reached 0.006043226 0.005721628maximum number of iterations reached 0.007203889 0.006719549maximum number of iterations reached 0.008621727 0.007931527maximum number of iterations reached 0.005222478 0.00502602maximum number of iterations reached 0.008070254 0.00750393maximum number of iterations reached 0.008681579 0.008073397maximum number of iterations reached 0.008176593 0.007583786maximum number of iterations reached 0.007858909 0.007317081maximum number of iterations reached 0.01238707 0.01098596maximum number of iterations reached 0.01014322 0.00927956maximum number of iterations reached 0.005706585 0.005417329maximum number of iterations reached 0.007324946 0.006918822maximum number of iterations reached 0.006049965 0.005785699maximum number of iterations reached 0.009834111 0.009166799maximum number of iterations reached 0.007251982 0.00689576maximum number of iterations reached 0.0078758 0.007258536maximum number of iterations reached 0.006748244 0.006377073maximum number of iterations reached 0.008264116 0.007675536maximum number of iterations reached 0.00490077 0.004682317maximum number of iterations reached 0.006318308 0.0060061maximum number of iterations reached 0.00799902 0.0073476maximum number of iterations reached 0.004840085 0.004642498maximum number of iterations reached 0.009027658 0.008417361maximum number of iterations reached 0.005558372 0.005306517maximum number of iterations reached 0.004869087 0.004669896maximum number of iterations reached 0.007419172 0.006980354maximum number of iterations reached 0.005746498 0.005422292maximum number of iterations reached 0.006126522 0.005665241maximum number of iterations reached 0.003805359 0.003671928maximum number of iterations reached 0.006868141 0.006426543maximum number of iterations reached 0.008099005 0.007539468maximum number of iterations reached 0.002439675 0.002390376maximum number of iterations reached 0.005293939 0.005054126maximum number of iterations reached 0.004750822 0.004563692maximum number of iterations reached 0.008183799 0.007610768maximum number of iterations reached 0.007287432 0.006862579maximum number of iterations reached 0.006680933 0.006284499maximum number of iterations reached 0.007879478 0.007526509maximum number of iterations reached 0.006270222 0.005909266maximum number of iterations reached 0.01034446 0.00909032maximum number of iterations reached 0.002240837 0.002207457maximum number of iterations reached 0.009960269 0.009047477maximum number of iterations reached 0.007708775 0.007236381maximum number of iterations reached 0.005754419 0.005461291maximum number of iterations reached 0.007595849 0.007075417maximum number of iterations reached 0.008795194 0.008164854maximum number of iterations reached 0.009266815 0.00851765maximum number of iterations reached 0.004775463 0.004566055maximum number of iterations reached 0.004784451 0.004606595maximum number of iterations reached 0.01235512 0.01086668

#Visualize accuracy versus values of C. This shows how the accuracy changes based on the level of the cost I chose.   
plot(svm\_caret)



#Obtain metrics of accuracy from training  
confusionMatrix(svm\_caret) #Accuracy (average) : 0.8962

## Cross-Validated (10 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction No Yes  
## No 89.6 10.4  
## Yes 0.0 0.0  
##   
## Accuracy (average) : 0.8962

logistic\_control <- trainControl(method = "cv", number = 10, classProbs = T)  
  
set.seed(100)  
logistic <- train(diabetes ~ ., data = train\_data, method = "glm", family = "binomial", trControl = logistic\_control)  
  
summary(logistic)

##   
## Call:  
## NULL  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.7755 -0.4689 -0.2872 -0.1702 3.2436   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -16.892123 4.035257 -4.186 2.84e-05 \*\*\*  
## age 0.064785 0.004083 15.869 < 2e-16 \*\*\*  
## race1Hispanic -0.082394 0.272714 -0.302 0.762555   
## race1Mexican -0.054968 0.244662 -0.225 0.822237   
## race1White -0.808818 0.159968 -5.056 4.28e-07 \*\*\*  
## race1Other 0.389281 0.235365 1.654 0.098139 .   
## `education9 - 11th Grade` -0.535589 0.226825 -2.361 0.018214 \*   
## `educationHigh School` -0.364389 0.212440 -1.715 0.086297 .   
## `educationSome College` -0.401296 0.210925 -1.903 0.057099 .   
## `educationCollege Grad` -0.508553 0.230369 -2.208 0.027275 \*   
## `hh\_income 5000-9999` 0.134519 0.480480 0.280 0.779501   
## `hh\_income10000-14999` 0.159433 0.422355 0.377 0.705813   
## `hh\_income15000-19999` -0.074774 0.431751 -0.173 0.862503   
## `hh\_income20000-24999` 0.050077 0.427147 0.117 0.906673   
## `hh\_income25000-34999` -0.351043 0.416022 -0.844 0.398776   
## `hh\_income35000-44999` 0.011569 0.411935 0.028 0.977594   
## `hh\_income45000-54999` -0.492290 0.433098 -1.137 0.255676   
## `hh\_income55000-64999` -0.046399 0.432535 -0.107 0.914572   
## `hh\_income65000-74999` -0.017468 0.432414 -0.040 0.967778   
## `hh\_income75000-99999` -0.222732 0.423696 -0.526 0.599105   
## `hh\_incomemore 99999` -0.437939 0.413902 -1.058 0.290021   
## weight -0.026763 0.021250 -1.259 0.207882   
## height 0.050939 0.023453 2.172 0.029860 \*   
## pulse 0.016663 0.004498 3.704 0.000212 \*\*\*  
## bmi 0.159034 0.060176 2.643 0.008221 \*\*   
## phys\_activeYes -0.099119 0.117231 -0.846 0.397830   
## smoke100Yes 0.187673 0.112327 1.671 0.094767 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 2967.2 on 4449 degrees of freedom  
## Residual deviance: 2391.1 on 4423 degrees of freedom  
## AIC: 2445.1  
##   
## Number of Fisher Scoring iterations: 6

confusionMatrix(logistic) #Accuracy (average) : 0.8926

## Cross-Validated (10 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction No Yes  
## No 88.6 9.4  
## Yes 1.0 1.0  
##   
## Accuracy (average) : 0.8955

## Q4: Select a “optimal” model and calculate final evaluation metrics in the test set.

#Checking out info about final model  
svm\_caret$finalModel

## Support Vector Machine object of class "ksvm"   
##   
## SV type: C-svc (classification)   
## parameter : cost C = 0.001   
##   
## Linear (vanilla) kernel function.   
##   
## Number of Support Vectors : 945   
##   
## Objective Function Value : -0.924   
## Training error : 0.10382   
## Probability model included.

#Make predictions in testset  
svm\_pred\_test <- predict(svm\_caret, test\_data)  
  
#Get evaluation metrics from test set  
confusionMatrix(svm\_pred\_test, test\_data$diabetes, positive = "No") #Accuracy 0.8966 #Sensitivity:1.00 #Specificity:0.00

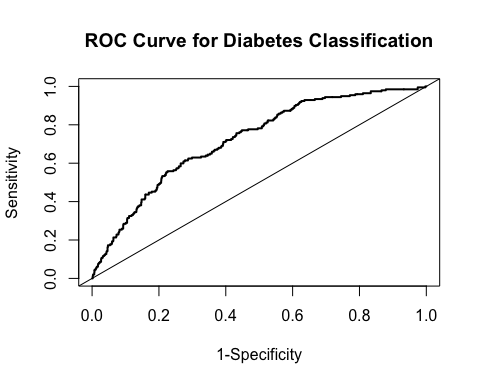
## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction No Yes  
## No 1709 197  
## Yes 0 0  
##   
## Accuracy : 0.8966   
## 95% CI : (0.8821, 0.91)  
## No Information Rate : 0.8966   
## P-Value [Acc > NIR] : 0.519   
##   
## Kappa : 0   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 1.0000   
## Specificity : 0.0000   
## Pos Pred Value : 0.8966   
## Neg Pred Value : NaN   
## Prevalence : 0.8966   
## Detection Rate : 0.8966   
## Detection Prevalence : 1.0000   
## Balanced Accuracy : 0.5000   
##   
## 'Positive' Class : No   
##

#Create ROC Curve for Analysis  
pred.prob <- predict(svm\_caret, test\_data, type = "prob")  
  
#Another potential evaluation: Area under the Receiver Operating Curve (AUROC)  
analysis <- roc(response = test\_data$diabetes, predictor = pred.prob[,2])

## Setting levels: control = No, case = Yes

## Setting direction: controls < cases

plot(1 - analysis$specificities,analysis$sensitivities,type = "l",  
ylab = "Sensitivity",xlab = "1-Specificity",col = "black",lwd = 2,  
main = "ROC Curve for Diabetes Classification")  
abline(a = 0,b = 1)



#varImp(svm\_caret)

## Q5: List and describe at least two limitations/considerations of the model generated by this analysis. Limitations can be analytical or they can be considerations that need to be made regarding how the model would be applied in practice.

Some advantages of support vector classification is that it is very effective with high dimensional data and it can be used for both regression and classification problem. Some disadvantages of this algorithm type is that took more time to train the data, which is most likely due to this being a data set. SVC is not a probabilistic model so we can not explain the diabetes classification in terms of probability, which could be an issue if we wanted to use this model to explain the predictions to those who do not have a machine learning background.