final\_project\_WillisBerkeley

Berkeley Willis

February 27th 2020

## Introduction

What I want to find is to find the best and highest correlative data points to select features to create a predictor for heart disease, and which model will be the most accurate.

## Questions for data

* Cholesterol is usually a well known contributor to heart disease, how strong is this correlative relationship?
* Subjective data from the subject may not be reliable, but in this case how is the correlation of these data points compared to those data points that are factual?
* Is there a particular group of people that are most affected by heart disease, can this give us insight to what to look for.
* What model is best at classifying or correctly predict the presence of heart disease based on the variables I have available.
* In the best models with the highest accuracy, what data points have the biggest impact on prediction.

## Approach

My approach will first be to find those variables with the highest correlation with the target variable and run multiple models with multiple combinations of predictive features. I will then identify those features in the best performing models that had the highest impact on the categorization.

## Details about Data

The dataset that I have selected is from kaggle, and contains a mixture of binary classifications, groups of classifiers, and various numeric (float and integer) datapoints. As well some of these datapoints are objective and therefore are factual, but some of these subjective such as activity levels and are determined more by the subject. Below is the description of that data from <https://www.kaggle.com/sulianova/cardiovascular-disease-dataset>:

* Age | Objective Feature | age | int (days)
* Height | Objective Feature | height | int (cm) |
* Weight | Objective Feature | weight | float (kg) |
* Gender | Objective Feature | gender | categorical code |
* Systolic blood pressure | Examination Feature | ap\_hi | int |
* Diastolic blood pressure | Examination Feature | ap\_lo | int |
* Cholesterol | Examination Feature | cholesterol | 1: normal, 2: above normal, 3: well above normal |
* Glucose | Examination Feature | gluc | 1: normal, 2: above normal, 3: well above normal |
* Smoking | Subjective Feature | smoke | binary |
* Alcohol intake | Subjective Feature | alco | binary |
* Physical activity | Subjective Feature | active | binary |
* Presence or absence of cardiovascular disease | Target Variable | cardio | binary |

The objective and examination features are both fairly factual, one being pretty standard factual information about the person and the other being the results from a doctor examining that subject. However some of the data like cholesterol and glucose is not fully quantified and is simply a classifier of how high those levels were. The other subjective data points is information from the patient so it may be less accurate. The target variable is also factual, and luckily each field is filled out for the entire dataset of 70,000 records.

## Required or Recommended Packages

* ggplot2
* class
* dplyr
* e1071
* caret

## Plots and Diagrams that will be used

* Scatterplots - will illustrate data points and color to identify which of them has heart disease.
* Bar Chart - will illustrate the accuracies of multiple logistic models depending on the variables it incorporates
* Point plots with lines - will illustrate the accruacy of clutering models

## Data Cleaning, Data Summary, and Intial Visualizations

First what needs to be done is a simple process to evaluate the quality of the data and then complete some basic data cleaning and visualizations. The visualizations in combination with some of the data summary overviews will give decent insight to the quality of the data before and after.

Here we are going to clean the data on what was recommended were some realistic boundaries, and the rest would have to be considered either bad data (like negative blood pressure values) or outliers.

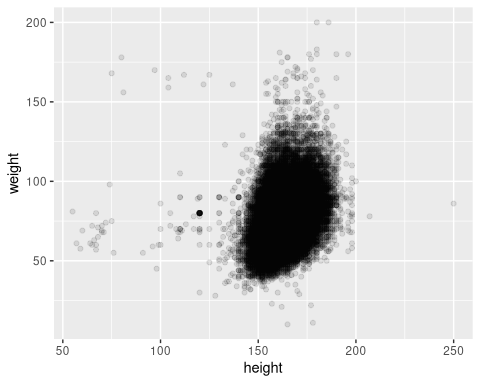
To get a simple summary of the data, I will be using the summary command, and will compare the summary outputs from before and after data cleaning.

Then for the intial visualizations of some of these data points will be via scatter plots with a related data point (weight and height, systolic and diastolic blood pressure). These will not intially incorporate color to identify weather or not that individual has heart disease but is to show outliers and their effect on the data as a whole.

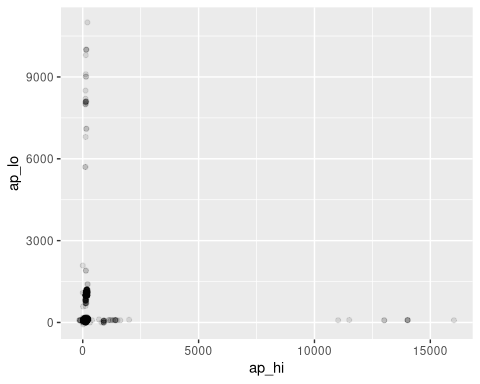
# Nothing really seems out of the ordinary, but pay close attention to the min and max values  
 # There are some extreme outliers for height, weight, ap\_hi, and ap\_lo. Som of which may not be very likely or possible  
 summary(cardio\_data)

## id age gender height   
## Min. : 0 Min. :10798 Min. :1.00 Min. : 55.0   
## 1st Qu.:25007 1st Qu.:17664 1st Qu.:1.00 1st Qu.:159.0   
## Median :50002 Median :19703 Median :1.00 Median :165.0   
## Mean :49972 Mean :19469 Mean :1.35 Mean :164.4   
## 3rd Qu.:74889 3rd Qu.:21327 3rd Qu.:2.00 3rd Qu.:170.0   
## Max. :99999 Max. :23713 Max. :2.00 Max. :250.0   
## weight ap\_hi ap\_lo cholesterol   
## Min. : 10.00 Min. : -150.0 Min. : -70.00 Min. :1.000   
## 1st Qu.: 65.00 1st Qu.: 120.0 1st Qu.: 80.00 1st Qu.:1.000   
## Median : 72.00 Median : 120.0 Median : 80.00 Median :1.000   
## Mean : 74.21 Mean : 128.8 Mean : 96.63 Mean :1.367   
## 3rd Qu.: 82.00 3rd Qu.: 140.0 3rd Qu.: 90.00 3rd Qu.:2.000   
## Max. :200.00 Max. :16020.0 Max. :11000.00 Max. :3.000   
## gluc smoke alco active   
## Min. :1.000 Min. :0.00000 Min. :0.00000 Min. :0.0000   
## 1st Qu.:1.000 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:1.0000   
## Median :1.000 Median :0.00000 Median :0.00000 Median :1.0000   
## Mean :1.226 Mean :0.08813 Mean :0.05377 Mean :0.8037   
## 3rd Qu.:1.000 3rd Qu.:0.00000 3rd Qu.:0.00000 3rd Qu.:1.0000   
## Max. :3.000 Max. :1.00000 Max. :1.00000 Max. :1.0000   
## cardio   
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :0.0000   
## Mean :0.4997   
## 3rd Qu.:1.0000   
## Max. :1.0000

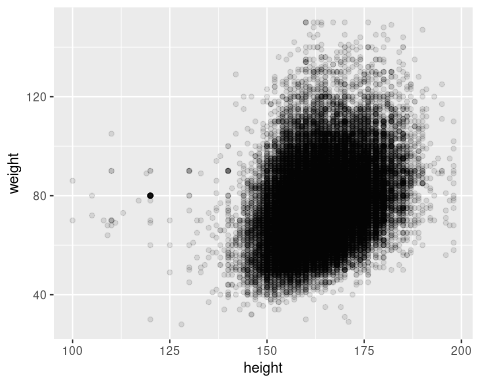
# To get a better look at the outlier here are two scatterplots that will show that there are some we may want to pull out  
 ggplot(cardio\_data, aes(x=height,y=weight)) +  
 geom\_point(alpha=0.1)



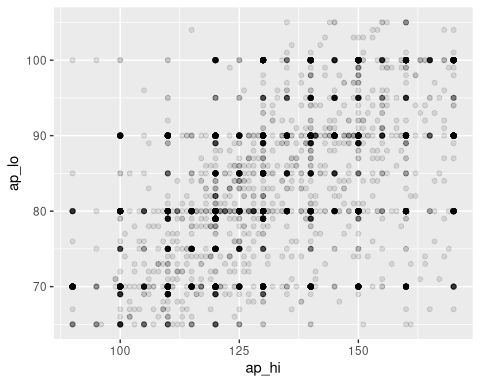
# And blood pressure, systolic (hi) and diastolic (lo) have some outliers that shouldn't exist like negative numbers or extremly high numbers  
 # Otherwise this data would look very different distributed in the graph  
 ggplot(cardio\_data, aes(x=ap\_hi,y=ap\_lo)) +  
 geom\_point(alpha=0.1)



# For height and weight there is a high concentration in the center so I am going to start to pull out some records  
 clean\_cardio\_data = cardio\_data %>%  
 # Removing those outlier based on height,  
 filter(height>=100 & height<=200) %>%  
 # There are a few outliers that have a low weight or even too heigh of a weight  
 filter(weight>=25 & weight<=150) %>%  
 # Going to filter by systolic blood pressure  
 filter(ap\_hi>=90 & ap\_hi<=170) %>%  
 # Finally by diastolic blood pressure  
 filter(ap\_lo>=65 & ap\_lo<=105) %>%  
 # Get non-duplicated records  
 unique  
  
 # Doing one last thing to ensure that the indicator for cardio is easy to handle by putting it as a factor  
 clean\_cardio\_data$disease = factor(NA, levels=c(NA, "No", "Yes"))  
 clean\_cardio\_data$disease[clean\_cardio\_data$cardio==1] = "Yes"  
 clean\_cardio\_data$disease[clean\_cardio\_data$cardio==0] = "No"  
  
 # It looks like there might be some more parsing down that we may need to do but both plots look much better  
 ggplot(clean\_cardio\_data, aes(x=height,y=weight)) +  
 geom\_point(alpha=0.1)



ggplot(clean\_cardio\_data, aes(x=ap\_hi,y=ap\_lo)) +  
 geom\_point(alpha=0.1)



summary(clean\_cardio\_data)

## id age gender height   
## Min. : 0 Min. :10859 Min. :1.000 Min. :100.0   
## 1st Qu.:24930 1st Qu.:17720 1st Qu.:1.000 1st Qu.:159.0   
## Median :50020 Median :19716 Median :1.000 Median :165.0   
## Mean :49970 Mean :19492 Mean :1.352 Mean :164.5   
## 3rd Qu.:74879 3rd Qu.:21343 3rd Qu.:2.000 3rd Qu.:170.0   
## Max. :99999 Max. :23713 Max. :2.000 Max. :198.0   
## weight ap\_hi ap\_lo cholesterol   
## Min. : 28.0 Min. : 90.0 Min. : 65.00 Min. :1.000   
## 1st Qu.: 65.0 1st Qu.:120.0 1st Qu.: 80.00 1st Qu.:1.000   
## Median : 72.0 Median :120.0 Median : 80.00 Median :1.000   
## Mean : 74.2 Mean :126.6 Mean : 81.79 Mean :1.364   
## 3rd Qu.: 82.0 3rd Qu.:140.0 3rd Qu.: 90.00 3rd Qu.:1.000   
## Max. :150.0 Max. :170.0 Max. :105.00 Max. :3.000   
## gluc smoke alco active   
## Min. :1.000 Min. :0.00000 Min. :0.00000 Min. :0.0000   
## 1st Qu.:1.000 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:1.0000   
## Median :1.000 Median :0.00000 Median :0.00000 Median :1.0000   
## Mean :1.226 Mean :0.08761 Mean :0.05302 Mean :0.8034   
## 3rd Qu.:1.000 3rd Qu.:0.00000 3rd Qu.:0.00000 3rd Qu.:1.0000   
## Max. :3.000 Max. :1.00000 Max. :1.00000 Max. :1.0000   
## cardio disease   
## Min. :0.0000 No :32320   
## 1st Qu.:0.0000 Yes:32110   
## Median :0.0000   
## Mean :0.4984   
## 3rd Qu.:1.0000   
## Max. :1.0000

## Views of the Data

The data wille be sliced in multiple ways for various uses. One way that I will be slicing the data will be into training and testing sets for modeling. But with that as well I plan to separate them into the two gender codes available to see if there are different patterns or relationships emerge from similar analyses, as well as seeing how the inclusion of some outliers may affect the accuracy of the models.

As for creating new points, I do not see a benefit since there really isn’t much data mining I can do, technically I can calculate BMI but this may be limited since it is directly related and calculated from weight, hieght, and age. Thus for this project there will be no new data points created.

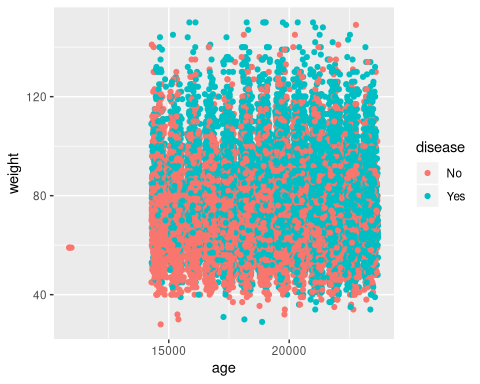
## Machine Learning

I do plan on using machine learning to answer my questions, primarily because I want to see if kmeans or knn might be able to find patterns that aren’t obvious to me, and what model regression or otherwise can create the most accurate model.

## Looking for Intial Patterns

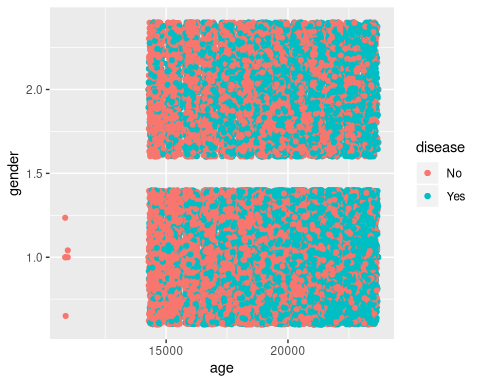
Next some simple noticble patterns in visualizations and through usages of other commands will give more insights to non-obvious patterns.

ggplot(clean\_cardio\_data,aes(x=age,y=weight, color=disease)) +  
 geom\_point()

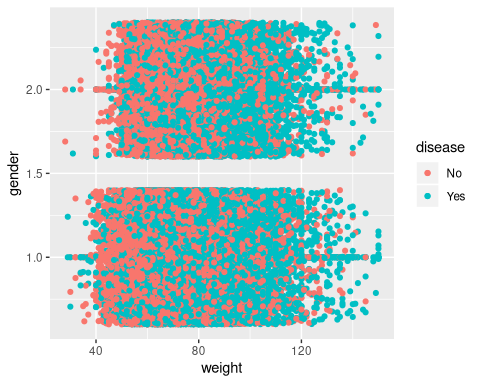


From this intial visualization that I’ve rendered is using age on the x axis and weight on the y, with two colors depicting the detection of heart disease. A pattern we can pickout here is that the heavier and older the person, it seems that they are more prone to heard disaese, since there is a large amount of clustering of the color that was positive for heart disease in the top right corner. Another thing we can see is that across all weights the detection of heart disease at nearly all weights the later in life they get.

ggplot(clean\_cardio\_data,aes(x=age,y=gender, color=disease)) +  
 geom\_point() +  
 geom\_jitter()



ggplot(clean\_cardio\_data,aes(x=weight,y=gender, color=disease)) +  
 geom\_point() +  
 geom\_jitter()



Something I was curious about here is if we separated the records by gender if there would be different patterns. Here I found that there seems be be some interesting trends here, had to put age on the y axis and put on some jitter to visualize it in a meaningful way, but it does affect the gender differently. We can see that gender identification 2 across both weight and age has positive heart disease indicators more spread out and less concentrated than gender identification 1. I think this may lead to predictors to possibly be better or more accurate when considering with records of gender ID 1, but this can only be confirmed when building models.

# This is a bit like swallowing the ocean but we are going to ignore those records in reference to the cardio variable and  
 var(clean\_cardio\_data[setdiff(colnames(clean\_cardio\_data), c("disease")) ])

## id age gender height  
## id 8.325654e+08 280799.56767 5.412516e+01 -468.6116880  
## age 2.807996e+05 6044502.89235 -3.790633e+01 -1800.2639094  
## gender 5.412516e+01 -37.90633 2.280381e-01 1.9356073  
## height -4.686117e+02 -1800.26391 1.935607e+00 62.9699975  
## weight 4.762672e+02 1575.99997 1.023406e+00 33.4089499  
## ap\_hi 1.714944e+03 7135.87682 3.021750e-01 -0.1262669  
## ap\_lo 4.855438e+02 2741.65317 1.801225e-01 0.9065841  
## cholesterol 1.133925e+02 255.54365 -1.338251e-02 -0.3116378  
## gluc 4.214573e+01 136.09952 -6.664102e-03 -0.1073197  
## smoke -2.362485e+01 -34.91996 4.547844e-02 0.4314055  
## alco 3.963718e+00 -16.43621 1.814829e-02 0.1713778  
## active 4.073876e+01 -9.47917 1.145534e-03 -0.0340756  
## cardio 7.507477e+01 286.32449 -5.131505e-04 -0.0829503  
## weight ap\_hi ap\_lo cholesterol  
## id 476.26724305 1.714944e+03 4.855438e+02 113.392467006  
## age 1575.99996970 7.135877e+03 2.741653e+03 255.543654082  
## gender 1.02340607 3.021750e-01 1.801225e-01 -0.013382507  
## height 33.40894992 -1.262669e-01 9.065841e-01 -0.311637795  
## weight 193.59005069 4.897091e+01 2.385204e+01 1.304117618  
## ap\_hi 48.97091033 2.069125e+02 7.839438e+01 1.904083859  
## ap\_lo 23.85203830 7.839438e+01 5.947750e+01 0.830902728  
## cholesterol 1.30411762 1.904084e+00 8.309027e-01 0.462777337  
## gluc 0.81329331 7.270765e-01 2.971494e-01 0.175812902  
## smoke 0.26962638 9.057735e-02 4.643789e-02 0.001657268  
## alco 0.20691998 9.773359e-02 5.756644e-02 0.004900002  
## active -0.09484962 7.555507e-03 -3.287514e-05 0.002431572  
## cardio 1.16146140 3.106981e+00 1.299409e+00 0.074558768  
## gluc smoke alco active  
## id 42.145734805 -23.624846720 3.963718103 4.073876e+01  
## age 136.099523838 -34.919956407 -16.436207297 -9.479170e+00  
## gender -0.006664102 0.045478442 0.018148292 1.145534e-03  
## height -0.107319656 0.431405480 0.171377830 -3.407560e-02  
## weight 0.813293311 0.269626382 0.206919983 -9.484962e-02  
## ap\_hi 0.727076492 0.090577350 0.097733593 7.555507e-03  
## ap\_lo 0.297149426 0.046437887 0.057566435 -3.287514e-05  
## cholesterol 0.175812902 0.001657268 0.004900002 2.431572e-03  
## gluc 0.327921314 -0.001097939 0.001058731 -1.676898e-03  
## smoke -0.001097939 0.079939411 0.021600661 2.727418e-03  
## alco 0.001058731 0.021600661 0.050208568 2.181252e-03  
## active -0.001676898 0.002727418 0.002181252 1.579421e-01  
## cardio 0.024642509 -0.002705311 -0.001170793 -7.475539e-03  
## cardio  
## id 7.507477e+01  
## age 2.863245e+02  
## gender -5.131505e-04  
## height -8.295030e-02  
## weight 1.161461e+00  
## ap\_hi 3.106981e+00  
## ap\_lo 1.299409e+00  
## cholesterol 7.455877e-02  
## gluc 2.464251e-02  
## smoke -2.705311e-03  
## alco -1.170793e-03  
## active -7.475539e-03  
## cardio 2.500012e-01

Something difficult with my dataset is that because it’s so large the shapiro.wiki command to check if data can fit a normal distribution so that test may not be helpful here. So the variance and covariance test due to this may have some limitations, but regression testing and other models should have plenty of data. ggplot(clean\_cardio\_data,aes(x=age,y=gender, color=as.factor(cardio))) + geom\_point() + geom\_jitter()

However, we can see with the variance calculations is that there is a high amount of variance between the age of a person and the cardio variable. In comparisson as well the weight variable does have some variance with cardio that suggests that they may have a fairly positive relationship here, as well as their relationship with both systolic and diastolic blood pressure. The rest of the relationships seem to have extremly small amounts of variance with the cardio indicator are so very small and go just varely positive or negative. This doesn’t really prove much except some strength and directionality of the variance of variables with cardio. This does support what we were able to visualize previously.

Next we will look at the covariance between all of the variables, which should tell us more of the strength of a relationship between two variables.

cor(clean\_cardio\_data[setdiff(colnames(clean\_cardio\_data), c("disease")) ])

## id age gender height weight  
## id 1.0000000000 0.003958284 0.003928134 -0.002046619 0.001186315  
## age 0.0039582840 1.000000000 -0.032287004 -0.092276060 0.046071717  
## gender 0.0039281339 -0.032287004 1.000000000 0.510795071 0.154029089  
## height -0.0020466194 -0.092276060 0.510795071 1.000000000 0.302589930  
## weight 0.0011863150 0.046071717 0.154029089 0.302589930 1.000000000  
## ap\_hi 0.0041318780 0.201777958 0.043990758 -0.001106189 0.244682739  
## ap\_lo 0.0021819403 0.144595857 0.048908899 0.014813746 0.222283697  
## cholesterol 0.0057768218 0.152791349 -0.041195304 -0.057729427 0.137780959  
## gluc 0.0025507014 0.096669949 -0.024369864 -0.023617157 0.102075346  
## smoke -0.0028958718 -0.050235754 0.336838232 0.192281874 0.068539415  
## alco 0.0006130624 -0.029835437 0.169606747 0.096382654 0.066370028  
## active 0.0035526275 -0.009701550 0.006036086 -0.010805074 -0.017153201  
## cardio 0.0052037182 0.232920191 -0.002149166 -0.020906438 0.166952220  
## ap\_hi ap\_lo cholesterol gluc smoke  
## id 0.004131878 2.181940e-03 0.005776822 0.002550701 -0.002895872  
## age 0.201777958 1.445959e-01 0.152791349 0.096669949 -0.050235754  
## gender 0.043990758 4.890890e-02 -0.041195304 -0.024369864 0.336838232  
## height -0.001106189 1.481375e-02 -0.057729427 -0.023617157 0.192281874  
## weight 0.244682739 2.222837e-01 0.137780959 0.102075346 0.068539415  
## ap\_hi 1.000000000 7.066677e-01 0.194583810 0.088267729 0.022271312  
## ap\_lo 0.706667682 1.000000e+00 0.158375418 0.067284294 0.021296861  
## cholesterol 0.194583810 1.583754e-01 1.000000000 0.451315114 0.008616407  
## gluc 0.088267729 6.728429e-02 0.451315114 1.000000000 -0.006781298  
## smoke 0.022271312 2.129686e-02 0.008616407 -0.006781298 1.000000000  
## alco 0.030322264 3.331226e-02 0.032145584 0.008251100 0.340955459  
## active 0.001321665 -1.072611e-05 0.008993984 -0.007368395 0.024272938  
## cardio 0.431990549 3.369756e-01 0.219200537 0.086065488 -0.019136639  
## alco active cardio  
## id 0.0006130624 3.552628e-03 0.005203718  
## age -0.0298354365 -9.701550e-03 0.232920191  
## gender 0.1696067469 6.036086e-03 -0.002149166  
## height 0.0963826542 -1.080507e-02 -0.020906438  
## weight 0.0663700276 -1.715320e-02 0.166952220  
## ap\_hi 0.0303222637 1.321665e-03 0.431990549  
## ap\_lo 0.0333122562 -1.072611e-05 0.336975591  
## cholesterol 0.0321455845 8.993984e-03 0.219200537  
## gluc 0.0082511002 -7.368395e-03 0.086065488  
## smoke 0.3409554593 2.427294e-02 -0.019136639  
## alco 1.0000000000 2.449447e-02 -0.010450093  
## active 0.0244944661 1.000000e+00 -0.037620323  
## cardio -0.0104500932 -3.762032e-02 1.000000000

It seems as if age does have some ssmall relationship with age, weight, cholesterol, and blood pressure readings with cardio. Another thing to note here though that might cause some intereference in modeling is that these variables have a high amount of covariance with each other, which may indicate that these variables are not independent with each other.

## Splitting Datasets for Modeling

For cross-validation to make sure created models aren’t over-fitted we need to split the data into a training set and a testing set. This will be relatively simple since we have such a large data set to sample from, and for this exercise I am just putting 80% of the records into the training data and the remainder to be used in the testing data.

# Going to start split the data very quickly into to two sets with 80% in training and 20% in test  
 training\_index1 = sample(1:nrow(clean\_cardio\_data), size=round(nrow(clean\_cardio\_data)\*0.8))  
 train\_cardio\_data = clean\_cardio\_data[ training\_index1, ]  
 test\_cardio\_data = clean\_cardio\_data[-training\_index1, ]

## Simple Logistic Regression Models

What we are going to do here is create some simple initial models with a single variable being used to predict whether or not that individual has cardio vascular disease. Creation of the model will use the training data and then the prediction will be done with the test data, with a confusion matrix being created for evaluation of the model in comparisson to the actuals.

# Now we are going to checkout various simple linear models and look at their accuracy  
 # Will not look at all of them but those that I suspect would be helpful indicators in a larger model  
  
 s\_glm1 = glm(cardio ~ age, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm1)

##   
## Call:  
## glm(formula = cardio ~ age, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.5423 -1.1371 0.1369 1.1039 1.6382   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.872e+00 7.508e-02 -51.57 <2e-16 \*\*\*  
## age 1.984e-04 3.814e-06 52.02 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 68575 on 51542 degrees of freedom  
## AIC: 68579  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm1, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 3592 2956  
## 1 2291 4047  
##   
## Accuracy : 0.5928   
## 95% CI : (0.5843, 0.6013)  
## No Information Rate : 0.5435   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.1868   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.6106   
## Specificity : 0.5779   
## Pos Pred Value : 0.5486   
## Neg Pred Value : 0.6385   
## Prevalence : 0.4565   
## Detection Rate : 0.2788   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.5942   
##   
## 'Positive' Class : 0   
##

s\_glm2 = glm(cardio ~ gender, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm2)

##   
## Call:  
## glm(formula = cardio ~ gender, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.17891 -1.17891 0.00064 1.17591 1.18018   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) 0.01358 0.02645 0.513 0.608  
## gender -0.01005 0.01846 -0.545 0.586  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 71455 on 51542 degrees of freedom  
## AIC: 71459  
##   
## Number of Fisher Scoring iterations: 3

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm2, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 2332 4216  
## 1 2251 4087  
##   
## Accuracy : 0.4981   
## 95% CI : (0.4895, 0.5068)  
## No Information Rate : 0.6443   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.001   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.5088   
## Specificity : 0.4922   
## Pos Pred Value : 0.3561   
## Neg Pred Value : 0.6448   
## Prevalence : 0.3557   
## Detection Rate : 0.1810   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.5005   
##   
## 'Positive' Class : 0   
##

s\_glm3 = glm(cardio ~ height, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm3)

##   
## Call:  
## glm(formula = cardio ~ height, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.31707 -1.17632 -0.03162 1.17850 1.24936   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.822551 0.183067 4.493 7.02e-06 \*\*\*  
## height -0.005001 0.001112 -4.498 6.85e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 71435 on 51542 degrees of freedom  
## AIC: 71439  
##   
## Number of Fisher Scoring iterations: 3

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm3, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 3456 3092  
## 1 3175 3163  
##   
## Accuracy : 0.5137   
## 95% CI : (0.505, 0.5223)  
## No Information Rate : 0.5146   
## P-Value [Acc > NIR] : 0.5872   
##   
## Kappa : 0.0269   
##   
## Mcnemar's Test P-Value : 0.3003   
##   
## Sensitivity : 0.5212   
## Specificity : 0.5057   
## Pos Pred Value : 0.5278   
## Neg Pred Value : 0.4991   
## Prevalence : 0.5146   
## Detection Rate : 0.2682   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.5134   
##   
## 'Positive' Class : 0   
##

s\_glm4 = glm(cardio ~ weight, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm4)

##   
## Call:  
## glm(formula = cardio ~ weight, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.9947 -1.1346 -0.1052 1.1683 1.6704   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.8239861 0.0502911 -36.27 <2e-16 \*\*\*  
## weight 0.0246063 0.0006694 36.76 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 70027 on 51542 degrees of freedom  
## AIC: 70031  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm4, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 4196 2352  
## 1 3100 3238  
##   
## Accuracy : 0.5769   
## 95% CI : (0.5683, 0.5855)  
## No Information Rate : 0.5662   
## P-Value [Acc > NIR] : 0.00722   
##   
## Kappa : 0.152   
##   
## Mcnemar's Test P-Value : < 2e-16   
##   
## Sensitivity : 0.5751   
## Specificity : 0.5792   
## Pos Pred Value : 0.6408   
## Neg Pred Value : 0.5109   
## Prevalence : 0.5662   
## Detection Rate : 0.3256   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.5772   
##   
## 'Positive' Class : 0   
##

s\_glm5 = glm(cardio ~ ap\_hi, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm5)

##   
## Call:  
## glm(formula = cardio ~ ap\_hi, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.59226 -0.99363 -0.04624 1.05063 2.35682   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -9.5054072 0.1080038 -88.01 <2e-16 \*\*\*  
## ap\_hi 0.0754704 0.0008611 87.65 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 60830 on 51542 degrees of freedom  
## AIC: 60834  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm5, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5250 1298  
## 1 2433 3905  
##   
## Accuracy : 0.7105   
## 95% CI : (0.7025, 0.7183)  
## No Information Rate : 0.5962   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.4191   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.6833   
## Specificity : 0.7505   
## Pos Pred Value : 0.8018   
## Neg Pred Value : 0.6161   
## Prevalence : 0.5962   
## Detection Rate : 0.4074   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7169   
##   
## 'Positive' Class : 0   
##

s\_glm6 = glm(cardio ~ ap\_lo, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm6)

##   
## Call:  
## glm(formula = cardio ~ ap\_lo, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.19077 -1.10837 -0.07882 1.24805 1.90912   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.065940 0.113051 -71.35 <2e-16 \*\*\*  
## ap\_lo 0.098767 0.001383 71.41 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 65312 on 51542 degrees of freedom  
## AIC: 65316  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm6, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5664 884  
## 1 3512 2826  
##   
## Accuracy : 0.6589   
## 95% CI : (0.6506, 0.667)  
## No Information Rate : 0.7121   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.313   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.6173   
## Specificity : 0.7617   
## Pos Pred Value : 0.8650   
## Neg Pred Value : 0.4459   
## Prevalence : 0.7121   
## Detection Rate : 0.4395   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.6895   
##   
## 'Positive' Class : 0   
##

s\_glm7 = glm(cardio ~ cholesterol, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm7)

##   
## Call:  
## glm(formula = cardio ~ cholesterol, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.6920 -1.0760 -0.1685 1.2823 1.2823   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.94406 0.02094 -45.08 <2e-16 \*\*\*  
## cholesterol 0.70077 0.01441 48.64 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 68813 on 51542 degrees of freedom  
## AIC: 68817  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm7, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5467 1081  
## 1 4261 2077  
##   
## Accuracy : 0.5854   
## 95% CI : (0.5769, 0.594)  
## No Information Rate : 0.7549   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.1639   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.5620   
## Specificity : 0.6577   
## Pos Pred Value : 0.8349   
## Neg Pred Value : 0.3277   
## Prevalence : 0.7549   
## Detection Rate : 0.4243   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.6098   
##   
## 'Positive' Class : 0   
##

s\_glm8 = glm(cardio ~ gluc, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm8)

##   
## Call:  
## glm(formula = cardio ~ gluc, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.41283 -1.14886 -0.09494 1.20623 1.20623   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.37045 0.02115 -17.52 <2e-16 \*\*\*  
## gluc 0.30290 0.01579 19.18 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 71077 on 51542 degrees of freedom  
## AIC: 71081  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm8, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5809 739  
## 1 5191 1147  
##   
## Accuracy : 0.5398   
## 95% CI : (0.5312, 0.5484)  
## No Information Rate : 0.8536   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.0689   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.5281   
## Specificity : 0.6082   
## Pos Pred Value : 0.8871   
## Neg Pred Value : 0.1810   
## Prevalence : 0.8536   
## Detection Rate : 0.4508   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.5681   
##   
## 'Positive' Class : 0   
##

s\_glm9 = glm(cardio ~ active, data=train\_cardio\_data, family=binomial)  
 summary(s\_glm9)

##   
## Call:  
## glm(formula = cardio ~ active, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.24478 -1.16117 -0.02483 1.19374 1.19374   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.15702 0.01993 7.879 3.29e-15 \*\*\*  
## active -0.19537 0.02222 -8.792 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 71378 on 51542 degrees of freedom  
## AIC: 71382  
##   
## Number of Fisher Scoring iterations: 3

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(s\_glm9, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5346 1202  
## 1 5008 1330  
##   
## Accuracy : 0.5181   
## 95% CI : (0.5094, 0.5267)  
## No Information Rate : 0.8035   
## P-Value [Acc > NIR] : 1   
##   
## Kappa : 0.0265   
##   
## Mcnemar's Test P-Value : <2e-16   
##   
## Sensitivity : 0.5163   
## Specificity : 0.5253   
## Pos Pred Value : 0.8164   
## Neg Pred Value : 0.2098   
## Prevalence : 0.8035   
## Detection Rate : 0.4149   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.5208   
##   
## 'Positive' Class : 0   
##

The above models and analysis had confirmed something that I was fairly certain of, and that is that no single variable alone is a great indicator of heart disease. Most are just above a coin toss chance of predicting correctly, just around 50%, with exception to systolic blood pressure (ap\_hi) which had a fairly high prediction rating in comparison (~70% during my intial tests). This could paritally be due to some statistical anomaly of sampling.

## Multiple Logistic Regression Modeling

Next

# Now we are going to be using various models with different formulas and try to findout which one is most accurate  
  
 # This one is one of the more promising ones I think because it's puretly objective and oversational features that seemed to have patterns in visualizations and previous models  
 m\_glm1 = glm(cardio ~ age + ap\_hi + ap\_lo, train\_cardio\_data, family=binomial)  
 summary(m\_glm1)

##   
## Call:  
## glm(formula = cardio ~ age + ap\_hi + ap\_lo, family = binomial,   
## data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.75530 -0.96520 -0.02686 0.97014 2.54051   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.265e+01 1.482e-01 -85.343 <2e-16 \*\*\*  
## age 1.544e-04 4.188e-06 36.858 <2e-16 \*\*\*  
## ap\_hi 6.452e-02 1.128e-03 57.190 <2e-16 \*\*\*  
## ap\_lo 1.847e-02 1.907e-03 9.686 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 59339 on 51540 degrees of freedom  
## AIC: 59347  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(m\_glm1, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5106 1442  
## 1 2235 4103  
##   
## Accuracy : 0.7147   
## 95% CI : (0.7068, 0.7224)  
## No Information Rate : 0.5697   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.428   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.6955   
## Specificity : 0.7399   
## Pos Pred Value : 0.7798   
## Neg Pred Value : 0.6474   
## Prevalence : 0.5697   
## Detection Rate : 0.3962   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7177   
##   
## 'Positive' Class : 0   
##

m\_glm2 = glm(cardio ~ age + gender + weight + ap\_hi + ap\_lo, train\_cardio\_data, family=binomial)  
 summary(m\_glm2)

##   
## Call:  
## glm(formula = cardio ~ age + gender + weight + ap\_hi + ap\_lo,   
## family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.90845 -0.95264 -0.03074 0.96347 2.53035   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.299e+01 1.539e-01 -84.425 < 2e-16 \*\*\*  
## age 1.544e-04 4.207e-06 36.712 < 2e-16 \*\*\*  
## gender -1.278e-01 2.114e-02 -6.048 1.47e-09 \*\*\*  
## weight 1.181e-02 7.605e-04 15.526 < 2e-16 \*\*\*  
## ap\_hi 6.273e-02 1.132e-03 55.414 < 2e-16 \*\*\*  
## ap\_lo 1.680e-02 1.913e-03 8.781 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 59080 on 51538 degrees of freedom  
## AIC: 59092  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(m\_glm2, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5083 1465  
## 1 2195 4143  
##   
## Accuracy : 0.716   
## 95% CI : (0.7081, 0.7237)  
## No Information Rate : 0.5648   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.4307   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.6984   
## Specificity : 0.7388   
## Pos Pred Value : 0.7763   
## Neg Pred Value : 0.6537   
## Prevalence : 0.5648   
## Detection Rate : 0.3945   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7186   
##   
## 'Positive' Class : 0   
##

m\_glm3 = glm(cardio ~ age + gender + weight + ap\_hi + ap\_lo + cholesterol, train\_cardio\_data, family=binomial)  
 summary(m\_glm3)

##   
## Call:  
## glm(formula = cardio ~ age + gender + weight + ap\_hi + ap\_lo +   
## cholesterol, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.93367 -0.92428 -0.05205 0.93503 2.52125   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.291e+01 1.545e-01 -83.573 < 2e-16 \*\*\*  
## age 1.426e-04 4.252e-06 33.537 < 2e-16 \*\*\*  
## gender -9.362e-02 2.135e-02 -4.385 1.16e-05 \*\*\*  
## weight 9.746e-03 7.703e-04 12.651 < 2e-16 \*\*\*  
## ap\_hi 6.036e-02 1.135e-03 53.171 < 2e-16 \*\*\*  
## ap\_lo 1.606e-02 1.928e-03 8.333 < 2e-16 \*\*\*  
## cholesterol 4.567e-01 1.590e-02 28.733 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 58218 on 51537 degrees of freedom  
## AIC: 58232  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(m\_glm3, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5136 1412  
## 1 2202 4136  
##   
## Accuracy : 0.7195   
## 95% CI : (0.7117, 0.7273)  
## No Information Rate : 0.5695   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.4378   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.6999   
## Specificity : 0.7455   
## Pos Pred Value : 0.7844   
## Neg Pred Value : 0.6526   
## Prevalence : 0.5695   
## Detection Rate : 0.3986   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7227   
##   
## 'Positive' Class : 0   
##

m\_glm4 = glm(cardio ~ age + gender + weight + ap\_hi + ap\_lo + cholesterol + gluc, train\_cardio\_data, family=binomial)  
 summary(m\_glm4)

##   
## Call:  
## glm(formula = cardio ~ age + gender + weight + ap\_hi + ap\_lo +   
## cholesterol + gluc, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.93807 -0.92533 -0.05659 0.93456 2.51914   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.285e+01 1.547e-01 -83.025 < 2e-16 \*\*\*  
## age 1.436e-04 4.259e-06 33.727 < 2e-16 \*\*\*  
## gender -9.515e-02 2.136e-02 -4.454 8.43e-06 \*\*\*  
## weight 9.953e-03 7.713e-04 12.904 < 2e-16 \*\*\*  
## ap\_hi 6.044e-02 1.136e-03 53.189 < 2e-16 \*\*\*  
## ap\_lo 1.591e-02 1.929e-03 8.247 < 2e-16 \*\*\*  
## cholesterol 5.105e-01 1.790e-02 28.519 < 2e-16 \*\*\*  
## gluc -1.366e-01 2.032e-02 -6.721 1.81e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 58172 on 51536 degrees of freedom  
## AIC: 58188  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(m\_glm4, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5138 1410  
## 1 2203 4135  
##   
## Accuracy : 0.7196   
## 95% CI : (0.7118, 0.7274)  
## No Information Rate : 0.5697   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.438   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.6999   
## Specificity : 0.7457   
## Pos Pred Value : 0.7847   
## Neg Pred Value : 0.6524   
## Prevalence : 0.5697   
## Detection Rate : 0.3987   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7228   
##   
## 'Positive' Class : 0   
##

m\_glm5 = glm(cardio ~ age + gender + weight + ap\_hi + ap\_lo + cholesterol + gluc + active, train\_cardio\_data, family=binomial)  
 summary(m\_glm5)

##   
## Call:  
## glm(formula = cardio ~ age + gender + weight + ap\_hi + ap\_lo +   
## cholesterol + gluc + active, family = binomial, data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.92509 -0.92077 -0.05626 0.93295 2.53982   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.266e+01 1.557e-01 -81.292 < 2e-16 \*\*\*  
## age 1.436e-04 4.263e-06 33.690 < 2e-16 \*\*\*  
## gender -9.348e-02 2.139e-02 -4.371 1.24e-05 \*\*\*  
## weight 9.844e-03 7.723e-04 12.747 < 2e-16 \*\*\*  
## ap\_hi 6.058e-02 1.138e-03 53.248 < 2e-16 \*\*\*  
## ap\_lo 1.591e-02 1.931e-03 8.235 < 2e-16 \*\*\*  
## cholesterol 5.143e-01 1.791e-02 28.716 < 2e-16 \*\*\*  
## gluc -1.399e-01 2.034e-02 -6.879 6.02e-12 \*\*\*  
## active -2.486e-01 2.522e-02 -9.859 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 58075 on 51535 degrees of freedom  
## AIC: 58093  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(m\_glm5, test\_cardio\_data, type="response"))) )

## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5121 1427  
## 1 2168 4170  
##   
## Accuracy : 0.721   
## 95% CI : (0.7132, 0.7287)  
## No Information Rate : 0.5657   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.4408   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.7026   
## Specificity : 0.7450   
## Pos Pred Value : 0.7821   
## Neg Pred Value : 0.6579   
## Prevalence : 0.5657   
## Detection Rate : 0.3974   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7238   
##   
## 'Positive' Class : 0   
##

m\_glm6 = glm(cardio ~ age + gender + weight + ap\_hi + ap\_lo + cholesterol + gluc + smoke + alco + active, train\_cardio\_data, family=binomial)  
 summary(m\_glm6)

##   
## Call:  
## glm(formula = cardio ~ age + gender + weight + ap\_hi + ap\_lo +   
## cholesterol + gluc + smoke + alco + active, family = binomial,   
## data = train\_cardio\_data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.91836 -0.92131 -0.05146 0.93247 2.53781   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.272e+01 1.562e-01 -81.438 < 2e-16 \*\*\*  
## age 1.422e-04 4.268e-06 33.324 < 2e-16 \*\*\*  
## gender -4.876e-02 2.266e-02 -2.152 0.031409 \*   
## weight 1.002e-02 7.733e-04 12.958 < 2e-16 \*\*\*  
## ap\_hi 6.073e-02 1.139e-03 53.310 < 2e-16 \*\*\*  
## ap\_lo 1.598e-02 1.933e-03 8.265 < 2e-16 \*\*\*  
## cholesterol 5.198e-01 1.794e-02 28.970 < 2e-16 \*\*\*  
## gluc -1.418e-01 2.036e-02 -6.967 3.23e-12 \*\*\*  
## smoke -1.457e-01 4.017e-02 -3.627 0.000287 \*\*\*  
## alco -2.223e-01 4.947e-02 -4.494 6.99e-06 \*\*\*  
## active -2.432e-01 2.523e-02 -9.637 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 71455 on 51543 degrees of freedom  
## Residual deviance: 58027 on 51533 degrees of freedom  
## AIC: 58049  
##   
## Number of Fisher Scoring iterations: 4

confusionMatrix(table(test\_cardio\_data$cardio, round(predict(m\_glm6, test\_cardio\_data, type="response"))) )

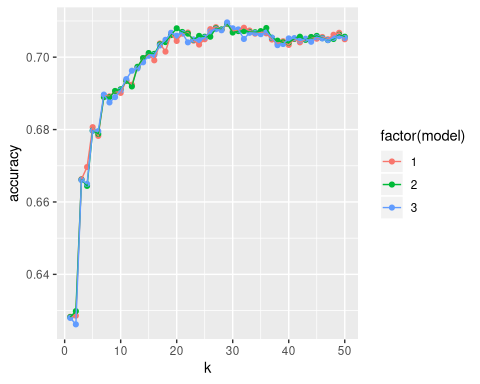
## Confusion Matrix and Statistics  
##   
##   
## 0 1  
## 0 5131 1417  
## 1 2167 4171  
##   
## Accuracy : 0.7219   
## 95% CI : (0.714, 0.7296)  
## No Information Rate : 0.5664   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.4425   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.7031   
## Specificity : 0.7464   
## Pos Pred Value : 0.7836   
## Neg Pred Value : 0.6581   
## Prevalence : 0.5664   
## Detection Rate : 0.3982   
## Detection Prevalence : 0.5081   
## Balanced Accuracy : 0.7247   
##   
## 'Positive' Class : 0   
##

With the usage of logistic regression modeles with multiple independent variables, we do see a mark improvement over most of the models with only a single variable. THe average accuracy of those single variable models was typically around 50% or 60% with the exception of the systolic blood pressure measurement which is arond 70%. The accuracy of most of these models seems to top out around 72.x% during my multitude of tests and is usually either the 4th model or the final that incorporates all datapoints.

## Clustering Analysis

Then some clusterin ganalysis should be done to see if there are more accurate patterns or even better prediction models using capabilities of cluster analysis. If they are more accurate what are they using or what variables are being used heavily in the prediction, is something that should be investigated.

# First let's use the knn algorithm and see if there is any patterns it can draw on  
  
 # Colnames of all available variables  
 var\_cols1 = c("age", "gender", "height", "weight", "ap\_hi", "ap\_lo", "cholesterol", "gluc", "smoke", "alco", "active")  
 var\_cols2 = c("age", "gender", "height", "weight", "ap\_hi", "ap\_lo", "cholesterol", "gluc", "active")  
 var\_cols3 = c("age", "height", "weight", "ap\_hi", "ap\_lo", "cholesterol", "gluc", "active")  
  
 # Going to run for a number of values of k for the clustering  
 knn\_accs1 = sapply(1:50, function(x){  
 confusionMatrix(test\_cardio\_data$disease, knn( train=train\_cardio\_data[var\_cols1], test=test\_cardio\_data[var\_cols1], cl=train\_cardio\_data$disease, k=x))$overall['Accuracy']  
 })  
  
 knn\_accs2 = sapply(1:50, function(x){  
 confusionMatrix(test\_cardio\_data$disease, knn( train=train\_cardio\_data[var\_cols2], test=test\_cardio\_data[var\_cols2], cl=train\_cardio\_data$disease, k=x))$overall['Accuracy']  
 })  
  
 knn\_accs3 = sapply(1:50, function(x){  
 confusionMatrix(test\_cardio\_data$disease, knn( train=train\_cardio\_data[var\_cols3], test=test\_cardio\_data[var\_cols3], cl=train\_cardio\_data$disease, k=x))$overall['Accuracy']  
 })  
  
 # So we aren't able to directly visualize what's being identified here but we can take a look at the accuracies of each model created with a different value of k  
 knn\_models = rbind(data.frame(k=1:50, accuracy=knn\_accs1, model="1"), data.frame(k=1:50, accuracy=knn\_accs2, model="2"), data.frame(k=1:50, accuracy=knn\_accs3, model="3"))  
 ggplot(knn\_models, aes(x=k, y=accuracy, color=factor(model))) +  
 geom\_point() +  
 geom\_line()



So there seems to be a point to which the accuracy of knn will max out, and for most of the cluster analysis that are run with various combinations of columns. The unfortunate part is that it maxes out around the same rate as the multi variate logistic models. They have relatively the same amount of accuracy, and the most effective value of k is around 25. This does make me question the accuracy or improvements that those columns I parsed out of the column sets, if they didn’t have any noticeable effect on the accuracy of the knn cluster analysis. That would lead me to believe that for a cluster analysis the variables that provide the best value are in var\_cols3, and it is possible to remove more of those.

## Gender Differences

Now that we have done enough analysis for some of this we are going to look at the same types of analysis but we are going to separate out the datasets based on gender and try to rebuild the same models and test their accuracy. The question becomes if we build separate models for different genders, can we improve the accuracy of one or more models. This time instead of just looking at the raw results here we are going visualize them.

# Split training data by the  
 training\_index\_gen1 = sample(1:nrow(clean\_cardio\_data[ clean\_cardio\_data$gender==1, ]), size=round(nrow(clean\_cardio\_data[ clean\_cardio\_data$gender==1, ])\*0.8))  
 train\_cardio\_data\_gen1 = clean\_cardio\_data[ training\_index\_gen1, ]  
 test\_cardio\_data\_gen1 = clean\_cardio\_data[-training\_index\_gen1, ]  
  
 training\_index\_gen2 = sample(1:nrow(clean\_cardio\_data[ clean\_cardio\_data$gender==2, ]), size=round(nrow(clean\_cardio\_data[ clean\_cardio\_data$gender==2, ])\*0.8))  
 train\_cardio\_data\_gen2 = clean\_cardio\_data[ training\_index\_gen2, ]  
 test\_cardio\_data\_gen2 = clean\_cardio\_data[-training\_index\_gen2, ]  
  
 # No re-run and test the accuracy of some of the better performing models to limit our scope here, but simply removing gender as a variable in these sets  
 m\_glm1\_gen1 = glm(cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol + gluc, train\_cardio\_data\_gen1, family=binomial)  
 summary(m\_glm1\_gen1)

##   
## Call:  
## glm(formula = cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol +   
## gluc, family = binomial, data = train\_cardio\_data\_gen1)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.9562 -0.9235 -0.3373 0.9320 2.5310   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.288e+01 1.899e-01 -67.844 < 2e-16 \*\*\*  
## age 1.388e-04 5.267e-06 26.358 < 2e-16 \*\*\*  
## weight 8.900e-03 9.542e-04 9.327 < 2e-16 \*\*\*  
## ap\_hi 6.135e-02 1.416e-03 43.315 < 2e-16 \*\*\*  
## ap\_lo 1.545e-02 2.393e-03 6.458 1.06e-10 \*\*\*  
## cholesterol 5.102e-01 2.247e-02 22.707 < 2e-16 \*\*\*  
## gluc -1.339e-01 2.542e-02 -5.268 1.38e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 46317 on 33410 degrees of freedom  
## Residual deviance: 37686 on 33404 degrees of freedom  
## AIC: 37700  
##   
## Number of Fisher Scoring iterations: 4

conf\_matx1\_gen1 = confusionMatrix(table(test\_cardio\_data\_gen1$cardio, round(predict(m\_glm1\_gen1, test\_cardio\_data\_gen1, type="response"))) )  
  
 m\_glm2\_gen1 = glm(cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol + gluc + active, train\_cardio\_data\_gen1, family=binomial)  
 summary(m\_glm2\_gen1)

##   
## Call:  
## glm(formula = cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol +   
## gluc + active, family = binomial, data = train\_cardio\_data\_gen1)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.9429 -0.9207 -0.3334 0.9300 2.5523   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.270e+01 1.910e-01 -66.469 < 2e-16 \*\*\*  
## age 1.389e-04 5.272e-06 26.339 < 2e-16 \*\*\*  
## weight 8.818e-03 9.554e-04 9.229 < 2e-16 \*\*\*  
## ap\_hi 6.143e-02 1.418e-03 43.326 < 2e-16 \*\*\*  
## ap\_lo 1.563e-02 2.396e-03 6.525 6.78e-11 \*\*\*  
## cholesterol 5.141e-01 2.248e-02 22.867 < 2e-16 \*\*\*  
## gluc -1.381e-01 2.544e-02 -5.429 5.66e-08 \*\*\*  
## active -2.552e-01 3.114e-02 -8.197 2.47e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 46317 on 33410 degrees of freedom  
## Residual deviance: 37619 on 33403 degrees of freedom  
## AIC: 37635  
##   
## Number of Fisher Scoring iterations: 4

conf\_matx2\_gen1 = confusionMatrix(table(test\_cardio\_data\_gen1$cardio, round(predict(m\_glm2\_gen1, test\_cardio\_data\_gen1, type="response"))) )  
  
 m\_glm3\_gen1 = glm(cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol + gluc + smoke + alco + active, train\_cardio\_data\_gen1, family=binomial)  
 summary(m\_glm3\_gen1)

##   
## Call:  
## glm(formula = cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol +   
## gluc + smoke + alco + active, family = binomial, data = train\_cardio\_data\_gen1)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.9161 -0.9197 -0.3295 0.9289 2.5435   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.272e+01 1.913e-01 -66.483 < 2e-16 \*\*\*  
## age 1.371e-04 5.280e-06 25.972 < 2e-16 \*\*\*  
## weight 9.341e-03 9.595e-04 9.736 < 2e-16 \*\*\*  
## ap\_hi 6.167e-02 1.420e-03 43.422 < 2e-16 \*\*\*  
## ap\_lo 1.570e-02 2.398e-03 6.546 5.93e-11 \*\*\*  
## cholesterol 5.177e-01 2.252e-02 22.988 < 2e-16 \*\*\*  
## gluc -1.396e-01 2.548e-02 -5.480 4.25e-08 \*\*\*  
## smoke -2.023e-01 4.770e-02 -4.241 2.23e-05 \*\*\*  
## alco -2.347e-01 6.083e-02 -3.858 0.000114 \*\*\*  
## active -2.478e-01 3.117e-02 -7.950 1.86e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 46317 on 33410 degrees of freedom  
## Residual deviance: 37570 on 33401 degrees of freedom  
## AIC: 37590  
##   
## Number of Fisher Scoring iterations: 4

conf\_matx3\_gen1 = confusionMatrix(table(test\_cardio\_data\_gen1$cardio, round(predict(m\_glm3\_gen1, test\_cardio\_data\_gen1, type="response"))) )  
  
  
 m\_glm1\_gen2 = glm(cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol + gluc, train\_cardio\_data\_gen2, family=binomial)  
 summary(m\_glm1\_gen2)

##   
## Call:  
## glm(formula = cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol +   
## gluc, family = binomial, data = train\_cardio\_data\_gen2)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.8280 -0.9125 -0.4234 0.9286 2.5018   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.322e+01 2.608e-01 -50.691 < 2e-16 \*\*\*  
## age 1.435e-04 7.212e-06 19.895 < 2e-16 \*\*\*  
## weight 1.005e-02 1.299e-03 7.739 1.00e-14 \*\*\*  
## ap\_hi 6.017e-02 1.899e-03 31.682 < 2e-16 \*\*\*  
## ap\_lo 1.868e-02 3.237e-03 5.769 7.99e-09 \*\*\*  
## cholesterol 4.903e-01 3.013e-02 16.270 < 2e-16 \*\*\*  
## gluc -1.096e-01 3.401e-02 -3.224 0.00127 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25130 on 18132 degrees of freedom  
## Residual deviance: 20372 on 18126 degrees of freedom  
## AIC: 20386  
##   
## Number of Fisher Scoring iterations: 4

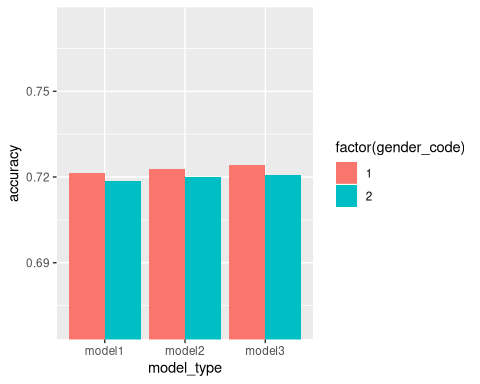
conf\_matx1\_gen2 = confusionMatrix(table(test\_cardio\_data\_gen2$cardio, round(predict(m\_glm1\_gen2, test\_cardio\_data\_gen2, type="response"))) )  
  
 m\_glm2\_gen2 = glm(cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol + gluc + active, train\_cardio\_data\_gen2, family=binomial)  
 summary(m\_glm2\_gen2)

##   
## Call:  
## glm(formula = cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol +   
## gluc + active, family = binomial, data = train\_cardio\_data\_gen2)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.8132 -0.9093 -0.4159 0.9254 2.5302   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.301e+01 2.624e-01 -49.571 < 2e-16 \*\*\*  
## age 1.433e-04 7.222e-06 19.838 < 2e-16 \*\*\*  
## weight 9.963e-03 1.302e-03 7.655 1.94e-14 \*\*\*  
## ap\_hi 6.042e-02 1.902e-03 31.768 < 2e-16 \*\*\*  
## ap\_lo 1.874e-02 3.244e-03 5.777 7.61e-09 \*\*\*  
## cholesterol 4.961e-01 3.017e-02 16.445 < 2e-16 \*\*\*  
## gluc -1.144e-01 3.407e-02 -3.358 0.000786 \*\*\*  
## active -3.014e-01 4.253e-02 -7.087 1.37e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25130 on 18132 degrees of freedom  
## Residual deviance: 20321 on 18125 degrees of freedom  
## AIC: 20337  
##   
## Number of Fisher Scoring iterations: 4

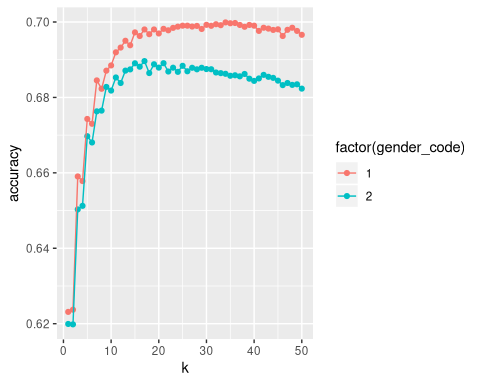
conf\_matx2\_gen2 = confusionMatrix(table(test\_cardio\_data\_gen2$cardio, round(predict(m\_glm2\_gen2, test\_cardio\_data\_gen2, type="response"))) )  
  
 m\_glm3\_gen2 = glm(cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol + gluc + smoke + alco + active, train\_cardio\_data\_gen2, family=binomial)  
 summary(m\_glm3\_gen2)

##   
## Call:  
## glm(formula = cardio ~ age + weight + ap\_hi + ap\_lo + cholesterol +   
## gluc + smoke + alco + active, family = binomial, data = train\_cardio\_data\_gen2)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.8247 -0.9099 -0.4137 0.9272 2.5255   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.301e+01 2.626e-01 -49.563 < 2e-16 \*\*\*  
## age 1.421e-04 7.232e-06 19.642 < 2e-16 \*\*\*  
## weight 1.033e-02 1.308e-03 7.903 2.73e-15 \*\*\*  
## ap\_hi 6.055e-02 1.904e-03 31.806 < 2e-16 \*\*\*  
## ap\_lo 1.877e-02 3.245e-03 5.785 7.24e-09 \*\*\*  
## cholesterol 4.980e-01 3.020e-02 16.491 < 2e-16 \*\*\*  
## gluc -1.162e-01 3.410e-02 -3.409 0.000653 \*\*\*  
## smoke -1.326e-01 6.412e-02 -2.068 0.038650 \*   
## alco -1.365e-01 8.112e-02 -1.682 0.092524 .   
## active -2.967e-01 4.256e-02 -6.971 3.15e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25130 on 18132 degrees of freedom  
## Residual deviance: 20311 on 18123 degrees of freedom  
## AIC: 20331  
##   
## Number of Fisher Scoring iterations: 4

conf\_matx3\_gen2 = confusionMatrix(table(test\_cardio\_data\_gen2$cardio, round(predict(m\_glm3\_gen2, test\_cardio\_data\_gen2, type="response"))) )  
  
 # Now to create a dataframe with the accuracies of all of these models and visualize them in a comparisson4  
 accs\_models\_gens\_c = c(conf\_matx1\_gen1$overall["Accuracy"], conf\_matx2\_gen1$overall["Accuracy"], conf\_matx3\_gen1$overall["Accuracy"],  
 conf\_matx1\_gen2$overall["Accuracy"], conf\_matx2\_gen2$overall["Accuracy"], conf\_matx3\_gen2$overall["Accuracy"]  
 )  
  
 accs\_models\_gen\_df = data.frame(  
 gender\_code = c(1, 1, 1, 2, 2, 2),  
 model\_type = c("model1", "model2", "model3", "model1", "model2", "model3"),  
 accuracy = accs\_models\_gens\_c  
 )  
  
 # Going to create a barchart here  
 ggplot(accs\_models\_gen\_df, aes(x=model\_type, y=accuracy, fill=factor(gender\_code))) +  
 geom\_bar(stat="identity", position=position\_dodge()) +  
 # Our data is so close and we know the max here so we are gonig to scale y here  
 coord\_cartesian(ylim=c(min(accs\_models\_gens\_c)-0.05,max(accs\_models\_gens\_c)+0.05))



# Finally let's take a look at the clustering model between the two, and comparsre the accuracy of both  
 var\_cols\_gen = c("age", "height", "weight", "ap\_hi", "ap\_lo", "cholesterol", "gluc", "active")  
  
 knn\_accs\_gen1 = sapply(1:50, function(x){  
 confusionMatrix(test\_cardio\_data\_gen1$disease, knn( train=train\_cardio\_data\_gen1[var\_cols\_gen], test=test\_cardio\_data\_gen1[var\_cols\_gen], cl=train\_cardio\_data\_gen1$disease, k=x))$overall['Accuracy']  
 })  
  
 knn\_accs\_gen2 = sapply(1:50, function(x){  
 confusionMatrix(test\_cardio\_data\_gen2$disease, knn( train=train\_cardio\_data\_gen2[var\_cols\_gen], test=test\_cardio\_data\_gen2[var\_cols\_gen], cl=train\_cardio\_data\_gen2$disease, k=x))$overall['Accuracy']  
 })  
  
 # Very similarly to our previous cluster analysis accuracies comparison, we are goint to look at these accruacis in a line plot  
 knn\_models\_gen = rbind(data.frame(k=1:50, accuracy=knn\_accs\_gen1, gender\_code="1"), data.frame(k=1:50, accuracy=knn\_accs\_gen2, gender\_code="2"))  
 ggplot(knn\_models\_gen, aes(x=k, y=accuracy, color=factor(gender\_code))) +  
 geom\_point() +  
 geom\_line()



Here we had some facinating results, but my overall conclusion is that these models do differ in accuracy depending on the gender, and oculd possibly effect the accuracy of a model the incorporates both. For the first set of multi-variate Logistic models, we see a trend upwards to the later models becoming more accurate but an interesting difference in each is that the accuracy of models that are built and testing off of gender code 2 seem to be just less accurate.

Now when it comes to the cluster models using the knn algorithm we see a vast difference in the performance of modeling and prediction based off different genders. We still see that clustering algorithms are still overall just a little less accurate with a plateau around k=25, but we see that gender 2’s models are much less accurate for predictions.

## Limitations

The limitations that are faced here are primarily due to data points that are available and possibly time for new models to be designed and built. The limitations of the data would be the limtation of features that are more opinion driven by answers of the subject, I do think it is possibly better to have their view or rating on their activity or intakes of certain substances but it may be too subjective. For example instead of gluc being their response to a question about how much sugar they consume, we could instead take their blood sugar levels, and similar methods for other subjective data points that we have here.

## Conclusions

The take away here is that the information gathered was enough to answer many of the starting questions here. The more accurate models that were designed in this project primarily on logistic multi-variate models that incorporated more of the variables available in this project, and clustering models didn’t find any patterns that made it more accurate that logistic models. This was a bit of a surprise considering I thought there may be more patterns that were less linear, but the clustering algorithms didn’t perform predictions better. Another conclusion that can be drawn is that gender in this case can play a considerable role in the accuracy of these models.

## Future Works

What should be done in the future is by gathering more objective data where possible, and attempting new patterns to both slice up the data and create new models that may have better predictions performance.