IST707 - Final Project - Stroke of genius

Garen Moghoyan

6/17/2021

Introduction

Per the World Health Organization, stroke is the second leading cause of death in the world with over 6 million casualties annually (11% of global deaths). Per the Centers for Disease Control and Preventions, more than 795,000 people suffer a stroke every year in the United States; of those, 610,000 are first-time strokes.

While the signs of a stroke are universal for both men and women (sudden numbness, sudden confusion, sudden trouble seeing and walking and sudden severe headache), there are nevertheless conditions that increase the risk of a stroke. A previous stroke, high blood pressure, high cholesterol, heart disease, diabetes and sickle cell disease all seem to play a role.

This study attempts to recognize potential causes of stroke by conducting clustering to identify trends (which factors seem to impact the likelihood of a stroke) and by buiding a model that predicts the likelihood of a stroke in patients.

Analysis and Models

About the data

The dataset can be found at www.kaggle.com/fedesoriano/stroke-prediction-dataset. It contains information about 5,110 patients, each with 12 variables, ranging from a system-assigned ID number to whether or not the patient suffered a stroke.

The data was imported into R. The first five lines of the dataset were used to get acquainted with the variables.

id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
9046	Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
51676	Female	61	0	0	Yes	Self-employed	Rural	202.21	N/A	never smoked	1
31112	Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
60182	Female	49	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
1665	Female	79	1	0	Yes	Self-employed	Rural	174.12	24	never smoked	1

Some values (hypertension, heart disease and stroke) are binary: a 1 indicates the patient suffers from that condition. The BMI variable seems to have N/A values.

Checking for missing values, complete and incomplete rows and getting overall structure.

```
length(which(is.na(strokeData)))
```

```
## [1] 0
```

```
nrow(strokeData[complete.cases(strokeData),])
```

```
## [1] 5110

nrow(strokeData[!complete.cases(strokeData),])

## [1] 0
```

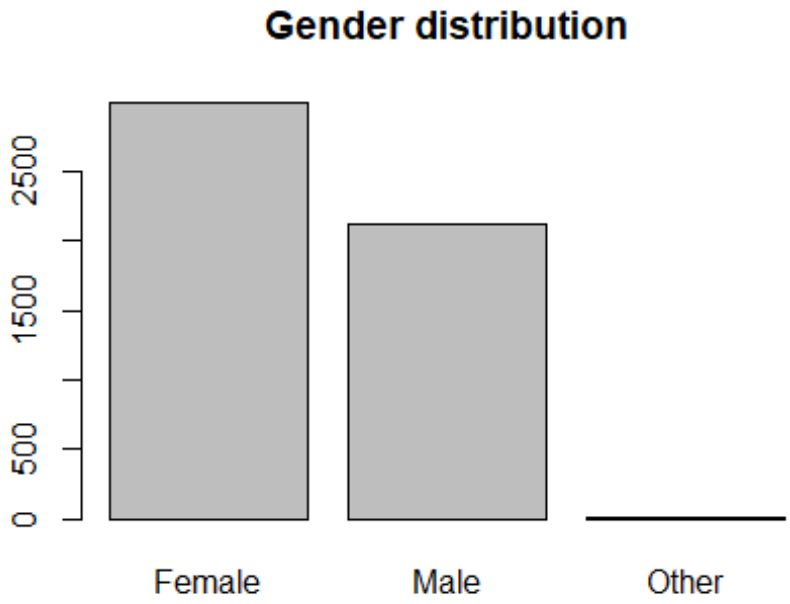
The data was imported in R and processed for analysis, the ID field was removed and missing (N/A) values were identified and dealt with. To manipulate it further, some values were converted to factor.

gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
Male	67	0	1	Yes	Private	Urban	228.69	36.60	formerly smoked	1
Female	61	0	0	Yes	Self-employed	Rural	202.21	28.89	never smoked	1
Male	80	0	1	Yes	Private	Rural	105.92	32.50	never smoked	1
Female	49	0	0	Yes	Private	Urban	171.23	34.40	smokes	1
Female	79	1	0	Yes	Self-employed	Rural	174.12	24.00	never smoked	1

```
##     gender          age         hypertension heart_disease ever_married
##  Female:2994   Min.   : 0.08   0:4612       0:4834         No :1757
##  Male  :2115   1st Qu.:25.00   1: 498       1: 276         Yes:3353
##  Other :   1   Median :45.00
##                Mean   :43.23
##                3rd Qu.:61.00
##                Max.   :82.00
##         work_type    Residence_type avg_glucose_level      bmi
##  children     : 687   Rural:2514     Min.   : 55.12    Min.   :10.30
##  Govt_job     : 657   Urban:2596     1st Qu.: 77.25    1st Qu.:23.80
##  Never_worked :  22                  Median : 91.89    Median :28.40
##  Private      :2925                  Mean   :106.15    Mean   :28.89
##  Self-employed: 819                  3rd Qu.:114.09    3rd Qu.:32.80
##                                      Max.   :271.74    Max.   :97.60
##         smoking_status stroke
##  formerly smoked: 885   0:4861
##  never smoked   :1892   1: 249
##  smokes         : 789
##  Unknown        :1544
##
##
```

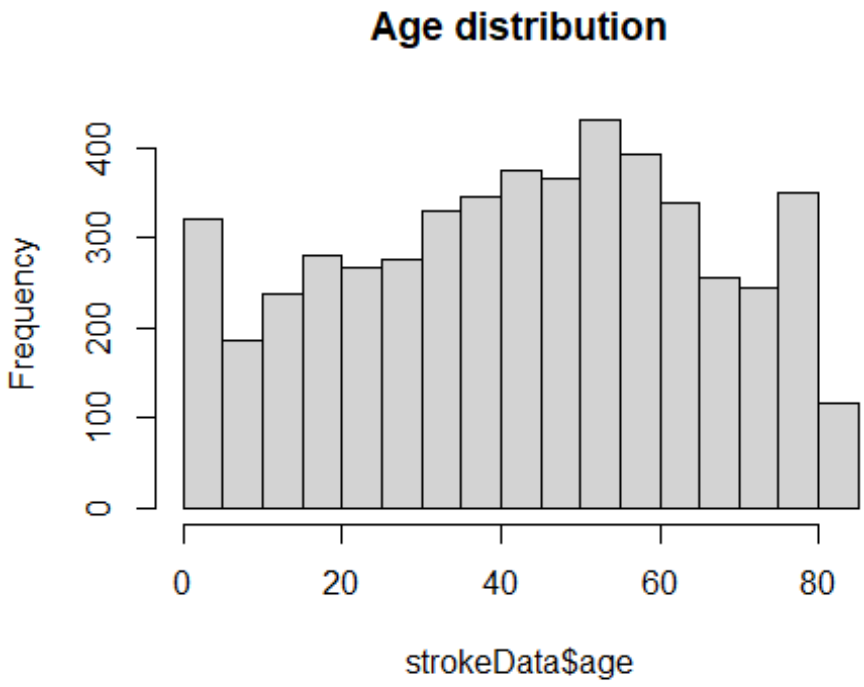
Taking a look at the variables

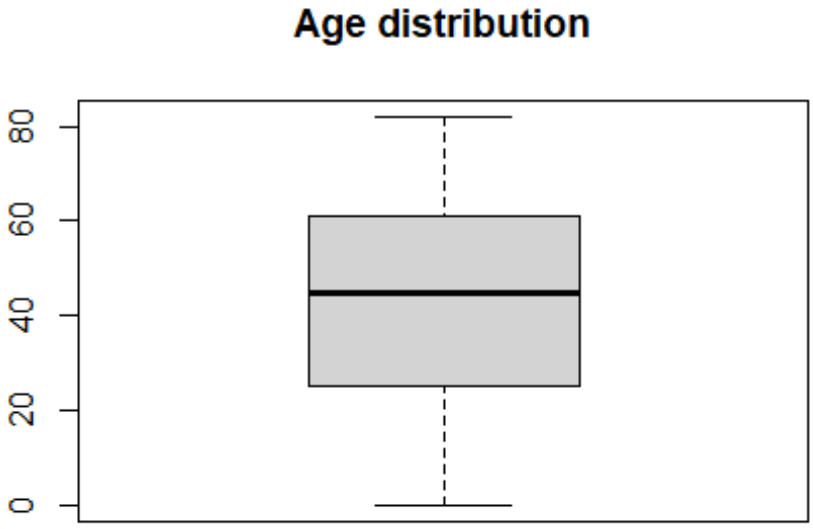
Gender distribution is as follows: 2,994 females (58.59%), 2,115 males (41.39%) and 1 other (0.02%).



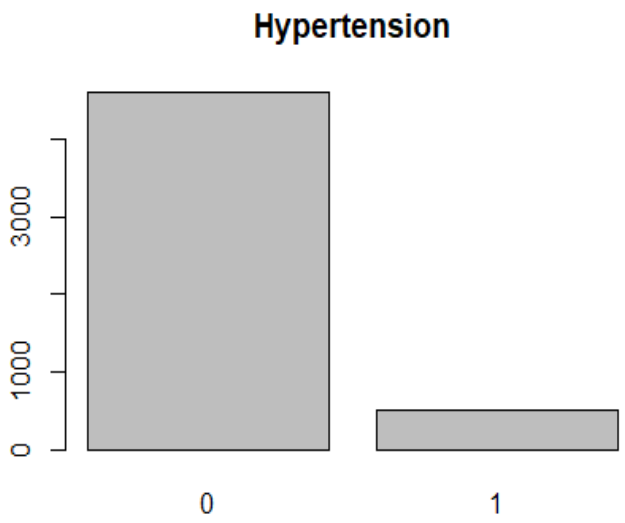

```
##
## Female   Male  Other
##   2994   2115      1
```

Most patients are between 35 and 65 years old. The median age is 45 and the average is 43.



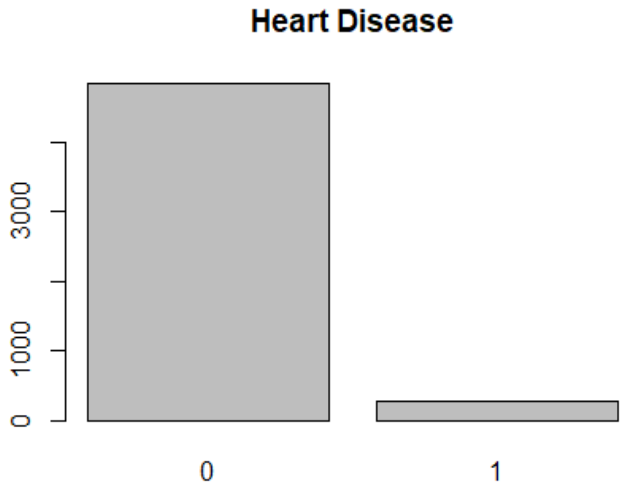


Only 9.75% (498) of patients suffer from hypertension.



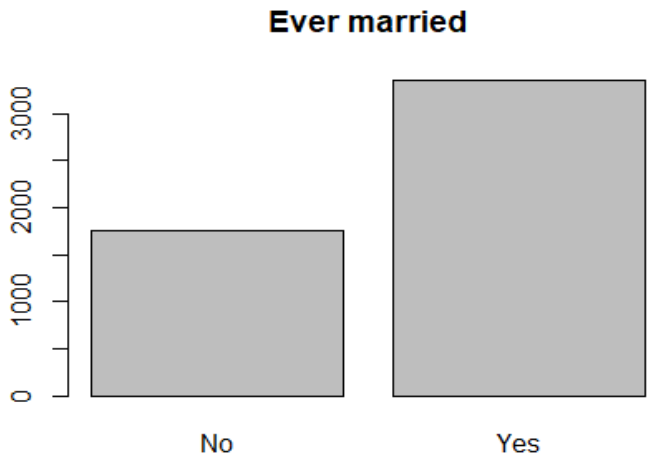

```
##
##    0    1
## 4612  498
```

Similarly, only 5.40% (276) of patients have heart disease.




```
##
##    0    1
## 4834  276
```

Close to a third of the patients (1,757 or 34.38%) have never been married.

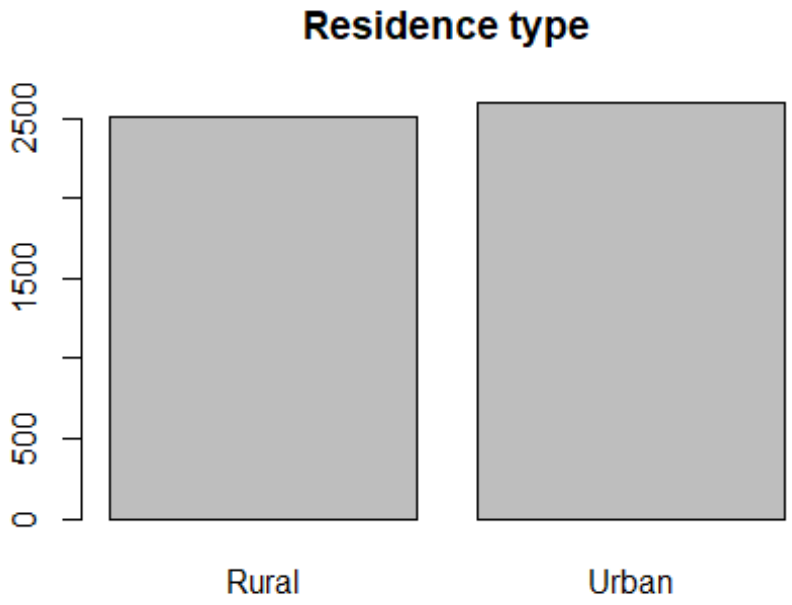



```
##
##   No  Yes
## 1757 3353
```

When looking at work types, those working in the private sector represent 57.24% of patients and that number shoots up to 66.13% when children are taking out of the population as they normally would not work before turning 18.

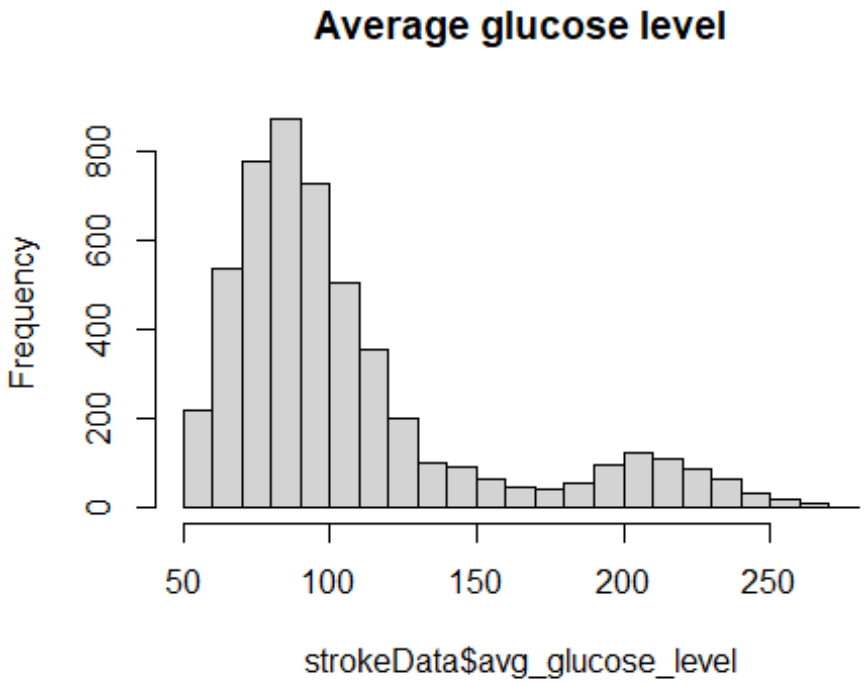

```
##
##      children      Govt_job  Never_worked       Private Self-employed
##           687           657            22          2925           819
```

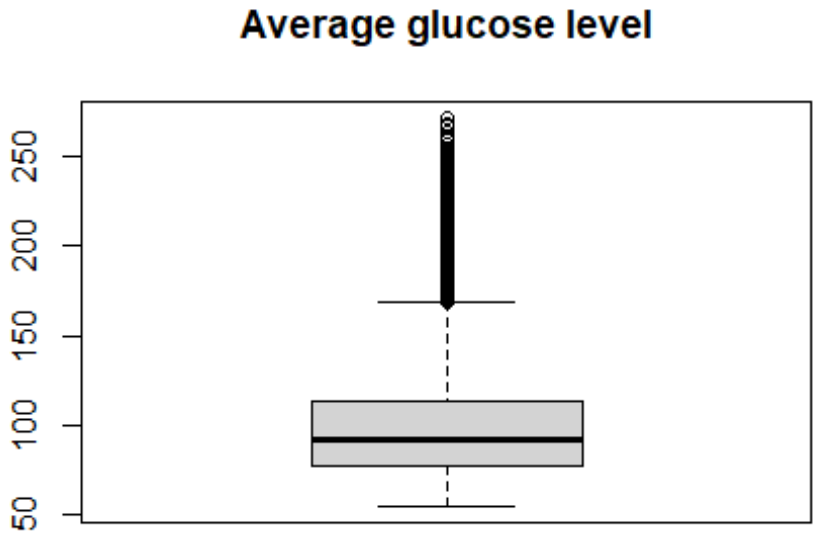
Residence type is the only variable with close to equal proportions (50.8% live in an urban setting to 49.2% who are in rural areas).



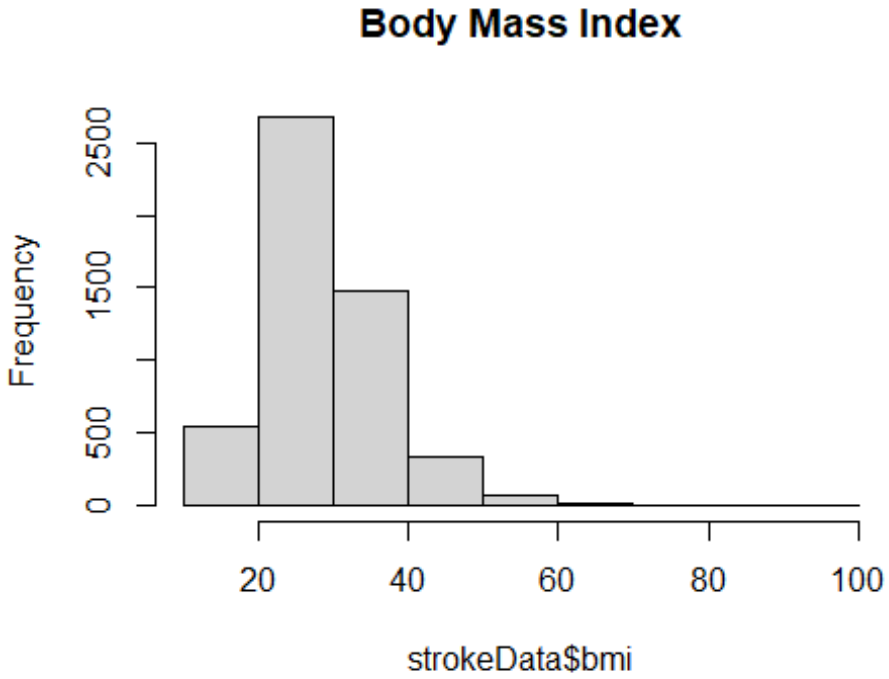

```
##
## Rural Urban
##  2514  2596
```

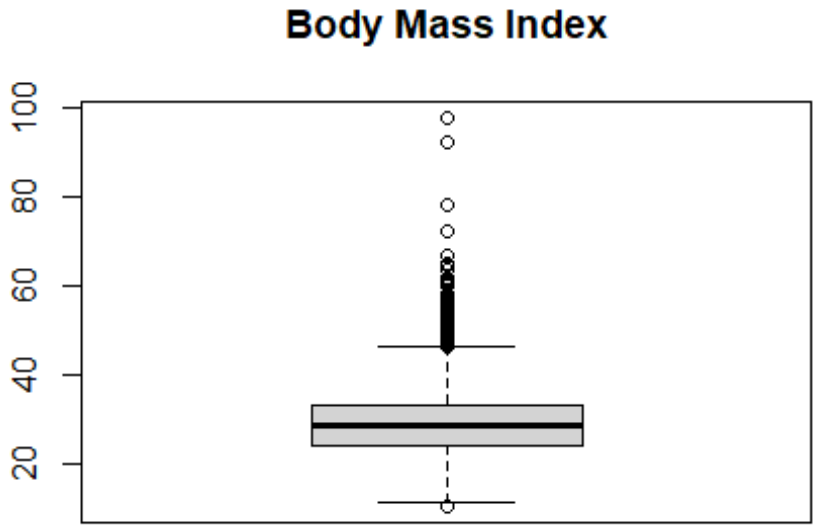
The median value for 'average glucose level' is 92 while the average value is 106. Per Mayo Clinic guidelines, anything less than 140mg/dl is considered normal.



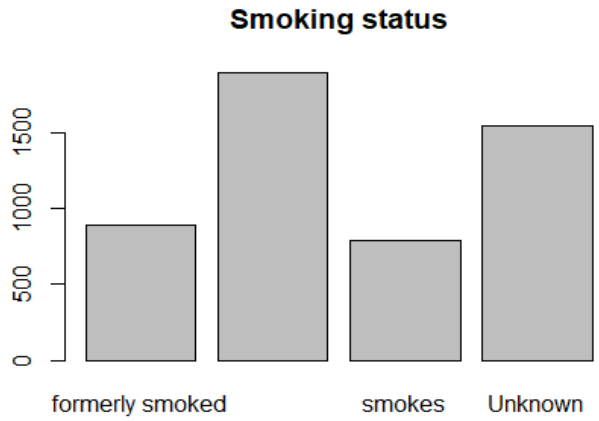


Per the CDC, the average Body Mass Index (BMI) in the US is 26.55 per adult. Our population has an average of 28.9 and an average of 28.4, slightly over the national figures.



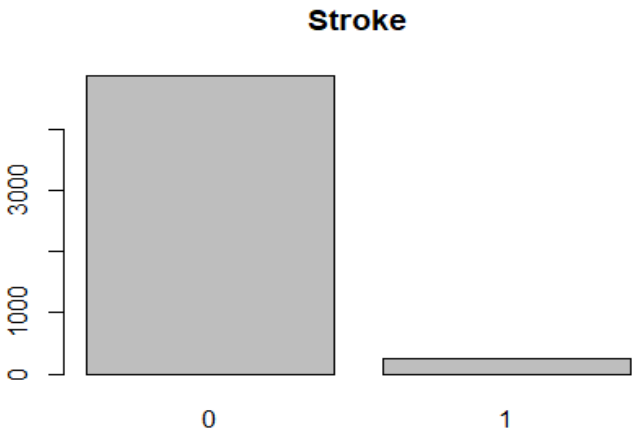


While 1,892 (37%) of the patients have never smoked, the smoking status for another 1,544 (30%) of the patients is unknown, making it difficult to consider smoking as a reliable variable.




```
##
## formerly smoked    never smoked          smokes         Unknown
##             885            1892             789            1544
```

There are 4,861 patients who have not suffered a stroke versus 249 who have. This 95%-5% split will lead to overfitting when building a model to predict stroke in patients. A sampling will be necessary in order to get a more accurate picture.




```
##
##    0    1
## 4861  249
```

Models and analysis

Association Rule Mining (ARM) is an interesting model to use in a medical case as it can help look for antecedents (symptoms) or consequents (diagnosis). In our case, ARM would help identify which pre-existing conditions most often lead to strokes.

Generating rules

Initial try, with support of 0.01, confidence of 0.5 and minimum length of 2, sorted by lift and by confidence

The findings are consistent with what would be expected given the variables: toddlers are considered children, they are not married and do not suffer from hypertension. Nothing groundbreaking in those findings.

```
##     lhs                rhs                        support confidence coverage
lift
## [1] {age=toddlers} => {work_type=children}     0.034   1          0.034
7.4
## [2] {age=toddlers} => {smoking_status=Unknown} 0.034   1          0.034
3.3
## [3] {age=toddlers} => {ever_married=No}        0.034   1          0.034
2.9
## [4] {age=toddlers} => {hypertension=0}         0.034   1          0.034
1.1
## [5] {age=children} => {ever_married=No}        0.056   1          0.056
2.9
##     count
## [1] 173
## [2] 173
## [3] 173
## [4] 173
## [5] 286

##     lhs                                  rhs            support confidence
coverage lift count
## [1] {gender=Male,
##      work_type=children,
##      Residence_type=Urban,
##      avg_glucose_level=(54.1,97.6],
##      bmi=(9.3,27],
##      smoking_status=Unknown}        => {age=children}   0.011       0.52
0.02  9.4    54
## [2] {gender=Male,
##      ever_married=No,
##      work_type=children,
##      Residence_type=Urban,
##      avg_glucose_level=(54.1,97.6],
##      bmi=(9.3,27],
##      smoking_status=Unknown}        => {age=children}   0.011       0.52
```

```
0.02  9.4    54
## [3] {gender=Male,
##      hypertension=0,
##      work_type=children,
##      Residence_type=Urban,
##      avg_glucose_level=(54.1,97.6],
##      bmi=(9.3,27],
##      smoking_status=Unknown}        => {age=children}   0.011       0.52
0.02  9.4    54
## [4] {gender=Male,
##      heart_disease=0,
##      work_type=children,
##      Residence_type=Urban,
##      avg_glucose_level=(54.1,97.6],
##      bmi=(9.3,27],
##      smoking_status=Unknown}        => {age=children}   0.011       0.52
0.02  9.4    54
## [5] {gender=Male,
##      work_type=children,
##      Residence_type=Urban,
##      avg_glucose_level=(54.1,97.6],
##      bmi=(9.3,27],
##      smoking_status=Unknown,
##      stroke=0}                      => {age=children}   0.011       0.52
0.02  9.4    54
```

Focusing on the stroke variable, various scenarios were ran with lhs and rhs being set to stroke=1 or 0

Stroke=1 to lhs

With stroke = 1 as a left-hand variable, the conclusions were as follows: patients who had suffered a stroke were likely to be in their seventies, suffered from hypertension, had above average glucose levels, were self-employed and were former smokers.

```
##     lhs           rhs                                 support confidence
coverage
## [1] {stroke=1} => {age=seventies}                  0.021   0.42
0.049
## [2] {stroke=1} => {hypertension=1}                 0.013   0.27
0.049
## [3] {stroke=1} => {avg_glucose_level=(185,228]}    0.011   0.22
0.049
## [4] {stroke=1} => {work_type=Self-employed}        0.013   0.26
0.049
## [5] {stroke=1} => {smoking_status=formerly smoked} 0.014   0.28
0.049
##     lift count
## [1] 3.6  105
## [2] 2.7   66
## [3] 2.6   55
```

```
## [4] 1.6   65
## [5] 1.6   70
```

Stroke=1 to rhs

With stroke =1 as a right-hand variable, we can see that the leading causes of stroke are an advanced age (seventies), hypertension, high glucose level, be self-employed and be a former smoker.

```
##     lhs                                rhs        support confidence
coverage
## [1] {age=seventies}                 => {stroke=1} 0.021   0.177
0.116
## [2] {hypertension=1}                => {stroke=1} 0.013   0.133
0.097
## [3] {avg_glucose_level=(185,228]}   => {stroke=1} 0.011   0.129
0.083
## [4] {work_type=Self-employed}       => {stroke=1} 0.013   0.079
0.160
## [5] {smoking_status=formerly smoked} => {stroke=1} 0.014   0.079
0.173
##     lift count
## [1] 3.6  105
## [2] 2.7   66
## [3] 2.6   55
## [4] 1.6   65
## [5] 1.6   70
```

Stroke=0 to lhs

With stroke set to 0 on the left-hand side, patients who were likely safe from a stroke had low average glucose levels, did not suffer from either hypertension or heart disease and were likely female.

```
##     lhs           rhs                               support confidence
coverage
## [1] {stroke=0} => {avg_glucose_level=(54.1,97.6]} 0.56    0.59       0.95
## [2] {stroke=0} => {hypertension=0}                0.87    0.91       0.95
## [3] {stroke=0} => {heart_disease=0}               0.91    0.95       0.95
## [4] {stroke=0} => {gender=Female}                 0.56    0.59       0.95
## [5] {}         => {work_type=Private}             0.57    0.57       1.00
##     lift count
## [1] 1    2883
## [2] 1    4429
## [3] 1    4632
## [4] 1    2853
## [5] 1    2925
```

Stroke=0 to rhs

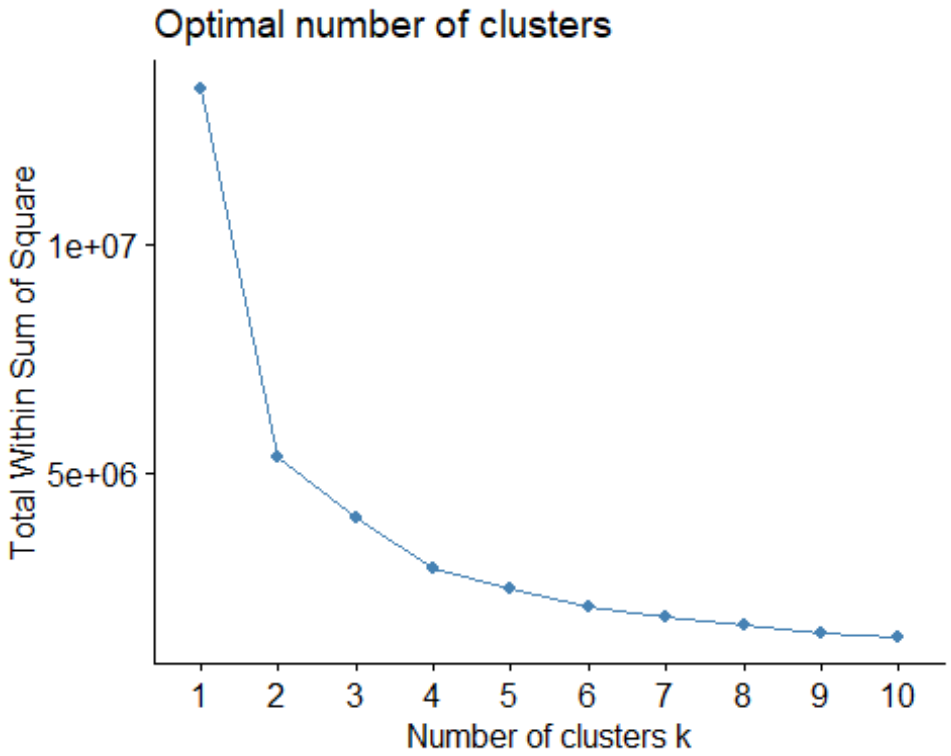
Stroke=0 on the right-hand side indicates that in order to avoid a stroke,one has to not suffer from heart disease or hypertension, have below average glucose level, and be a female.

```
##     lhs                                    rhs          support confidence
coverage lift count
## [1] {hypertension=0,
##      avg_glucose_level=(54.1,97.6]} => {stroke=0}    0.53       0.97
0.54    1  2692
## [2] {heart_disease=0,
##      avg_glucose_level=(54.1,97.6]} => {stroke=0}    0.54       0.97
0.56    1  2781
## [3] {hypertension=0,
##      heart_disease=0}               => {stroke=0}    0.83       0.97
0.86    1  4251
## [4] {avg_glucose_level=(54.1,97.6]} => {stroke=0}    0.56       0.96
0.58    1  2883
## [5] {gender=Female,
##      hypertension=0}                => {stroke=0}    0.51       0.96
0.53    1  2616
## [6] {hypertension=0}                => {stroke=0}    0.87       0.96
0.90    1  4429
```

Association Rule Mining can be a useful tool in medical diagnostics as it can help look for antecedents and prevent the development of a disease in a patient or look for consequents to establish healthy lifestyles and habits in order to avoid any medical complications in patients.

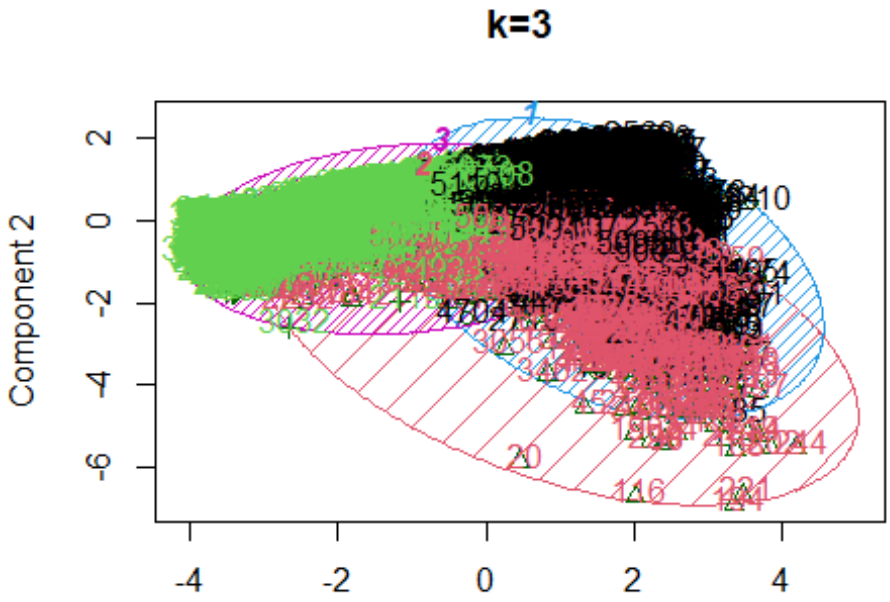
Association Rule Mining is also useful in creating groupings and finding out patterns, many questions could thus be answered using ARM: "are females more likely to have hypertension?", "is residence type associated with heart disease?" or even "is Body Mass Index linked to age?"

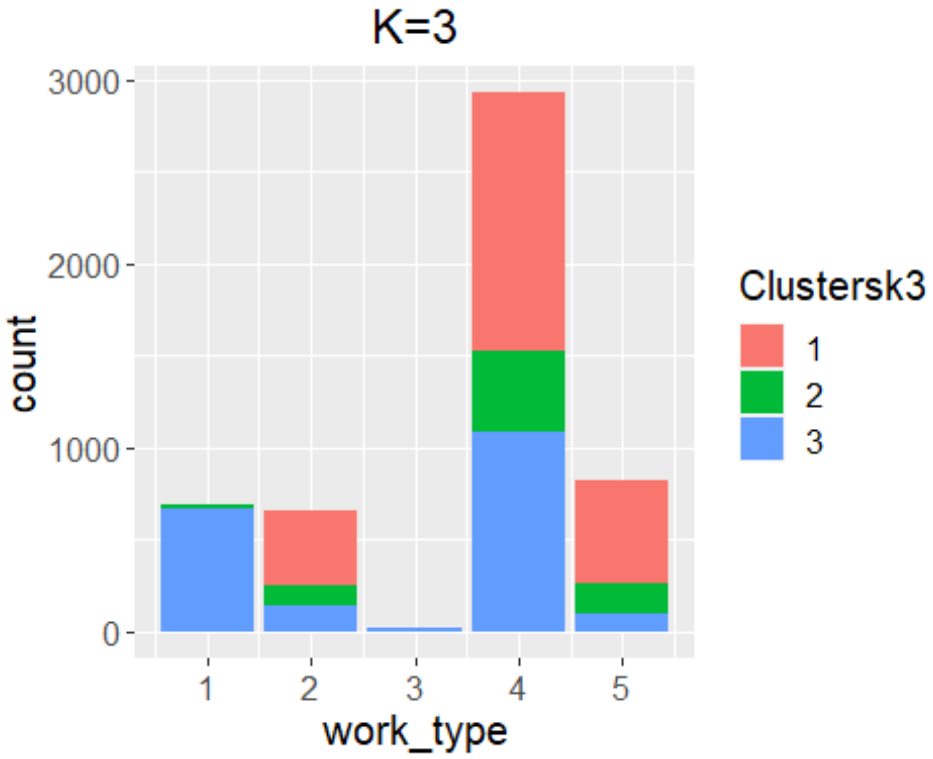
Clustering is another way of processing the data: by dividing it into clusters of similar items to look for similarities. As it is unsupervised, it should provide an insight into the natural groupings found within the data.

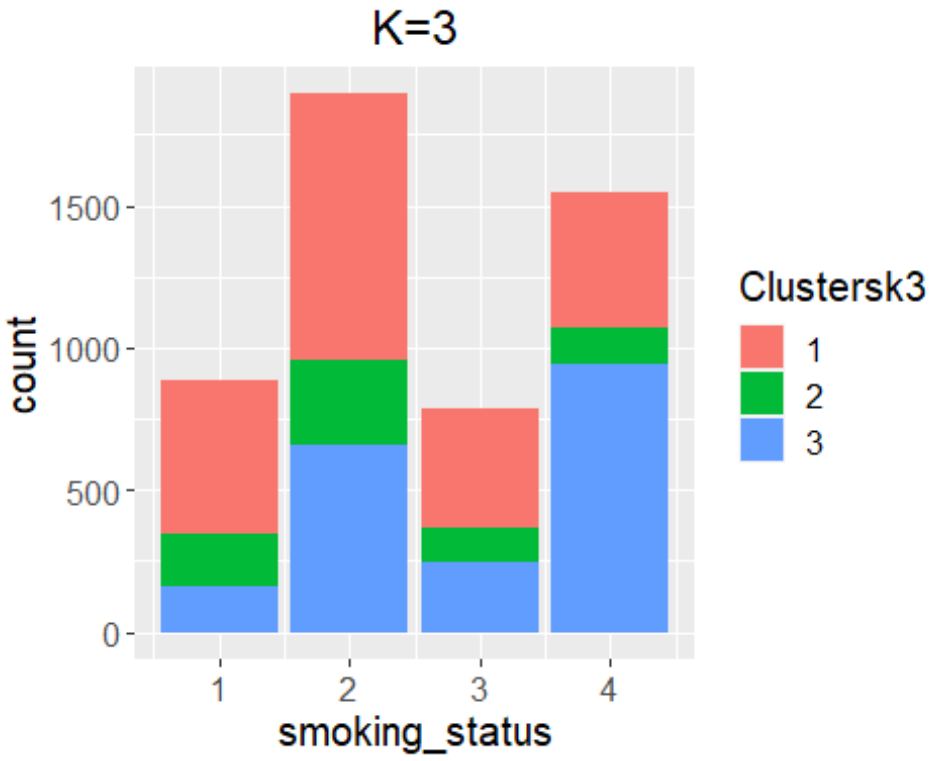


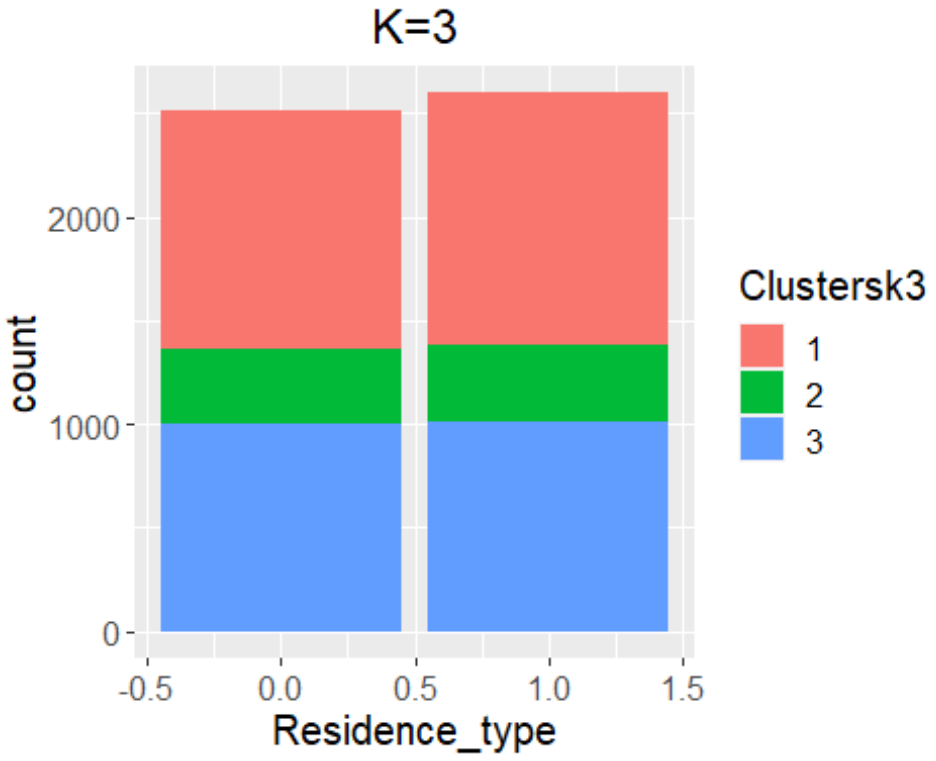
Per the elbow method, 3 was the optimal number of clusters but clustering was also done with 4 and 5 centroids.

```
## List of 9
##  $ cluster     : int [1:5110] 2 2 1 2 2 2 1 1 1 1 ...
##  $ centers     : num [1:3, 1:11] 0.6 0.527 0.591 58.124 58.402 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:3] "1" "2" "3"
##   .. ..$ : chr [1:11] "gender" "age" "hypertension" "heart_disease" ...
##  $ totss       : num 13415256
##  $ withinss    : num [1:3] 1387189 791400 1309702
##  $ tot.withinss: num 3488292
##  $ betweenss   : num 9926964
##  $ size        : int [1:3] 2367 730 2013
##  $ iter        : int 5
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"

##   gender age hypertension heart_disease ever_married work_type
Residence_type
## 1   0.60  58        0.123        0.0710         0.91       3.9
0.51
## 2   0.53  58        0.244        0.1438         0.87       3.9
0.51
## 3   0.59  20        0.014        0.0015         0.29       2.9
0.50
##   avg_glucose_level bmi smoking_status stroke
## 1                89  30            2.3  0.065
## 2               204  32            2.3  0.123
## 3                91  26            3.0  0.003
```


k=3
Component 2
2
0
-2
-4
-6
-4
-2
0
2
4
Component 1
These two components explain 38.74 % of the point variab

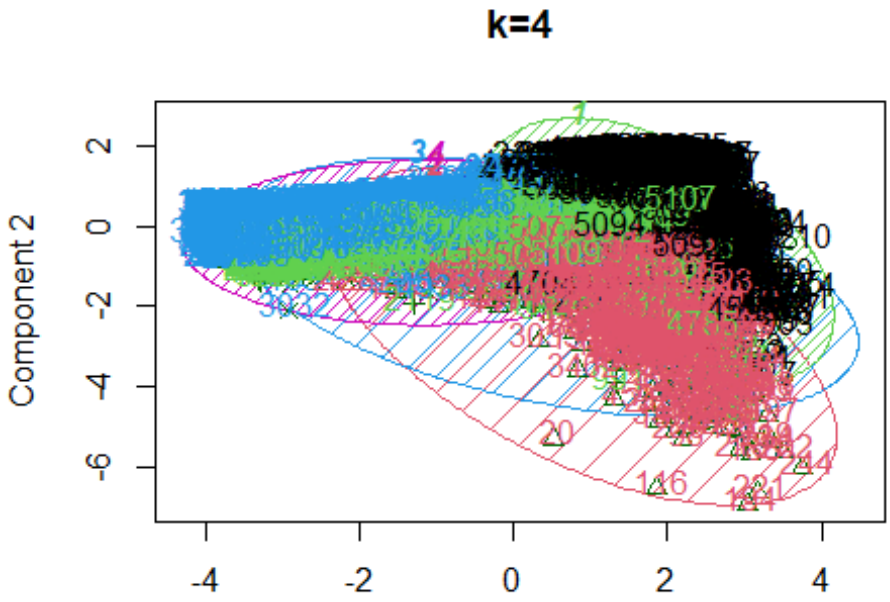

K=3
count
3000
2000
1000
0
1
2
3
4
5
work_type
Clustersk3
1
2
3

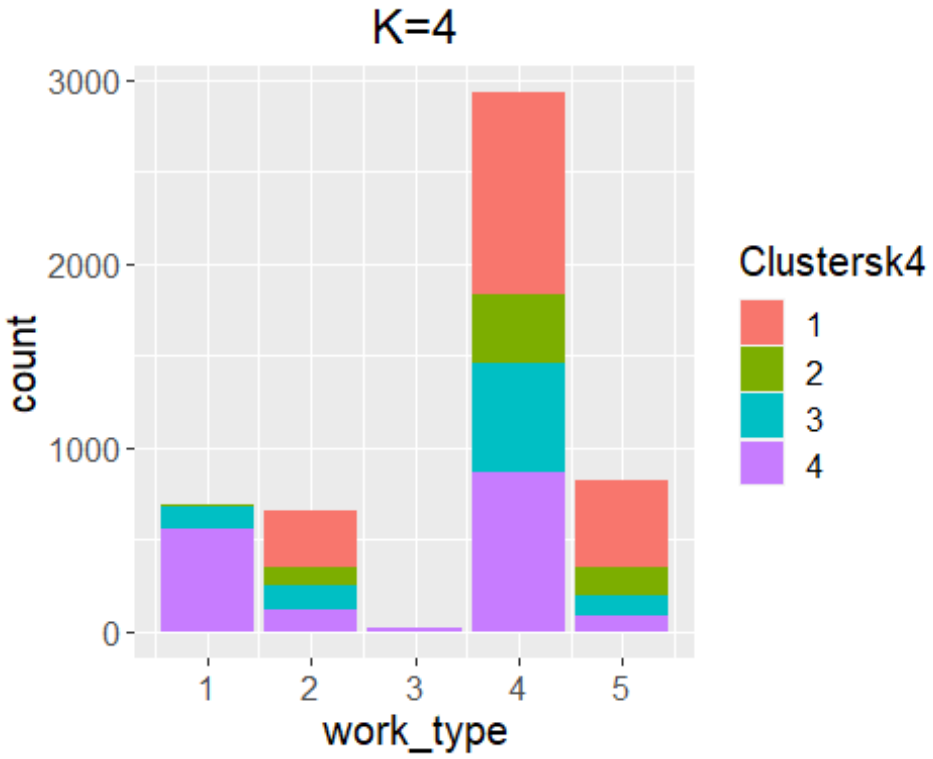


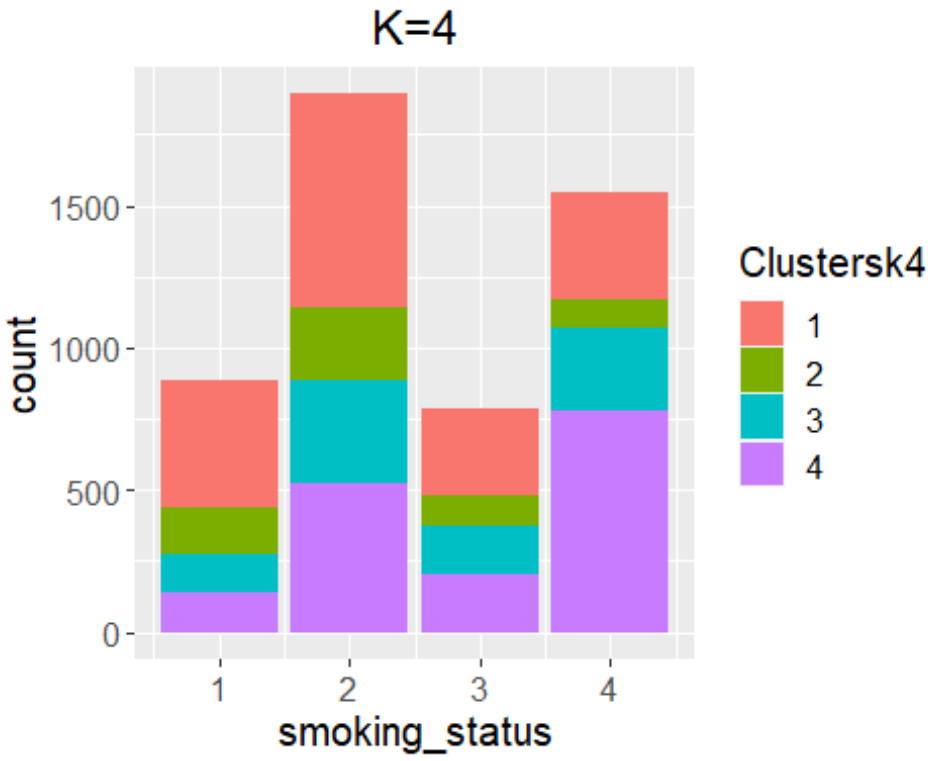


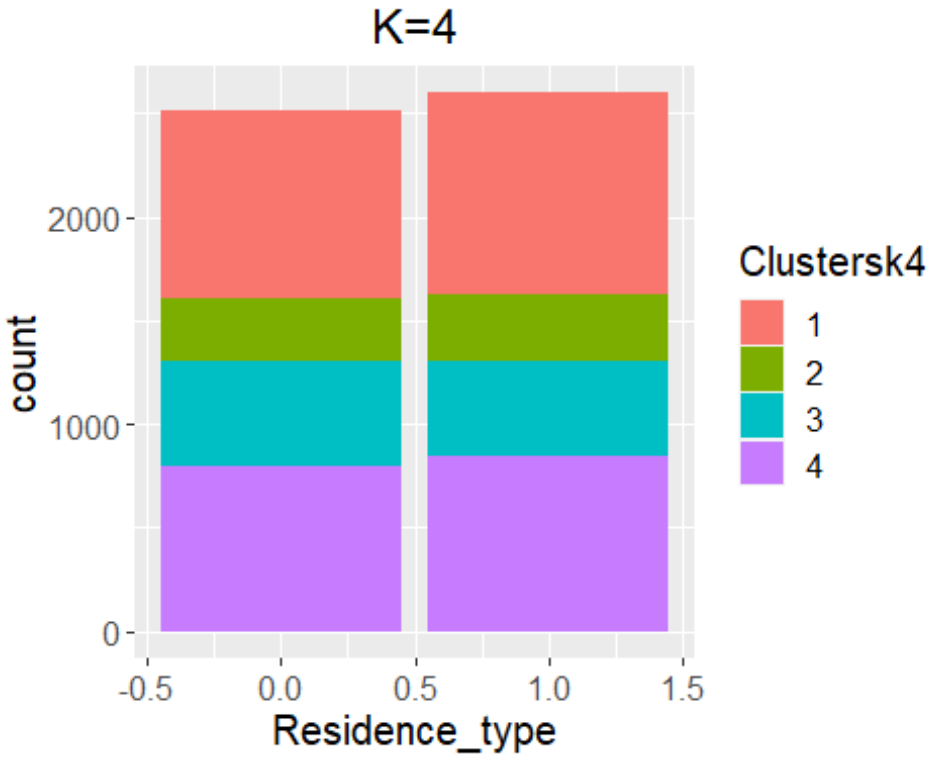

```
## List of 9
##  $ cluster     : int [1:5110] 2 2 1 2 2 2 1 1 1 1 ...
##  $ centers     : num [1:4, 1:12] 0.6 0.524 0.584 0.595 59.886 ...
##   ..- attr(*, "dimnames")=List of 2
```

```
##   .. ..$ : chr [1:4] "1" "2" "3" "4"
##   .. ..$ : chr [1:12] "gender" "age" "hypertension" "heart_disease" ...
##  $ totss       : num 13419611
##  $ withinss    : num [1:4] 742016 476574 667355 655695
##  $ tot.withinss: num 2541639
##  $ betweenss   : num 10877973
##  $ size        : int [1:4] 1876 630 960 1644
##  $ iter        : int 4
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"

##   gender age hypertension heart_disease ever_married work_type
Residence_type
## 1   0.60  60        0.131       0.07623         0.91       3.9
0.52
## 2   0.52  61        0.260       0.16349         0.88       3.9
0.52
## 3   0.58  39        0.066       0.03021         0.64       3.5
0.48
## 4   0.59  20        0.015       0.00061         0.29       2.9
0.51
##   avg_glucose_level bmi smoking_status stroke Clustersk3
## 1                83  30            2.3 0.0682        1.0
## 2               211  33            2.2 0.1365        2.0
## 3               124  29            2.6 0.0292        2.0
## 4                82  26            3.0 0.0043        2.9
```


k=4
Component 2
2
0
-2
-4
-6
-4
-2
0
2
4
Component 1
These two components explain 40.56 % of the point variab

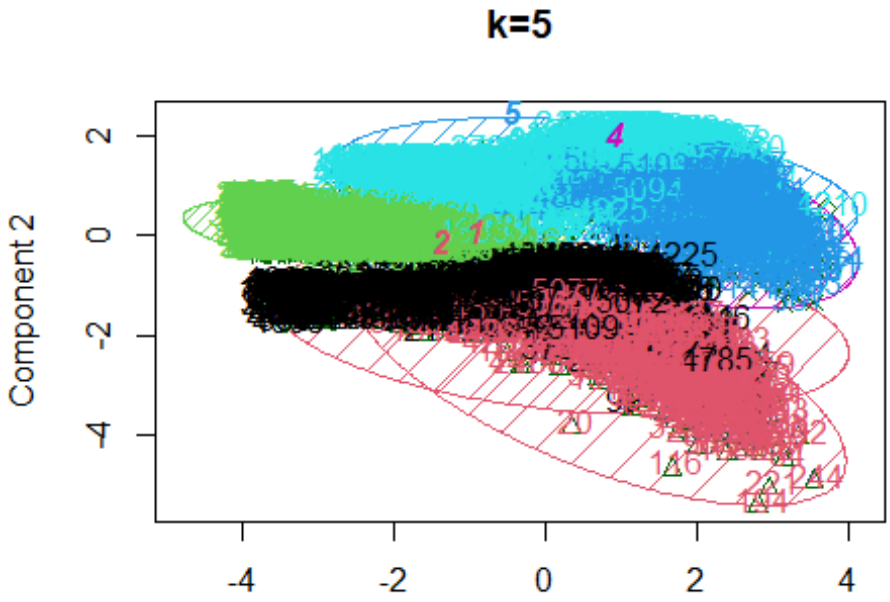

K=4
count
3000
2000
1000
0
1
2
3
4
5
work_type
Clustersk4
1
2
3
4

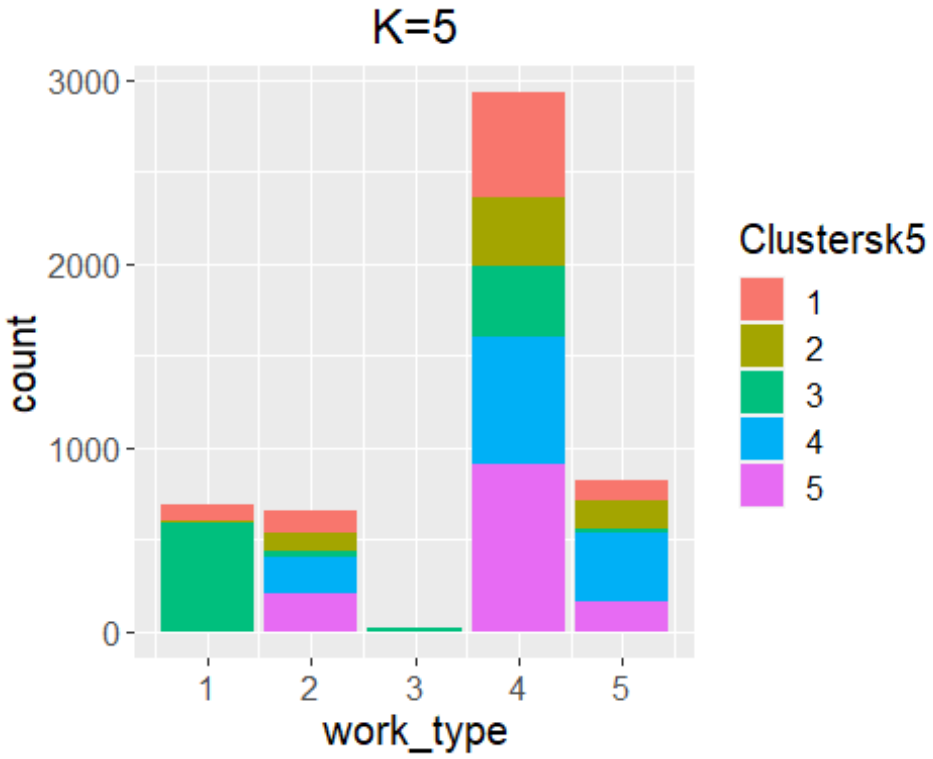


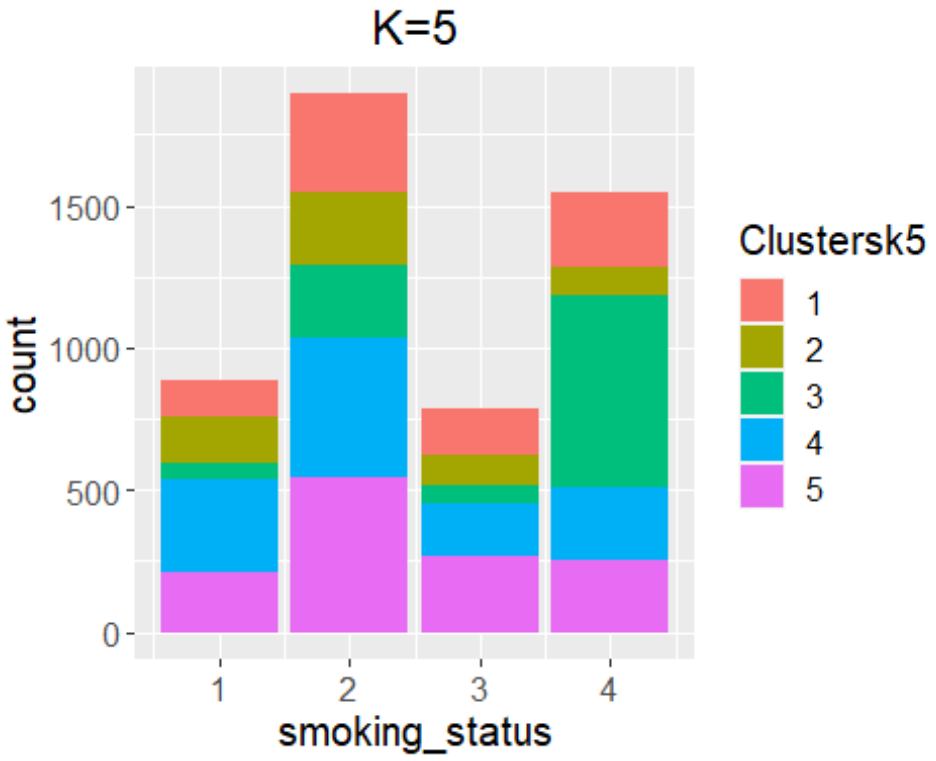


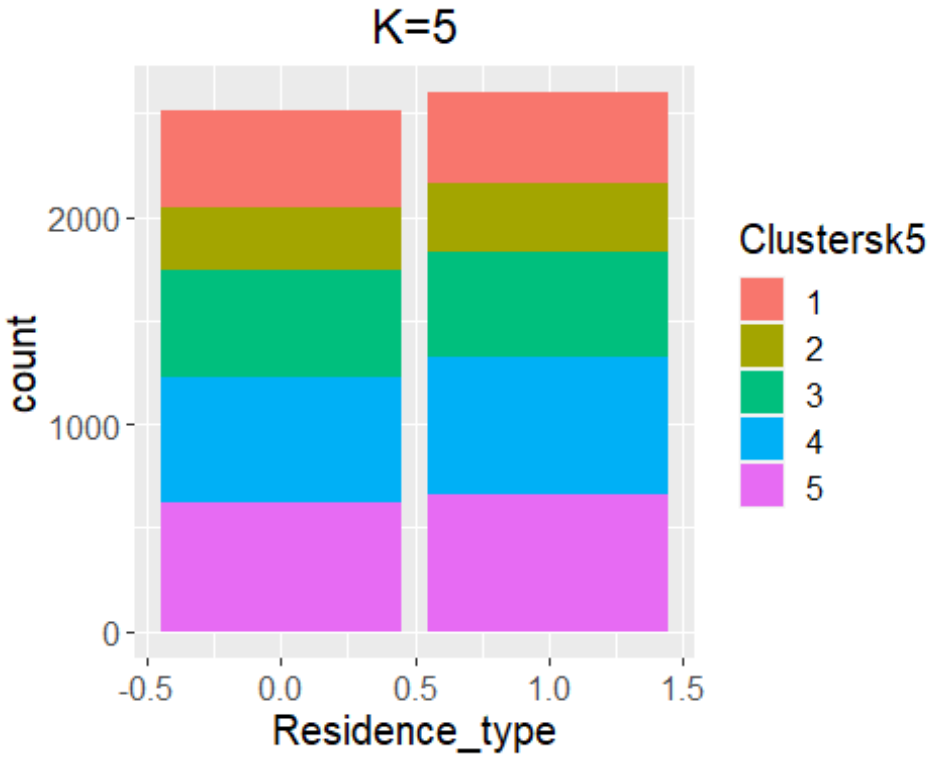

```
## List of 9
##  $ cluster     : int [1:5110] 2 2 4 2 2 2 4 4 4 4 ...
##  $ centers     : num [1:5, 1:13] 0.587 0.524 0.544 0.594 0.642 ...
##   ..- attr(*, "dimnames")=List of 2
```

```
##   .. ..$ : chr [1:5] "1" "2" "3" "4" ...
##   .. ..$ : chr [1:13] "gender" "age" "hypertension" "heart_disease" ...
##  $ totss       : num 13427922
##  $ withinss    : num [1:5] 603005 476574 334199 423003 375426
##  $ tot.withinss: num 2212208
##  $ betweenss   : num 11215714
##  $ size        : int [1:5] 903 630 1038 1261 1278
##  $ iter        : int 5
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"

##   gender age hypertension heart_disease ever_married work_type
Residence_type
## 1   0.59  39       0.0642       0.03101        0.654       3.5
0.48
## 2   0.52  61       0.2603       0.16349        0.884       3.9
0.52
## 3   0.54  13       0.0019       0.00096        0.077       2.2
0.50
## 4   0.59  66       0.1570       0.10071        0.916       4.0
0.52
## 5   0.64  39       0.0595       0.01330        0.759       3.8
0.52
##   avg_glucose_level bmi smoking_status stroke Clustersk3 Clustersk4
## 1               125  29            2.6 0.0299        2.0        3.0
## 2               211  33            2.2 0.1365        2.0        2.0
## 3                87  23            3.3 0.0019        3.0        4.0
## 4                86  29            2.3 0.0904        1.0        1.0
## 5                76  31            2.4 0.0156        1.9        2.5
```


k=5
Component 2
Component 1
These two components explain 41.22 % of the point variab


K=5
count
work_type
Clustersk5
1
2
3
4
5


K=5
count
1500
1000
500
0
1
2
3
4
smoking_status
Clustersk5
1
2
3
4
5


K=5
count
2000
1000
0
-0.5
0.0
0.5
1.0
1.5
Residence_type
Clustersk5
1
2
3
4
5

Classification

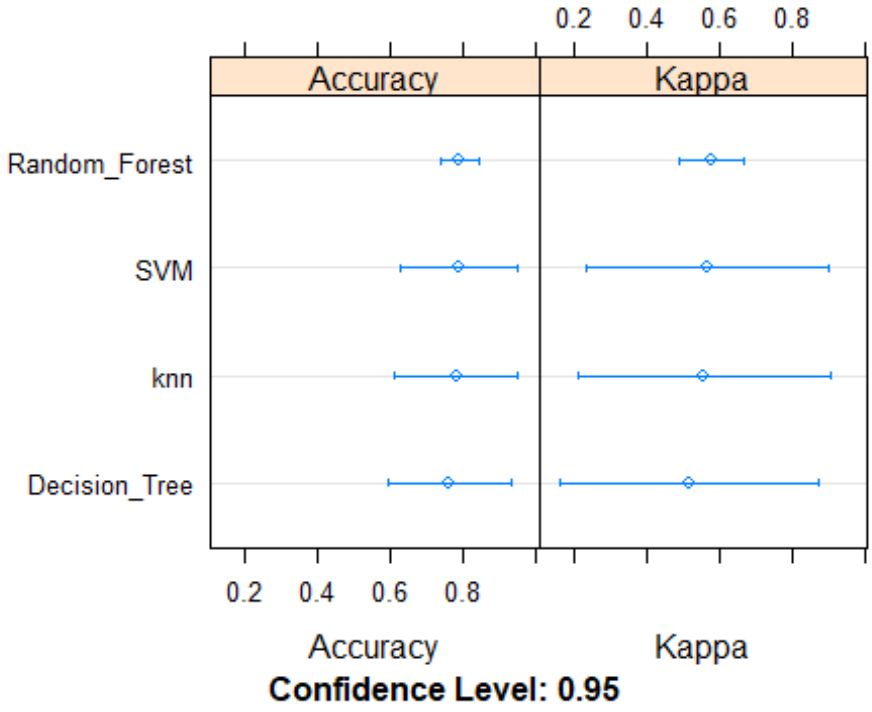
In the classification exercise, due to supervised learning taking place, testing and training datasets will be necessary. As seen previously, the stroke factor (which is the independent variable that is being measured) has an important disproportion with a 95-5 split. In order to manipulate and use classifiers, a sample dataset will need to be generated. As such a random sampling of 300 patients will be selected as the number of stroke cases is 249.

The training dataset was used with various classifier models and the results were processed through a confusion matrix. The results were as follows: decision tree and random forest had the highest accuracy values with 76% and 75% respectively.

The final confusion matrices show very similar results for both models as both proved to be reliable classifying models for this dataset.

The decision tree shows nodes that make sense as stroke and other medical conditions often depend on age and other pre-existing conditions, such as hypertension, body mass index or heart disease. It is therefore not surprising to see the first node as age (<56), followed by glucose level (>=77) as those are variables often encountered in medical cases.

```
##
##    0    1
## 4861    0

##
##   0   1
## 300   0

##
##   0   1
## 300 249
```




```
cm(tree.model)
```

```
## Confusion Matrix and Statistics
##
##     truth
## pred    0    1
##    0 3375   39
##    1 1486  210
##
##                Accuracy : 0.702
##                  95% CI : (0.689, 0.714)
##     No Information Rate : 0.951
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.143
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.694
##             Specificity : 0.843
##          Pos Pred Value : 0.989
##          Neg Pred Value : 0.124
##              Prevalence : 0.951
##          Detection Rate : 0.660
##    Detection Prevalence : 0.668
##       Balanced Accuracy : 0.769
##
```

```
##        'Positive' Class : 0
##

#
cm(svm.model)

## Confusion Matrix and Statistics
##
##     truth
## pred    0    1
##    0 3430   52
##    1 1431  197
##
##                Accuracy : 0.71
##                  95% CI : (0.697, 0.722)
##     No Information Rate : 0.951
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.137
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.706
##             Specificity : 0.791
##          Pos Pred Value : 0.985
##          Neg Pred Value : 0.121
##              Prevalence : 0.951
##          Detection Rate : 0.671
##    Detection Prevalence : 0.681
##       Balanced Accuracy : 0.748
##
##        'Positive' Class : 0
##

#
cm(rf.model)

## Confusion Matrix and Statistics
##
##     truth
## pred    0    1
##    0 3614   41
##    1 1247  208
##
##                Accuracy : 0.748
##                  95% CI : (0.736, 0.76)
##     No Information Rate : 0.951
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.176
##
```

```
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.743
##             Specificity : 0.835
##          Pos Pred Value : 0.989
##          Neg Pred Value : 0.143
##              Prevalence : 0.951
##          Detection Rate : 0.707
##    Detection Prevalence : 0.715
##       Balanced Accuracy : 0.789
##
##        'Positive' Class : 0
##

#
cm(knn.model)

## Confusion Matrix and Statistics
##
##     truth
## pred    0    1
##    0 3454   52
##    1 1407  197
##
##                Accuracy : 0.714
##                  95% CI : (0.702, 0.727)
##     No Information Rate : 0.951
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.14
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.711
##             Specificity : 0.791
##          Pos Pred Value : 0.985
##          Neg Pred Value : 0.123
##              Prevalence : 0.951
##          Detection Rate : 0.676
##    Detection Prevalence : 0.686
##       Balanced Accuracy : 0.751
##
##        'Positive' Class : 0
##

#
print(tree.model)

## CART
##
## 219 samples
```

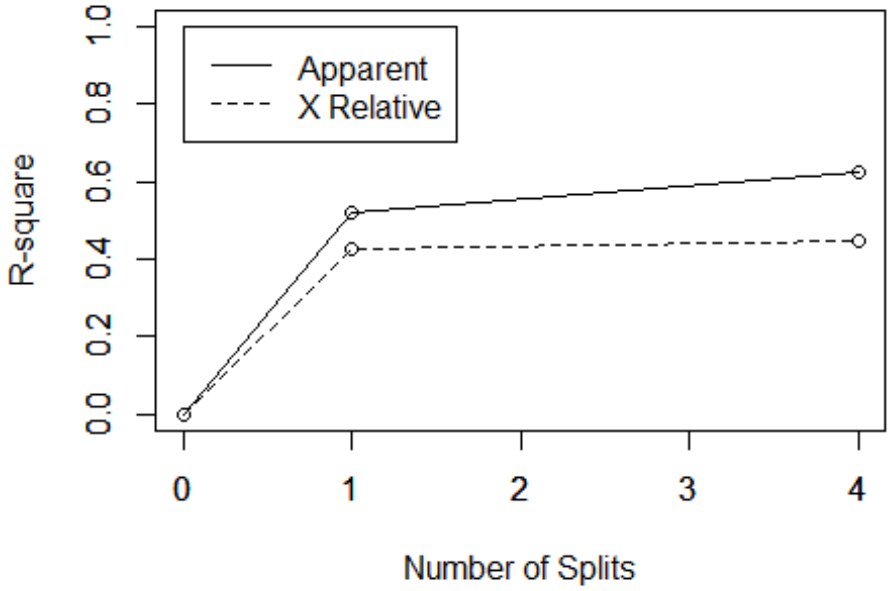
```
##  10 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 146, 146, 146
## Resampling results across tuning parameters:
##
##   cp      Accuracy  Kappa
##   0.0000  0.75      0.49
##   0.0052  0.75      0.49
##   0.0312  0.76      0.52
##   0.0347  0.76      0.52
##   0.5208  0.68      0.33
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.035.

#
print(svm.model)

## Support Vector Machines with Radial Basis Function Kernel
##
## 219 samples
##  10 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 146, 146, 146
## Resampling results across tuning parameters:
##
##   C     Accuracy  Kappa
##   0.25  0.78      0.55
##   0.50  0.79      0.57
##   1.00  0.79      0.57
##   2.00  0.77      0.53
##   4.00  0.76      0.51
##
## Tuning parameter 'sigma' was held constant at a value of 0.0017
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were sigma = 0.0017 and C = 0.5.

#
print(rf.model)

## Random Forest
##
## 219 samples
##  10 predictor
##   2 classes: '0', '1'
```

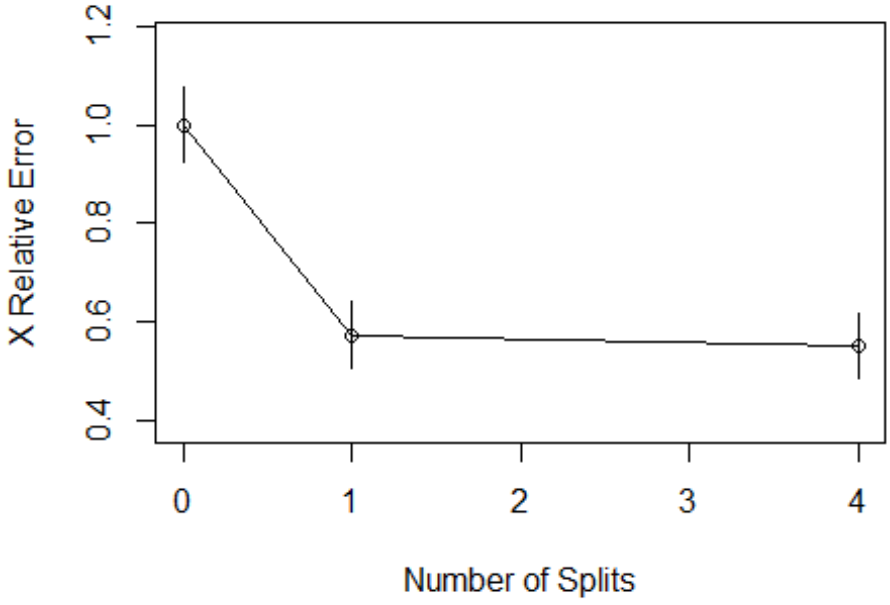
```
##
## No pre-processing
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 146, 146, 146
## Resampling results across tuning parameters:
##
##   mtry  Accuracy  Kappa
##    2    0.79      0.56
##    5    0.79      0.58
##    9    0.78      0.55
##   12    0.77      0.54
##   16    0.78      0.55
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 5.

#
print(knn.model)

## k-Nearest Neighbors
##
## 219 samples
##  10 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold)
## Summary of sample sizes: 146, 146, 146
## Resampling results across tuning parameters:
##
##   k   Accuracy  Kappa
##    5  0.78      0.56
##    7  0.78      0.56
##    9  0.78      0.56
##   11  0.77      0.53
##   13  0.77      0.54
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 9.
```

Creating a decision tree

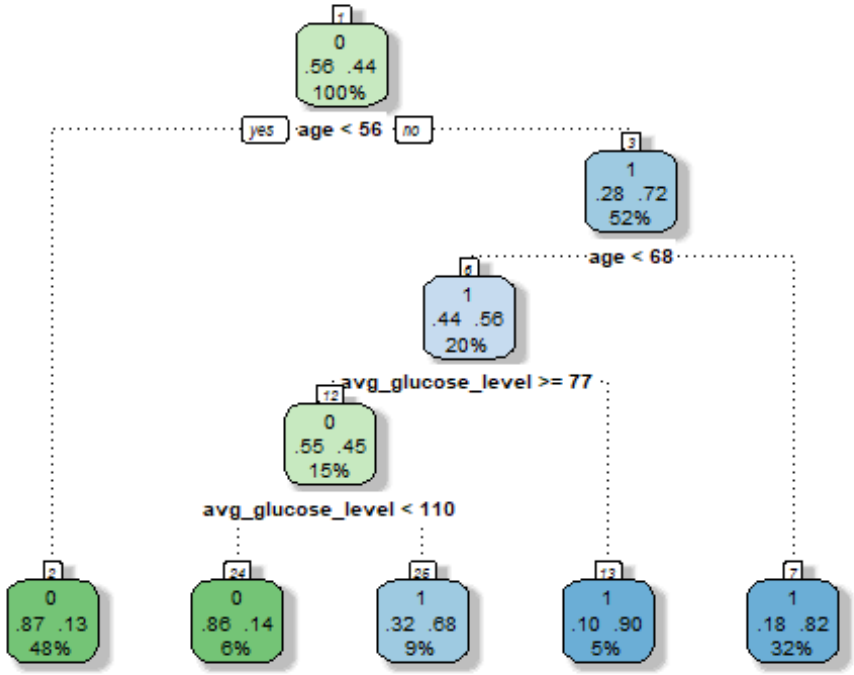
```
# Tree 1
DT1<-rpart(stroke~., data = train, method = "class", control =
rpart.control(cp=0.0347))

# Predicting the test dataset and plotting splits and the decision tree
prediction1= predict(DT1, test, type="class")
rsq.rpart(DT1)
```

```
##
## Classification tree:
## rpart(formula = stroke ~ ., data = train, method = "class", control =
rpart.control(cp = 0.0347))
##
## Variables actually used in tree construction:
## [1] age               avg_glucose_level
##
## Root node error: 96/219 = 0
##
## n= 219
##
##   CP nsplit rel error xerror xstd
## 1  1      0         1      1    0
## 2  0      1         0      1    0
## 3  0      4         0      1    0
```






```
fancyRpartPlot(DT1)
```




```
# Making a confusion matrix for correct/incorrect predictions
table(stroke=prediction1, true=test$stroke)

##       true
## stroke   0   1
##      0 137  38
##      1  40 115
```

Creating a random forest

```
# RF
RF<-randomForest(stroke~.,data = train,mtry=5)
predictrf=predict(RF, test, type="class")
# making a confusion matrix for correct/incorrect predictions
table(stroke=predictrf, true=test$stroke)

##       true
## stroke   0   1
##      0 130  42
##      1  47 111
```

```
```