Stroke prediction report

Figure 1: visualization of a stroke (MCOR, 2020)

Name: Pascal Visser

Class: BFV3

Student number: 410729

Course: Bio-informatica/Theme 09

Date: 13-11-2022

Contents

1. Introduction

At some point in a person's life, the word stroke becomes relevant. Everyone knows someone with family members or friends who have been affected by a stroke. But what is a stroke exactly?

A stroke is a cardiovascular disease that affects the brain, by preventing it from getting oxygen and a proper blood flow. A person with a stroke has a problem with arteries to and within the brain. These arteries can either be clogged or broken/leaking. This will result in braincells not getting enough oxygen and nutrients, so the brain cells die. The dying of brain cells results in damage to the brain, and it won't function the same. A stroke will result in permanent brain damage, If the stroke hits the sensitive parts of the brain, (like moving and sensing) the result can be for the person to become paralyzed. In worse cases strokes can lead to death. (American stroke association, 2022)

There are two types of strokes: Ischemic and Hemorrhagic

Ischemic stroke:

An ischemic stroke is most common type of stroke, it happens when a blood vessel is clogged or blocked. Deposits of fatty substances called plaque can build up in blood vessels leading to the brain. When the blood vessels get thinner into the brain, they get clogged up and block the blood flow leading to a stroke.

Hemorrhagic stroke:

A Hemorrhagic stroke is less common than an Ischemic stroke. It happens when a blood vessel in the brain leaks or ruptures. The leaking blood puts damage onto the brain cells, which is cause them to die. High blood pressure is the most leading cause for hemorrhagic stroke symptoms.

There is also something called a TIA, which is in fact a mini stroke. It is a type of ischemic stroke, but temporally. A blood vessel can be clogged for a short period of time and then unclog itself by blood flow/pressure. A TIA can be an important warning for a future stroke. (CDC, 2022)

It depends which region of the brain was hit with a stroke to determine the outcome. Common problems people with stroke history experience are: Paralysis, Numbness, muscle movements problems , trouble thinking, learning, or making new memories. People can (partly) recover, but it takes time to rehabilitate. (CDC, 2022)

Strokes can be prevented by living a healthy lifestyle, but the changes are never zero. Hearth disease, high blood pressure, atrial fibrillation, high cholesterol, and diabetes are all cause of stroke. (CDC, 2022)

Globally 1 in 4 adults will have a stroke in their lifetime, over 110 million in the world have experienced strokes. Yearly 6.5 million people die as result for a stroke. (WSO, 2022). It is the second leading cause of dead worldwide. (WHO, 2020)

Because of the commonness of strokes and their risks, it is important to accurate give people diagnoses about their condition or risk levels of strokes. This research focuses on achieving this.

With a dataset and machine learning, it creates the following research question:

Is it possible to produce an accurate algorithm with machine learning, to calculate the risk of a stroke based on lifestyle variables of people with stroke history and people that never had a stroke?

2. Materials & methods

This section is about the used materials and methods, materials include datasets and used software, including version and parameters. The methods are the way how the materials are used, what was done and how it was done. It is arranged in a chronological order. The workflow consists of an Exploratory Data Analysis (EDA) follow up by data cleaning for machine learning, after that a Wrapper is made to execute the machine learning model as a command line application.

2.1 Materials

In the introduction was said how severe a stroke can be, and how hard the effects could be if the person survives. There are many factors that can contribute towards a higher chance of a stroke, the so called stroke chance. The goal of this research is to predict the risk of a stroke. To achieve this a set of data is needed. On Kaggle.com a dataset was found with 11 clinical features for predicting stroke events, and a column to indicate if this person had a stroke or not.

2.2.1 The data

The data consists of 5110 persons with their clinical features, these clinical features can be split into health and lifestyle. Examples of health variables are BMI, glucose levels, hypertension and if the person has a heart disease or not. The lifestyle variables are: residence type, smoking, married or type of work. The idea behind the lifestyle variables is that they can contribute towards stress, which can lead to high blood pressure. Which in that case increases the chance of hemorrhagic Strokes.

	id <int>	gender <chr>	age <dbl>	hypertension <int>	heart_disease <int>	ever_married <chr>	work_type <chr>	Residence_type <chr>	avg_glucose_level <dbl>
1	9046	Male	67	0	1	Yes	Private	Urban	228.69
2	51676	Female	61	0	0	Yes	Self-employed	Rural	202.21
3	31112	Male	80	0	1	Yes	Private	Rural	105.92
4	60182	Female	49	0	0	Yes	Private	Urban	171.23
5	1665	Female	79	1	0	Yes	Self-employed	Rural	174.12
6	56669	Male	81	0	0	Yes	Private	Urban	186.21
7	53882	Male	74	1	1	Yes	Private	Rural	70.09
8	10434	Female	69	0	0	No	Private	Urban	94.39

Figure 2: Example of the data

In figure 2, a part of the data is seen. Every person has their own records in the dataset. This dataset is the core of the research. It is used in the EDA and the machine learning (ML). The origin of the data is unfortunately unknown because the source is confidential.

Link to dataset: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset

2.2.2 Software

In this research, several programs and two programming languages are used. There are three programs that are included in this research: RStudio, Weka and IntelliJ. And there are two programming languages exploited: R and Java.

RStudio:

RStudio is an IDE for using the statistical programming language R. In RStudio the research log, the EDA is done, reviewed, and discussed. Also, the Machine learning part of the project is done in the log.

The used version of RStudio is 2022.07.01 build 554.

The used version of R is 4.1.2

There are also several packages used in the research, these are:

Table 1: Table of used packages

Package name	Used version number
tidyverse	1.3.2
naniar	0.6.1
readr	2.3.1
ggplot2	3.3.6
knitr	1.40
pander	0.6.5
dplyr	1.0.10
DMwR**	0.4.1
gridExtra	2.3
farff	1.1.1
cowplot	1.1.1

**The package 'DMwR' was removed from the CRAN repository trough several problems. The package can be retrieved by instruction in the log.

IntelliJ:

IntelliJ is the IDE where the wrapper for the machine learning model is made. The programming language used to make the wrapper is java. In the IDE the wrapper is constructed with the machine learning model.

The used version of IntelliJ is IntelliJ IDEA 2021.2.1 (Ultimate edition).

The used Java version is 18.0.2.1.

Weka:

Weka is a collection of machine learning algorithms for data mining tasks. It contains tools for data preparation, classification, regression, clustering, association rules mining, and visualization.

The used version of Weka is 3.9.6.

All these tools are a collection of materials, materials to efficiently research the topic and get fast and accurate results. Note that the speed and usage of these tools is also dependent on the system it is used on. By this mainly the speed is sensitive.

2.2 Methods

Now the materials are known, it is time to talk about the method. Method is this case is how are the materials used, and what was done. The research can be divided into a couple of parts:

- Data preparation
- EDA
- Data cleaning
- Machine learning

All the steps below have taken places into RStudio and Weka

Data preparation

The data preparation is about the first look at the data, first the necessary libraries were loaded in, after that the data was loaded into RStudio and viewed. Following up, a codebook was constructed to clearly explain what all the variables mean and what datatype they are. Also, all the unique values of categorical variables were shown.

By viewing the original data, a few issues came to rise. The variables age and BMI where in the wrong datatype. Age, a whole number, was in the double datatype. This was casted to a integer. BMI was a character datatype, which doesn't make sense. This was cast to double. Other variables were cast to factor to make it more workable.

After casting the types, a check on missing values was performed. This raised a few missing values in the BMI columns, those were simple replaced by the column mean. A more serious issue was the smoking status column. Where there was looked for unique values at categorical attributes, smoking status showed four: never, formally, smokes and unknown. Unknown is basically a missing value but categorical. This unknown value was replaced with other values in the column, based on their occurrence in the column.

Exploratory Data Analysis

The Exploratory Data Analysis (EDA) is about learning and understanding the data, finding outliers and patterns. Discover underlying correlations and making connections between variables. The first part of the EDA focuses on the individual variables and their own distribution, How is the gender and age ratio? Is the majority smoking or not? Are there many people with underlying cardiovascular diseases? The first part looks at those questions.

The second part focuses more on the variables in relation to the classifier stroke. If there are many people with a high BMI, how many of them have stroke history? Cardiovascular diseases are related to stroke, but how many people with these diseases had strokes? Are old people more sensitive to stroke? Or are people with a busy lifestyle more prone to strokes? The second part answers all these kinds of questions.

Data cleaning

The data cleaning is the preparation of the machine learning, the data need to be prepped to be suitable for Weka. In this dataset, the biggest issue was an imbalance in the classifier, which, when unattended, will result in a skewed result. The accuracy will be high, but the reason why will not be good. To tackle this imbalance, four techniques are explored and weighted. One of the four techniques was applied on the dataset to tackle the imbalance. To apply the technique, certain

variables needed to be converted to factors, this was because the technique don't work well with numeric values. So, most variables like yes and no are converted to 1 and 0. This conversion was necessary to successfully use the technique.

Machine learning

There are two datasets now, the original with the imbalance and the 'fixed' dataset. The machine learning part takes place in Weka. Weka self has a couple of filters. One of these filters is classBalances, which automatically fixes an imbalance in the classifier. This filter on the original was also tested. First the three datasets were tested on some machine learning algorithms. The important factors where accuracy, speed, and the ratio of True positives, False positives, True negatives, and False negatives. The outcome of this was a dataset that was good enough and suitable for the 'real' machine learning.

At the start, the quality metrics were determined. These metrics were descriptions of what is meant by a good machine learning model. Firstly, the base algorithm ZeroR was tested. Just to determine the baseline of the model. Everything worse that the baseline is considered bad. After that, more advanced algorithms where tested. These included popular algorithms and some more specific algorithms. These algorithms where also tested on accuracy, speed, and classifying performance. An example of an algorithm test is below in table 2

Table 2: Example of algorithm comparison

			Advanced algorithms				
Algorithms ->	**OneR**	**J48**	**RandomForest**	**Naive.Bayes**	**Simple.logistic**	**Logistic**	**SMO**
Accuracy (%)	97,7	90,2	95,6	80,1	81,7	81,8	82
Speed (sec)	0.02	0.43	3.6	0.04	1.14	0.2	18.5
T TP (%)	100	85,2	93,1	77,4	77,1	77,5	78,3
T FP (%)	3,9	6,2	2,5	18	15	15,2	15,3
F TP (%)	96,1	93,8	97,5	82	85	84,8	84,7
F FP (%)	0	14,8	6,9	22,6	22,9	22,5	21,7
Size of tree	.	925	.	.	.	.	.

Finally, some ensemble learners were also tested against the dataset, The main purpose of these learners is using an ensemble model to use a group of weak learners and form a strong learner.

With the tests done some algorithms looked good but needed optimalization. The focus here was to make the results less overfitted and more reliable. With a trough analysis of the selected algorithms, one was selected to build a model of. The last part was a verdict of why this algorithm is selected above the other. With some comparison visualization and a ROC-curve, to support the choice for that algorithm.

JavaWrapper

The model was selected and save in a file (.model format). This model than was used in a wrapper. A command line application to classify new, unknown instances of a dataset. The goal is to make this wrapper easy to use and efficient for local usage.

Link to research log:
https://github.com/PascalVisser/Thema_09_MLAnalysis/tree/main/EDA%20%26%20ML%20log

Link to javaWrapper:
https://github.com/PascalVisser/Thema_09_MLAnalysis/tree/main/JavaWrapper

3. Results

In the martials and methods section it became clear what was used and how it was done. This workflow also resulted into results. The results section can be divided into Exploratory Data Analysis and machine learning.

3.1 Exploratory Data Analysis

This research focuses on the correctly classifying stroke risks of patients. So, it is important to know the training data well. The EDA Focus on highlighting patterns of variables and finding abnormalities or correlations. Roughly there are two parts of the EDA, variable distribution, and relation to classifier.

One of the most important variables is stroke, because this is the variable that is used as classifier. In figure 3 the distribution of the stroke variable is shown

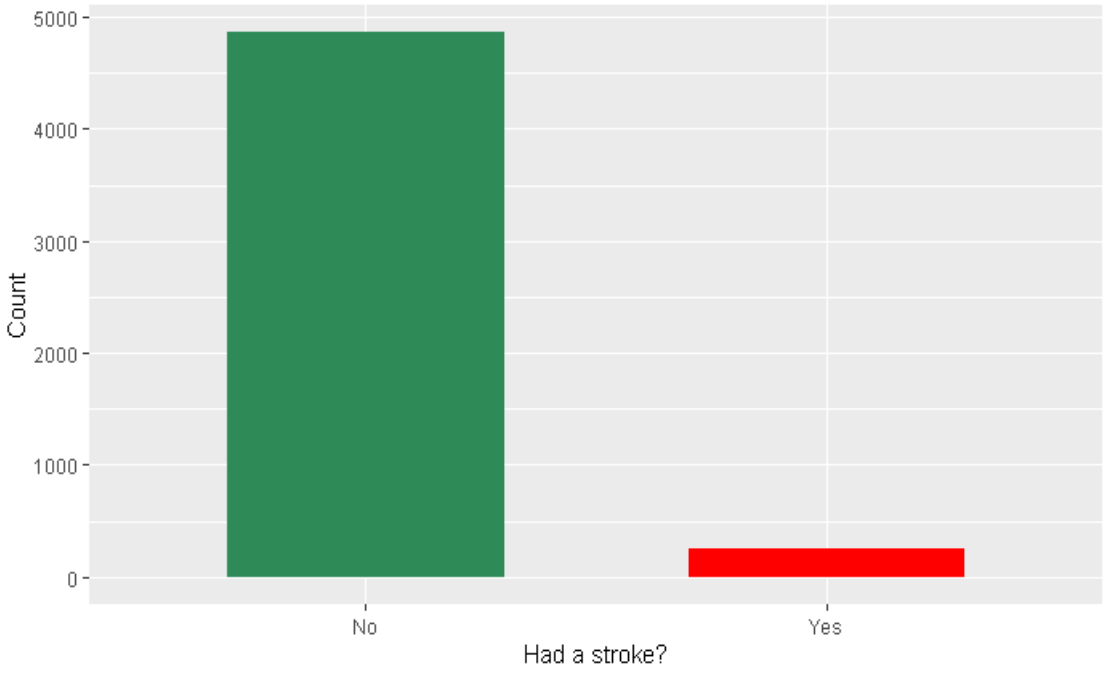

Figure 3: Stroke distribution

This is where the first problem begun to rise. The classifier is serious imbalanced. The group that had a stroke is around 5%, which is a problem with machine learning. An algorithm will tent towards the highest accuracy. By saying that everybody did not have a stroke, it is right for 95%. So, it is easy to get a high accuracy by choosing always false. This imbalance will shift the focus off other variables. Later in the data prep for machine learning this is fixed.

In the introduction is said that a bad health increases the chance of strokes. A couple of variables in the dataset: Hypertension, heart disease, BMI, glucose, and smoking can be considered health variables.

Age and gender

The first columns included age and gender, cause this dataset is about stroke, it's important to know the ages and gender distribution. Figure 4 shows the ages and genders.

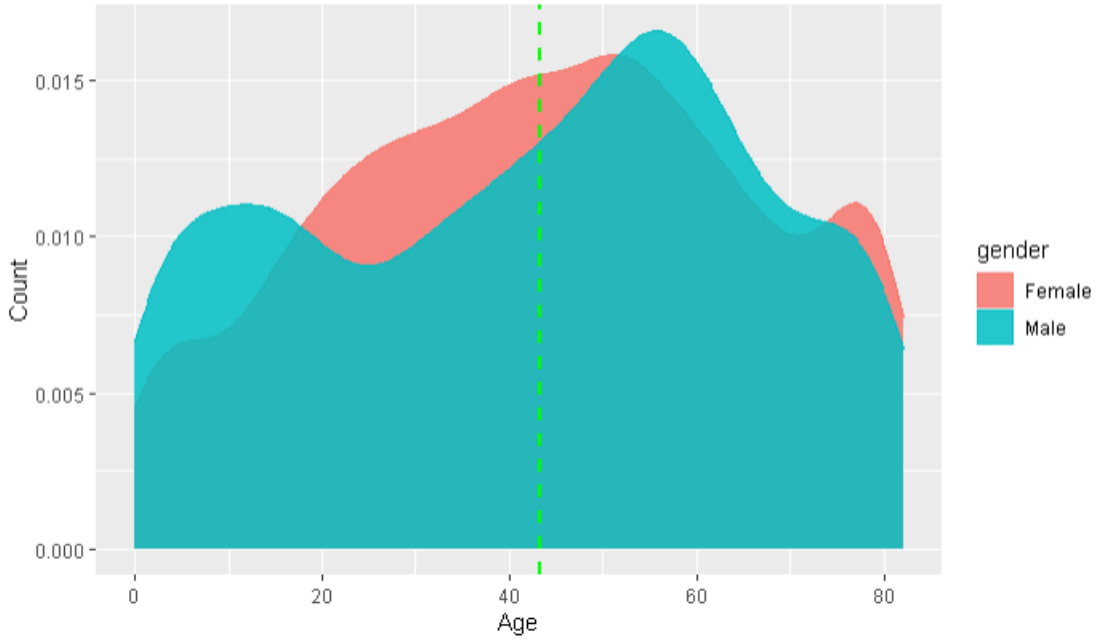

Figure 4: Age and gender distribution

Most people are around 30 - 60 years old, with some more females round the 20- 40 mark. The mean age lies around the 43. So, this dataset is relatively young, which could explain the few stroke cases. Figure 5 shows the age and gender with stroke cases.

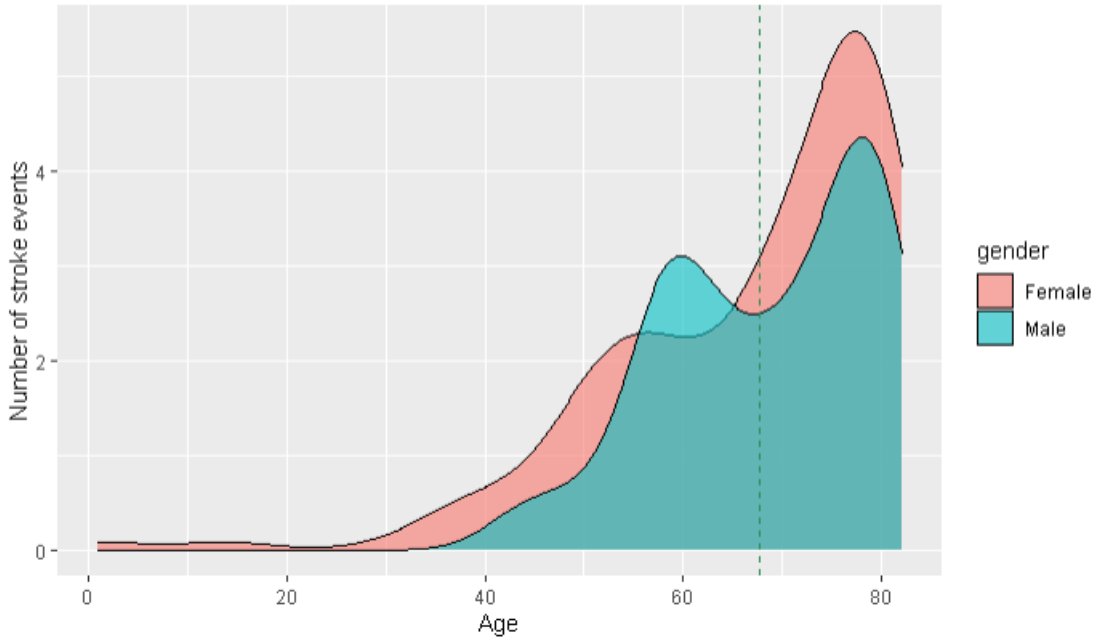

Figure 5: age and gender in relation with stroke

Figure 5 shows that the most stroke cases happen at older age. Especially women of 70+ are sensitive. Men around their 60-life year also seem to have an increased chance.

Cardiovascular diseases

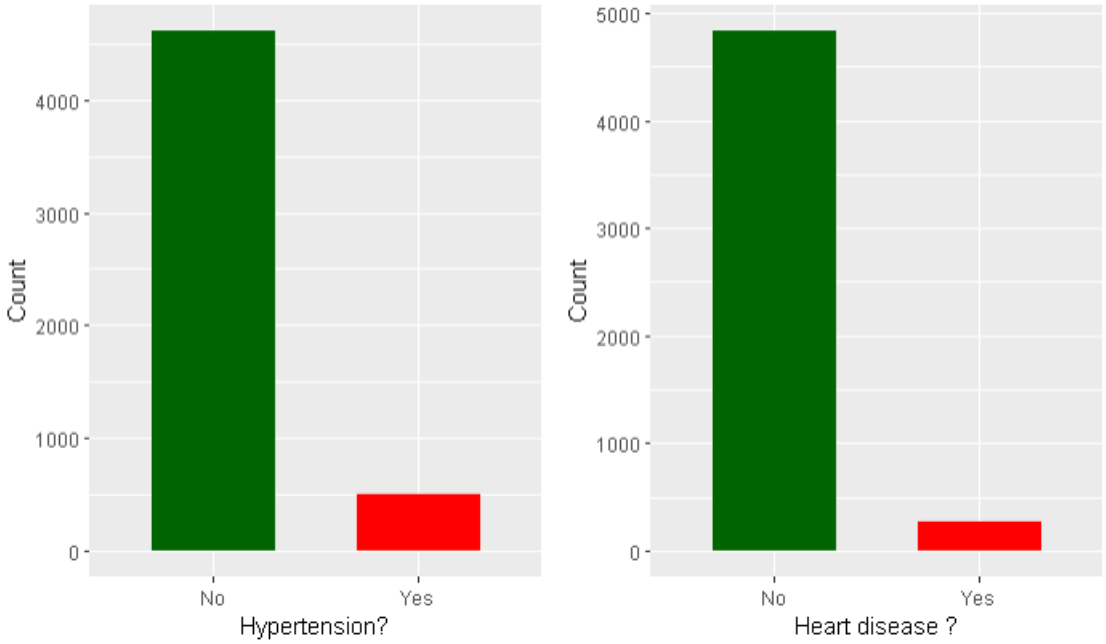

Figure 6: cardiovascular diseases

Looking a figure 6, approximately the same situation applies as with the stroke variable, there is a ~95/5 ratio. This can be the result of that cardiovascular diseases are not common in this dataset. But is this related to stroke?

```
Number of cases with stroke:  249
Number of cases with heart disease:  276
Number of cases with hypertension:  498
```

Figure 7: Cases with cardiovascular diseases

Above numbers (figure 7) are form the EDA. It says that there are 249 cases with stroke of the total 5110 people. 276 of the 5110 have a heart disease of some kind and 498 people have high blood pressure problems. So, there are a small number of people with cardiovascular diseases. But how are they related to stroke events. Figure 6 has number of the relation stroke to cardiovascular diseases.

```
The number of case with stroke and heart disease are:  47
The number of case with stroke and hypertension are:  66
The number of case with stroke and heart disease & hypertension are:  13
```

Figure 8: relation stroke and other cardiovascular diseases

There is a total of 47 people that had a stroke and have a heart disease. Hypertension does it a little bit ‘better’ with 66 combined cases of stroke and hypertension. Only 13 people of the 249 stroke cases have both heart problems and hypertension. Which is around 5%. So, it is safe to say that, based on this dataset, cardiovascular diseases and stroke are not related.

BMI and glucose

Other health variables included BMI and glucose, a high BMI in linked to obesity and a high glucose level is linked to diabetes and a high body weight as well. In the graph below (figure 7) the distribution of BMI and average glucose level in visualized.

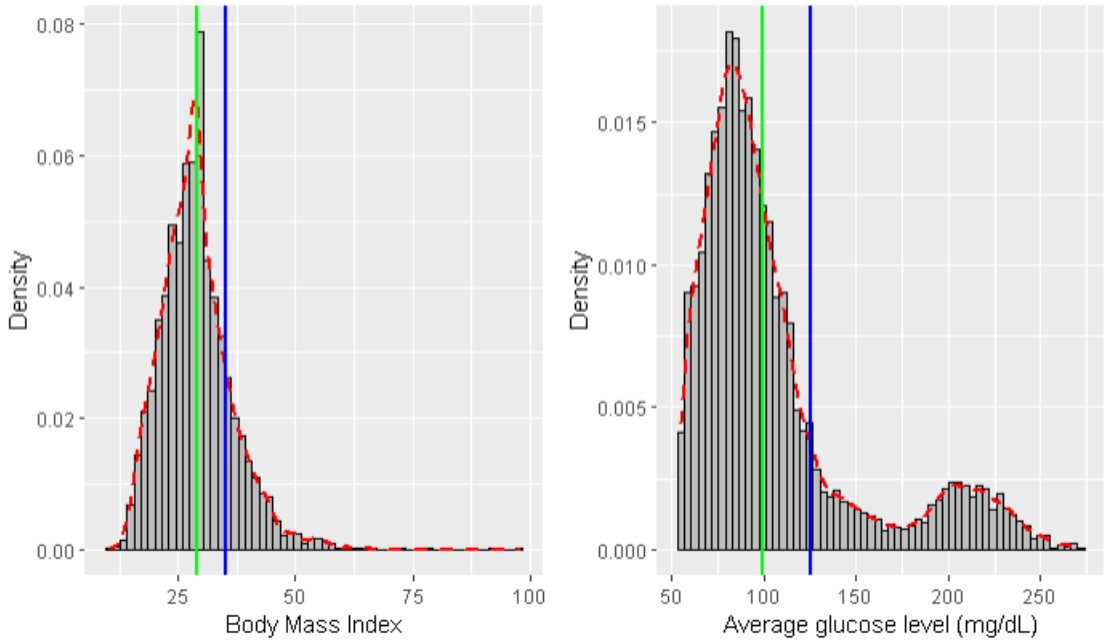

Figure 9: BMI and glucose level distribution

BMI has a sort of normal distribution, with a mean of ~ 28 (green line). A BMI between 25 - 30 means overweight and 30+ means obese. There is a fair amount of people with a BMI higher than 35 (blue line). This means that a reasonable amount of people in the dataset is serious overweight.

For glucose level, the distribution is a little bit right skewed with a bulge toward the end of the x axis. The mean of 106 is high. 99 mg/dL or lower is normal, 100 to 125 mg/dL indicates you have pre-diabetes, and 126 mg/dL or higher indicates you have diabetes. The green line represents 99 mg/dL and the blue line 125 mg/dL. These borders show that, according to the above information. All the people left of the green line are healthy and all the people right of the blue line have diabetes. So, there are some serious unhealthy people according to these numbers.

The distribution tells that there is a fair amount of people that are considered unhealth. But how does this looks in relation to stroke? The assumption is that the most stroke cases are people that are overweight and have a high glucose level. Figure 10 shows all the people that had a stroke, along with the BMI and glucose levels.

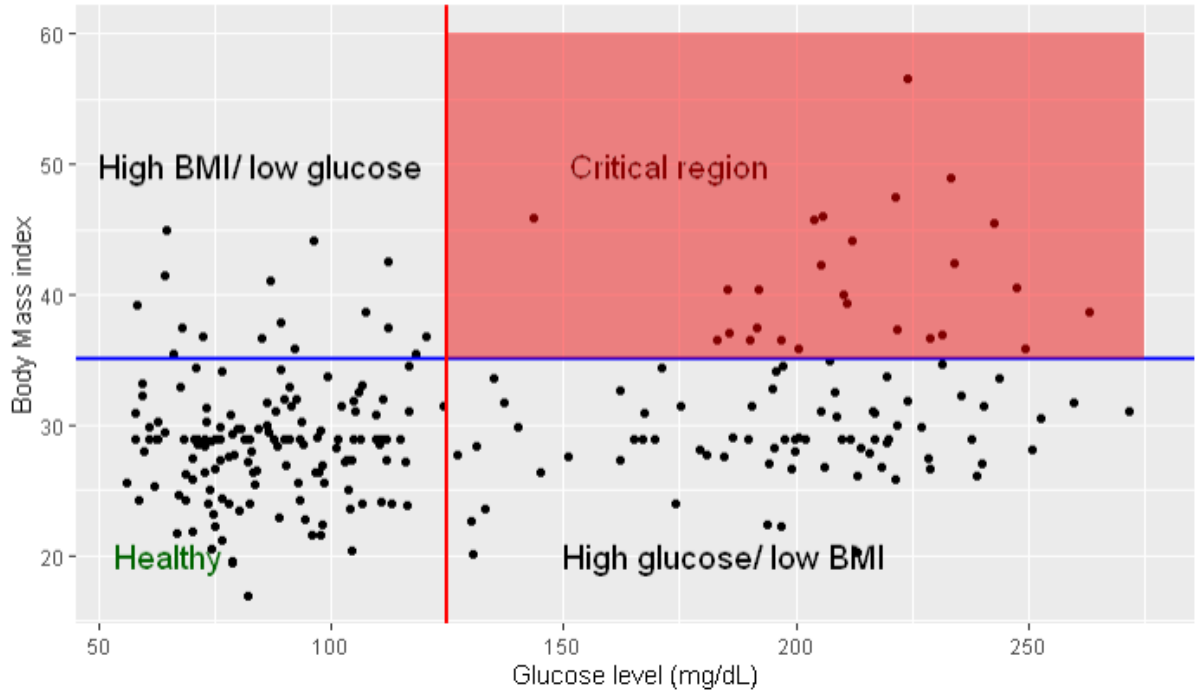

Figure 10: BMI and glucose level relation toward stroke cases

The plot shows all the 259 stroke cases. The visualization is divided into four sections, the blue line represents the border between a good and to high BMI. The red line does the same for glucose level. The top right section are the people that had a stroke and have a high BMI and glucose level. But there are relatively fewer people than expected. Only 26 people living up to that expectation. 74 people have a high glucose level and stroke, and 16 people have a high BMI and stroke. Which means that of the 259 stroke cases 133 people, based on BMI and glucose in this dataset, are healthy.

The only result that can be retrieved from the plot is that glucose may influence stroke chances. Around 30% of the stroke cases have a to high glucose level.

Smoking

Smoking is the last health variable to explore, generally smoking is bad for health and environment. The bad for health part could raise the stakes for a stroke event. In the dataset there are three labels, never smoked, formally smoked and smokes. The distribution is shown below.

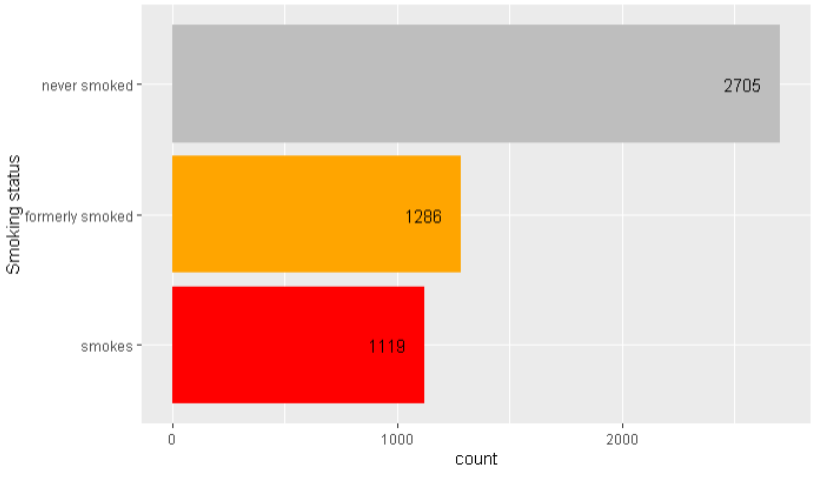

Figure 11: Smoke status distribution

In figure 11 is shown that the majority never smoked. And around 1200 people smoke or have formally smoked. Figure 10 shows the number in relation to stroke.

```
The number of case with stroke and never smoked are:  114
The number of case with stroke and Formerly smoked are:  79
The number of case with stroke and smoking are:  56
```

Figure 12: Relation smoking and stroke

Most people that had a stroke did not smoke in their lifetime. 79 people with strokes formerly smoked, and the smoking group have 56 people in it. SO, this data says that the most people with stroke don't smoke.

The term formerly is vague. A formerly smoked could have quitted 50 years ago and still be out into this box. Also, the people that smoke, are labeled as smoker. But it says nothing about the frequency of smoking. A person could smoke 1 cigarette a week or 500 a day, both will be labeled the same. Probably the smoking status says not much about stroke chances.

Verdict health variables

There isn't a single health variable that is not disputed, they all have their flaws, and none lead to a clear sign of a health issues that leads to strokes. So, the relationship stroke and a bad health is low. If we compare stroke against all the health variables: hypertension, heart disease, high BMI, high glucose level and smoking. Then it will result in only 1 person. One person to have hit with all bad health variables. This does not confirm the theory from the introduction about bad health increases the chances of stroke. But that is only based on this dataset.

Lifestyle variables

There are a couple of other variables that are labeled as lifestyle, these variables are: residence type, work type and marriage status. The idea is that these factors can contribute towards high stress levels, which can affect the health in a bad why. Stress can lead to a high blood pressure and possibly hypertension.

The below plot (figure 13) shows the distribution of marriage. A married life could be stressful, even as a divorce. The 'yes' in ever married tells not the whole story, because a person could have been married in the past but is now divorced. These previously married people also are categorized into married. Which makes the 'yes' records a bit iffy**.**

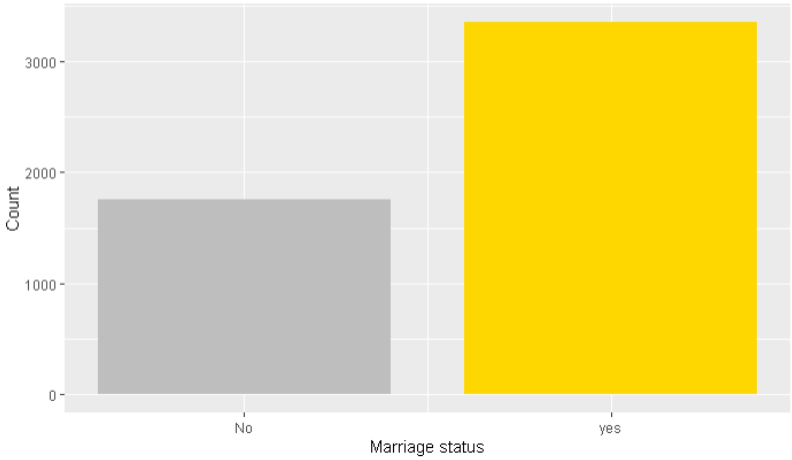

Figure 13: Ever married? distribution

Most of the people are married or have been married at some point in their life. But marriage is age sensitive. Older people have a high chance to be married than young people.

```
Mean age of not married:  21.98
Mean age of ever married:  54.34
```

Figure 14: mean age of marriage

Figure 14 confirms this theory. Age influences marriage status heavily. So, saying that marriage affect stroke chances is more of an age-related thing. 11.6% of stroke cases occurred with none married people. Which means that the majority (88.4%) where married. But as said married people are older and figure 5 showed that stroke occur more with elderly people.

The type of living area can have influence on someone's life, living in a busy urban area maybe contributes to stress. The plot of residence type below (figure 15) shows the spread of living areas.

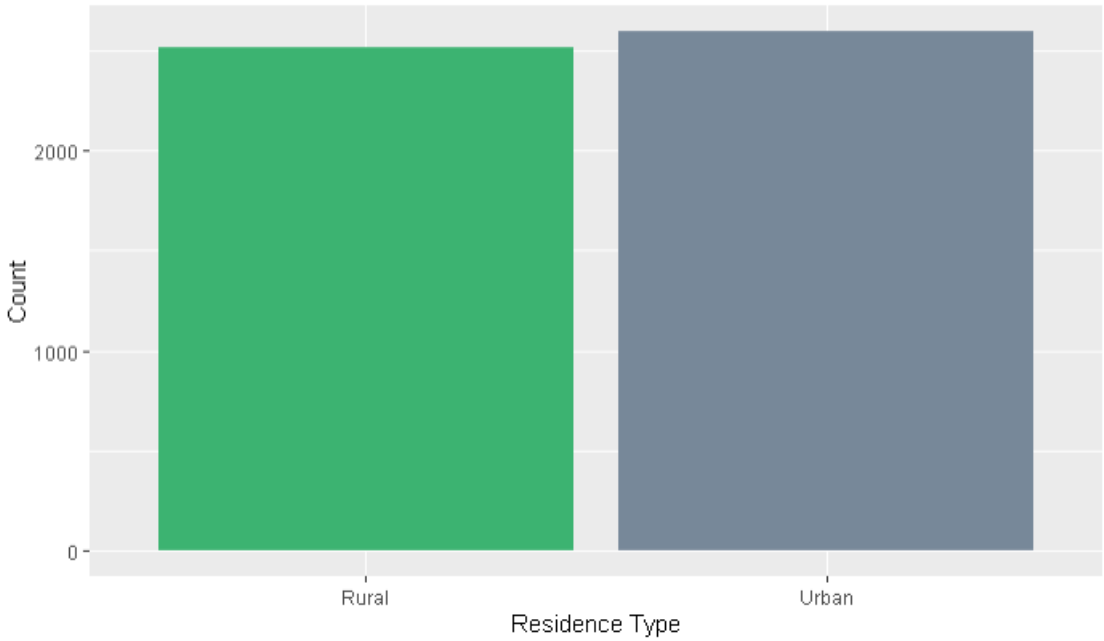

Figure 15: distribution living area

The bar plot of residence type shows a balance between the two types. There can be assumed that mostly young people live in urban areas and elderly people leave the busy areas.

```
Mean age of urban:  43.53
Mean age of rural:  42.89
```

Figure 16: mean age of living area

The mean ages did not show any significant age difference between living rural or urban, So, age and living area are not related. Looking at the cases with stroke, 45.8% lives Rural and 54.2% lives Urban. Not significant difference here.

Also, working type didn’t have an impact on stroke chances. It is safe to say that the lifestyle variables don’t count in for a big difference in the data.

3.2 Machine learning

The Goal in the machine learning project is to predict a chance of a stroke event based of the health and lifestyle variables. In the section below the word accuracy is many times being said. In this experiment accuracy means the precision of the ML algorithm in which it can rightly predict the chance of a stroke label. So, an accuracy of 70% means that ,based on the inputted values of that person, there is a 70% chance that a person's stroke history is right. In other words, if the output is true, you're likely to get a stroke somewhere in the near future.

Fixing imbalance

With imbalanced data sets, an algorithm doesn't get the necessary information about the minority class to make an accurate prediction. Which will lead to misleading accuracies.

There are a few ways to fix imbalanced dataset:

- Under sampling
- Oversampling
- Synthetic Data Generation (SMOTE)
- Cost Sensitive Learning

In the log, all four techniques are evaluated and considered. But the SMOTE technique is the best suited for this research. The SMOTE algorithm creates artificial data based on feature space. similarities from minority samples. It creates a random set of minority class observations to shift the classifier learning bias toward the minority class. The result Is a bigger dataset, but balanced.

Weka self-offers a handful of filters and tools. One of these tools is 'ClassBalancer'. It reweights the instances in the data so that each class has the same total weight. This form of balancing will be applied on the imbalanced data, which makes the datasets to test a total of three. Normal, Normal with classbalancer and SMOTE modified data. These datasets are tested against eight algorithms.

Table 3: comparing dataset

Type of algorithm	Normal	Normal (classBalancer)	SMOTE
ZeroR	95.1	49.8	59.2
OneR	95.1	-	97.7
J48	95.1	60.7	90.2
RandomForest	94.4	54.6	95.6
NaiveBayes	89.1	76.2	80.1
SimpleLogistic	95.1	75.1	81.7
Logistic	95.1	76.4	81.8
SMO	95.1	76.3	82

Table 3 shows the difference in accuracy between the datasets. the normal imbalanced dataset almost always has an accuracy of 95%. This is because the imbalance in the dataset is equal to 95/5. So, the algorithm deals with this by saying that almost everything is false. In 95% of the cases, that is correct. As earlier concluded, the dataset with the imbalance is not good.

Looking at the dataset with the ClassBalance filter. The accuracies are reasonable balanced, with the bayes and function algorithms getting +70% accuracies. But the way how the ClassBalance filter acts is questionable. To put it simply, records form the minority are duplicated to achieve a 50/50 ratio and random duplication is not desirable.

The SMOTE dataset has a couple of overfitted accuracies, but the logistics algorithms and bayes preforms well. The way that created the extra records is also more desirable that the ClassBalance filter. So, The SMOTE dataset is the one to go with in the next section of the ML phase.

Baseline

Beforehand it is important to know, what makes algorithm 'good'? What are the characteristics of a desirable algorithm. For this question some quality metrics are determined.

Example of different quality metrics:

- accuracy
- speed
- size of tree*
- Confusion matrix

Obviously a high accuracy is desired. Then the question is, how accurate is good enough? As base, an accuracy of 90% is considered good. with diseases like strokes, 80% is not good enough. Which means that one in five is classified wrong, which can have serious consequences. Also, how is this accuracy achieved? Look out for overfitting.

Speed of the algorithm is important by big usage. Implementing the model in a local environment, 1 to 2 minutes is okay. As it would not calculate more than 10 times a day. But in big scale usage, a fast algorithm is a demand.

In case of a tree, the size of a Decision tree is an important factor. With a large tree with many leave, overfitting could be possible. As the algorithm is too specific on the trainings set. A small, but accurate tree is desirable.

A confusion matrix is a specific table that allows visualization of the performance of an algorithm. Each row of the matrix represents the instances in an actual class while each column represents the instances in a predicted class. This also indicates the rate of true positives, true negatives, false positives and false negatives. The errors in the matrix will weight heavy on the model performance.

In conclusion: accuracy, confusion matrix and size of tree will weigh the heaviest in selecting a good algorithm.

The baseline is set with zeroR, a simple algorithm that looks at the distribution of the classifier. In The test options, cross-validation is set to 10 Folds (default).

TP = True positive rate

FP = False positive rate

Table 4: Baseline performance

Algorithm ->	ZeroR
Accuracy (%)	59,2
Speed (sec)	0,01
T TP (%)	0
T FP (%)	0
F TP (%)	100
F FP (%)	100

Table 4 shows the baseline 59.2%. this is the distribution of the True and False. So, every algorithm that performs worse than the baseline is considered bad.

Advance algorithms

Advanced algorithms include algorithms that are 'smarter' than zeroR, the algorithms and their performance are listed in table 5

Table 5: Advanced algorithms performance

Algorithms ->	OneR	J48	RandomForest	Naive.Bayes	Simple.logistic	Logistic	SMO
Accuracy (%)	97,7	90,2	95,6	80,1	81,7	81,8	82
Speed (sec)	0.02	0.43	3.6	0.04	1.14	0.2	18.5
T TP (%)	100	85,2	93,1	77,4	77,1	77,5	78,3
T FP (%)	3,9	6,2	2,5	18	15	15,2	15,3
F TP (%)	96,1	93,8	97,5	82	85	84,8	84,7
F FP (%)	0	14,8	6,9	22,6	22,9	22,5	21,7
Size of tree	.	925	.	.	.	.	.

there are three algorithms with accuracy >90% and four algorithms with accuracies around the 80%. Most algorithms are fast, apart from SMO. Especially SMO is slow. The high scoring algorithms have a low FP rate by True and False. The lower scoring algorithms have a more divided spread of TP and FP rates.

On paper, there are three excellent algorithms, but they are likely infected with (a form of) overfitting.

Ensemble learners

While the advanced algorithms look good, there are also ensemble learners. The main purpose of these is using an ensemble model to use a group of weak learners and form a strong learner. Random forest is an example of an ensemble learner. It uses bagging (which is explained later) and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

There are 3 ensemble learners that will be tested: Bagging, AdaBoostM1 and stacking.

Table 6: Ensemble learners

Algorithm ->	Bagging	AdaBoostM1	Stacking
Accuracy (%)	91,5	81,5	59,2
Speed (sec)	1.09	0.32	215.84
T TP %	89,2	75,7	0
T FP %	6,9	14,4	0
F TP %	93,1	85,6	100
F FP%	10,8	24,3	100

Table 6 shows that bagging preforms the best, fast, accurate and a low percentage of false positives. AdaBoostM1 is not worse, but also not great. Stacking took over 45 minutes to produce the baseline accuracy as best.

Parameter optimalization

Three algorithms where selected to continue with, J48 and RandomForest for their parameter flexibility and Bagging for it overall performance. The parameters are tweaked to achieve even higher accuracies. Mainly J48 and Randomforest were lightly overfitted. To tackle this, some parameters were adjusted.

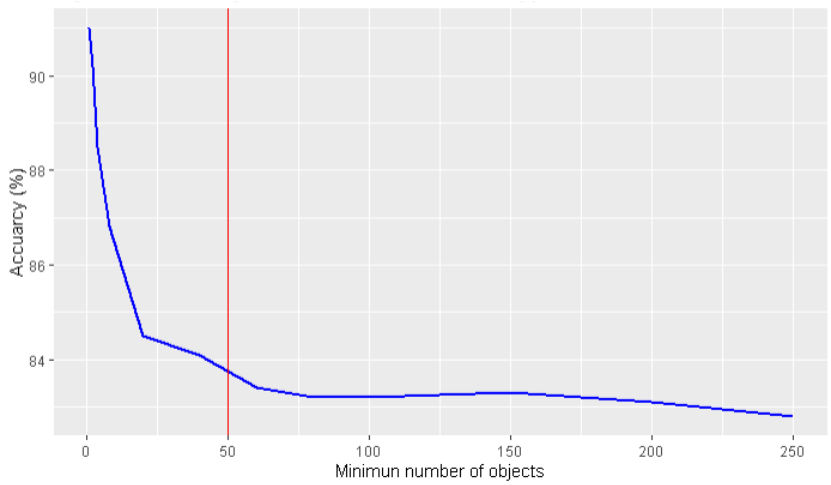

Figure 17: Accuracy J48 in relation to minNumObj parameter

The number of objects in figure 16 represents how many objects are minimal needed for a leaf to branch off. A high number of objects is stricter. It seems that after 50 objects the accuracy stays stable.

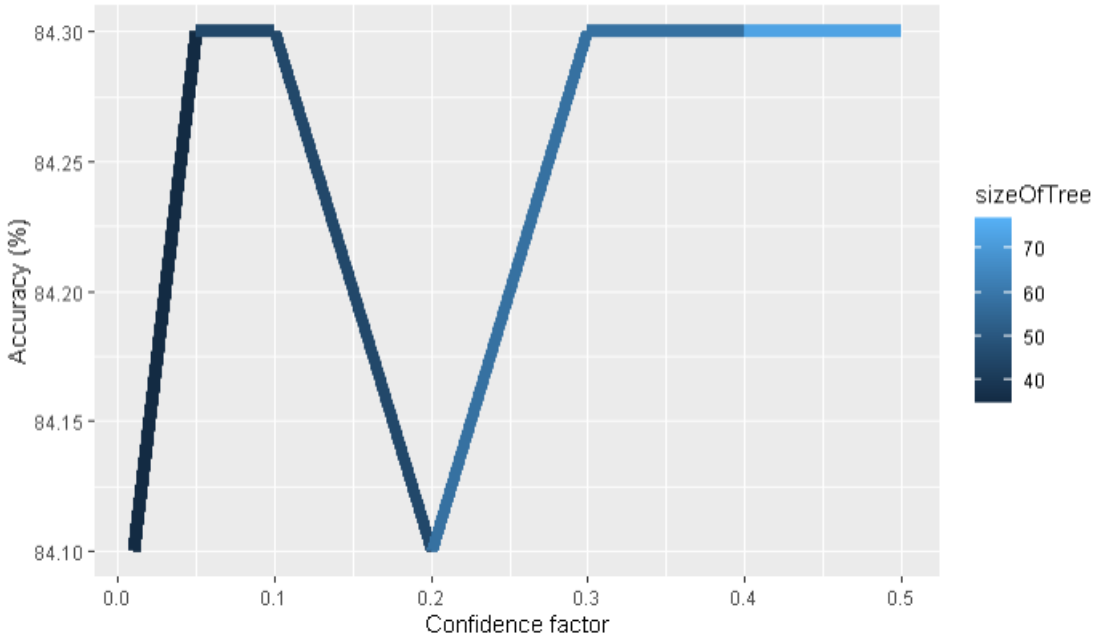

Figure 18: J48 confidence factor difference with minNumObj = 50

Figure 18 describes the effect of raising the confidence factor, this does not seem to matter to much, only the size of the tree grows which is unwanted. The optimal result of J48 = **84.13%**

Random forest takes a parameter called maxDepth. This determines how 'deep' the tree may go. The default is 0, which means unlimited. This unlimited depth causes complex overfitted trees, which get high accuracies. So, by limiting its depth, the accuracy gets more reliable. The figure 19 underneath looks at the effect on accuracy with different maxDepth.

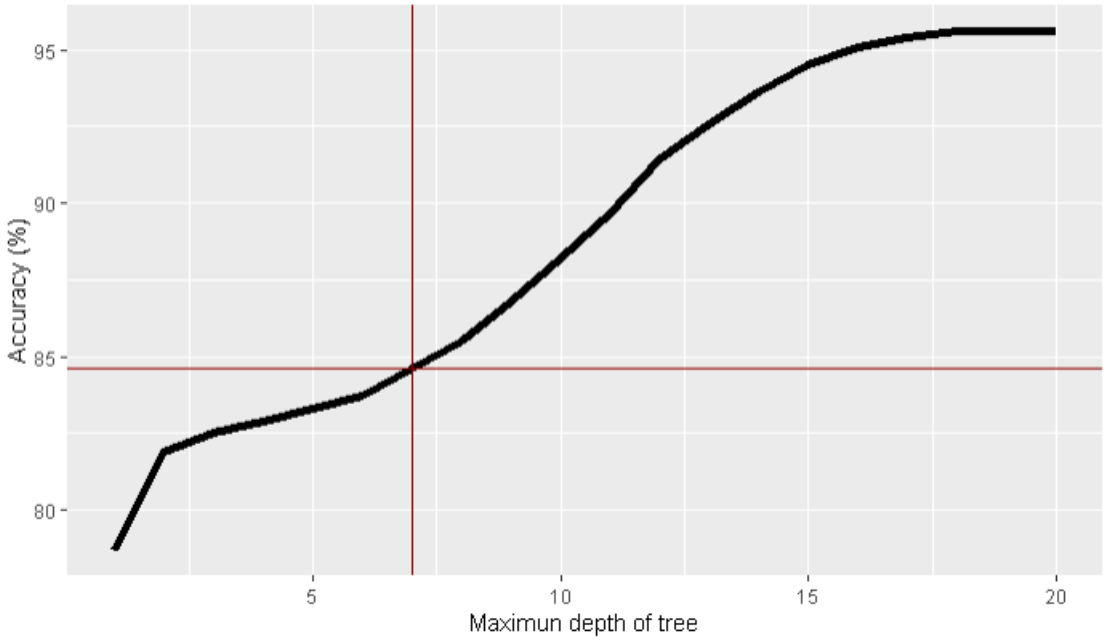

Figure 19: RandomForest: Effect of maxDepth parameter on Accuracy

The accuracy increases slowly with each higher depth level. After a depth of 15, it flattens out around the 95% mark. To tackle overfitting, a low depth is desirable, but a depth of 2 gives 'low' accuracies. but a too high depth creates overfitting problems. So, a depth of 7 is taken. Not too low and not too high. A depth of 7 gives an accuracy of **84.6%.**

Bagging or **B**ootstrap **agg**regat**ing** is an ensemble model to improve stability and accuracy of combined machine learning algorithms. It also reduces variance and helps avoid overfitting. The preventing of overfitting is great in this experiment. The Bagging model has a couple of parameters to tweak, but the most impacting is numIteration. The number of iterations the model preforms. The default is 10. Which earlier gave an accuracy of 91,5%. But more iteration means that it will be slower. Time in this case is also an important aspect. Figure 20 shows the effect of number of iteration and time.

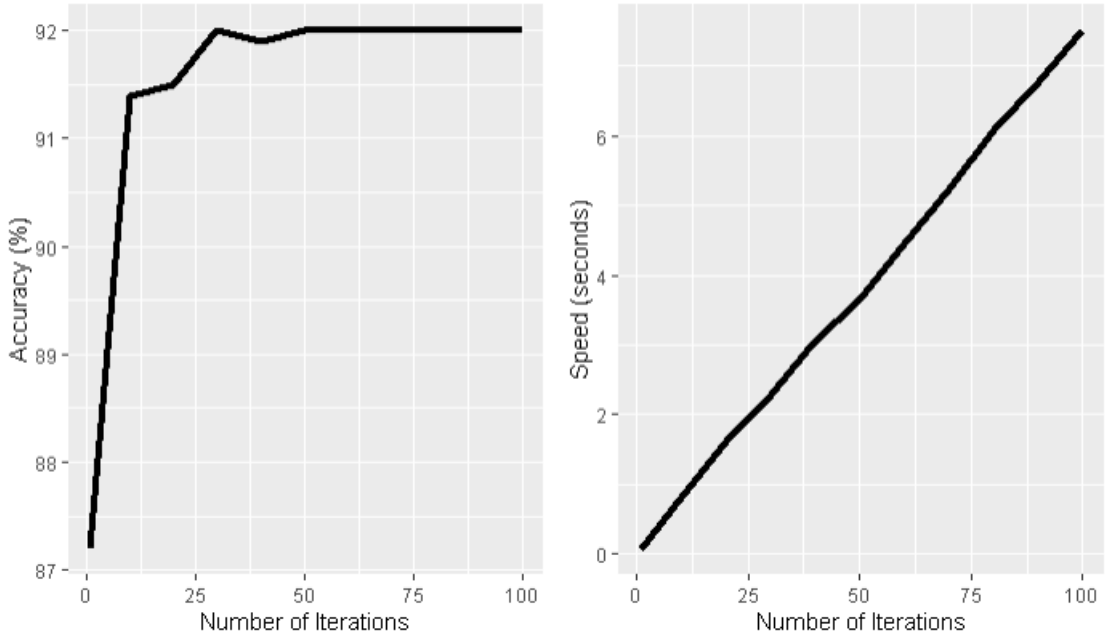

Figure 20: Bagging parameters optimalization

The optimal number of iterations is around 30, where it peaks in accuracy and only takes 2.29 seconds to build a model. Fast and accurate. Which gives an accuracy of **92.0%**

Machine learning verdict

After parameter optimization, there are three accuracies of three models: J48 with 84.13%, RandomForest with 84.6% and Bagging with 92%.

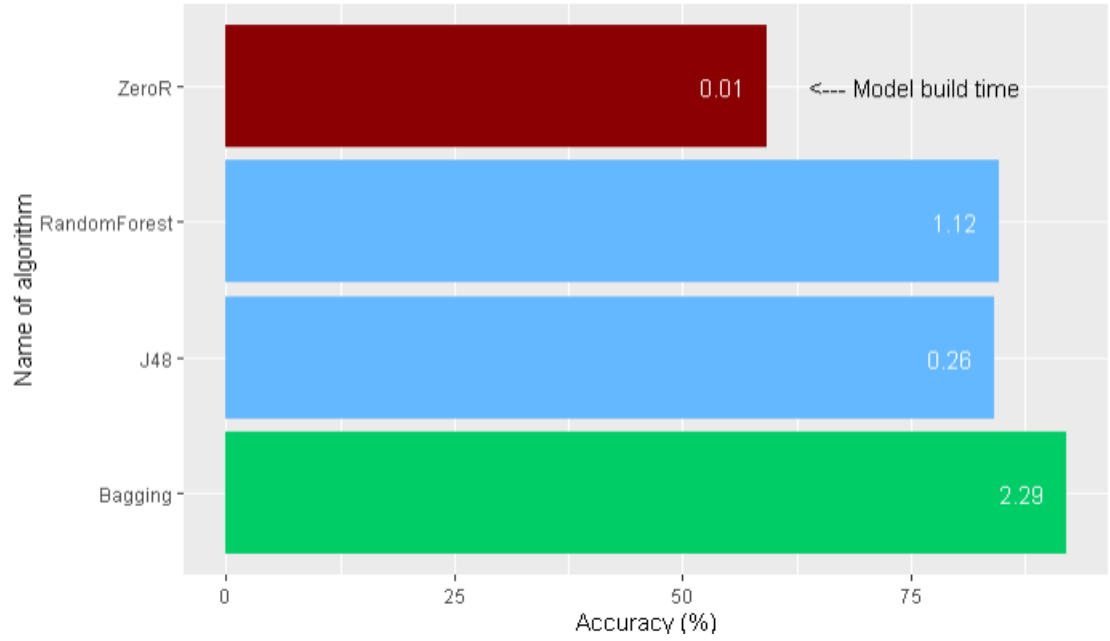

Figure 21: comparing optimized algorithms

Figure 21 describes the differences in the optimized algorithms. Overall Bagging preforms the best, it is the 'slowest' of the 4 but still fast enough for not heavy usage. If we recall the quality metrics: The demand was a high scoring algorithm that also was fast. A good confusion matrix was also desirable.

Table 7: Confusion matrix

	TRUE	FALSE
TRUE	90.6	9.4
FALSE	6.9	93.1

Table 7 shows the confusion matrix. This confusion matrix shows at the top the predicted value and the rows as actual values. The numbers are percentages. The matrix shows a solid 90+ percentage for The True positives of True and false. There is a 9.4% false positive rate, which is not great but not the worst. 6.9% is the false negative rate, which is a little bit too high. It means that roughly 7% of all the people are told that they don't have a higher change on stroke events, but they do.

A ROC curve is a plot that illustrates the diagnostic ability of a binary classifier system. It is a tool to control and select optimal models and

discard worse models. It is related to a cost/benefit analysis. The Area under the curve (AUC), describes the performance. A perfect classification has an AUC of 1. Which means that the lines lay prefect along the axis. How more the bulge lies in the direction of the top left corner, the better the AUC is.

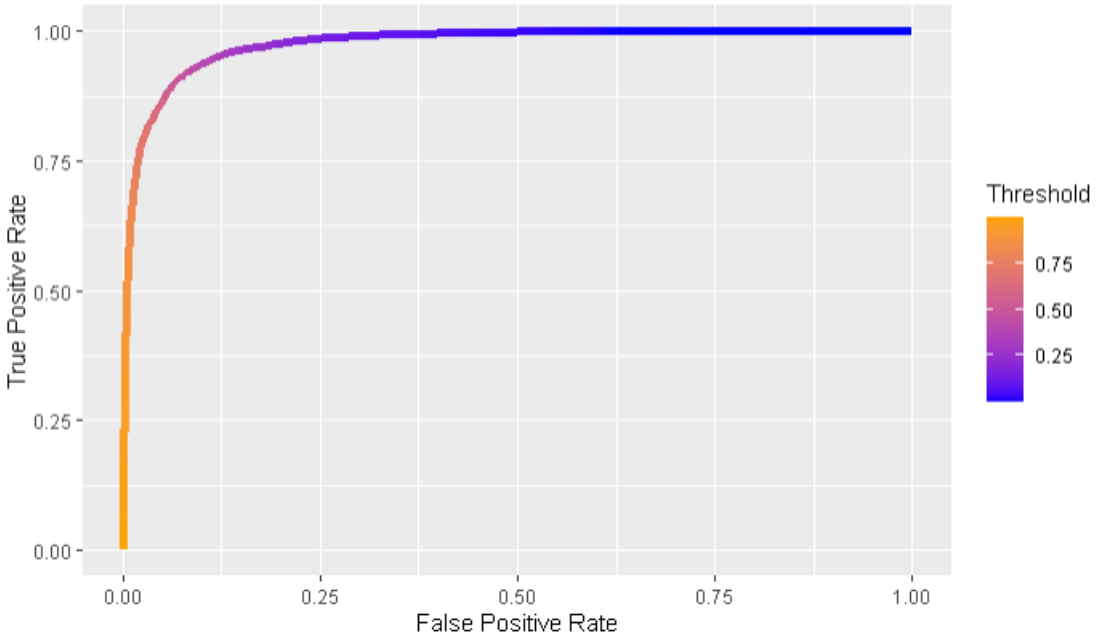

Figure 22: ROC Curve, Area under ROC = 0.9751

The ROC curve of the bagging algorithm is shown in figure 22, with an AUC of 0.9751, the model preforms very well. The curve indicates a good model.

Overall, the bagging algorithm is the best. High accuracy, fast and a good scoring confusion matrix. Also, the AUC is promising. Therefore, is the bagging algorithm chosen to be used.

4. Conclusion & Discussion

With a trough analysis of the data and a multistep machine learning part, the result is an optimized bagging model that can be used into the JavaWrapper. This model is the result of a critical data analysis where the data is formed to suit the algorithms.

Coming back to the research question : ***Is it possible to produce an accurate algorithm with machine learning, to calculate the risk of a stroke based on lifestyle variables of people with stroke history and people that never had a stroke?***

Yes, for 92% sureness it can be predicted if someone has a higher chance on stroke than other people.

In the EDA it became clear that not all variables were very significant, only age showed a relation between it and stroke chances. The health variables were not exciting, and the lifestyle variables added little to nothing. Nevertheless, with this set of data, 92% is still possible. Which mostly came form the great algorithm of the bagging model.

But there are also point of criticism. The SMOTE method introduced more observations to the minority. But it was never verified that the method was responsible for a huge change. The accuracy of the algorithms improved, but there was not sufficient analysis of the effect on the variables themselves.

Another discussion point is the data itself. The idea of the variables is good, but lifestyle variables added little. The variables of health where on the other hand better to work with. In the spirit of data science, the data needed a reasonable amount of modifying. Which could have done better. Also, the group with stroke events was relatively small. An equally divided group of people with stroke history and people who don't was more practical. Practical in the sense of comparing two groups against each other.

Finally, the research produced an algorithm that can predict a higher chance of stroke with a sureness of 92%. This can be called a success.

5. Project proposal

A possible project for High-Throughput High Performance Bio-computing minor, can be continuing this project to achieve a far higher accuracy. The goal is to improve the model in such a way that it can be used and deploy in the real world. That it would be a useful and user-friendly application. Such a as desktop app, where a user can input the health variables, and the model accurately predicts the chance of stroke. The target audience will be the medical world. The output of the program will be the stroke chance and some useful visualizations.

Bibliography

American stroke association. (2022, 05). *About Stroke*. Retrieved from Stroke.org: https://www.stroke.org/en/about-stroke

CDC. (2022, 11 02). *About stroke*. Retrieved from Center for disease control and prevention: https://www.cdc.gov/stroke/about.htm#Ischemic

MCOR. (2020, 07 27). *Man With brain stroke symptoms*. Retrieved from Medical center of Oak Ridge: https://www.mmcoakridge.com/getting-your-life-back-after-a-stroke/

WHO. (2020, 12 09). *The top 10 causes of death*. Retrieved from World Health organization : https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

WSO. (2022). *Learn about stroke*. Retrieved from World Stroke organization: https://www.world-stroke.org/world-stroke-day-campaign/why-stroke-matters/learn-about-stroke#:~:text=Stroke%20has%20already%20reached%20epidemic,the%20world%20have%20experienced%20stroke.