Context

According to the World Health Organisation (WHO), stroke is the 2nd leading cause of death globally, responsible for approximately 11% of total deaths.

Scope

The repository contains the code to data analysis of underlying factors that lead up to stroke and finding an ML model that is able to predict whether a patient is likely to get a stroke (stroke == 1) based on input parameters like gender, age, various diseases and smoking status. The focus is on trying out different techniques.

Dataset used

You can find the original dataset in the stroke_data.csv file. The data contains information about:

- id: unique patient identifier
- gender: "Male", "Female" or "Other"
- age: age of the patient
- hypertension: 0 (doesn't have hypertension) or 1 (has hypertension)
- heart_disease: 0 (doesn't have a heart disease) or 1 (has a heart disease)
- ever_married: "No" or "Yes"
- work_type: "children", "Govt_job", "Never_worked", "Private" or "Self-employed"
- Residence_type: "Rural" or "Urban"
- avg_glucose_level: average glucose level in the blood
- bmi: body mass index
- smoking_status: "formerly smoked", "never smoked", "smokes" or "Unknown"
- stroke: 1 (if the patient had a stroke) or 0 (if the patient didn't have a stroke)

Exploratory data analysis

Basic understanding of data

Some interesting findings so far from looking at the DF:

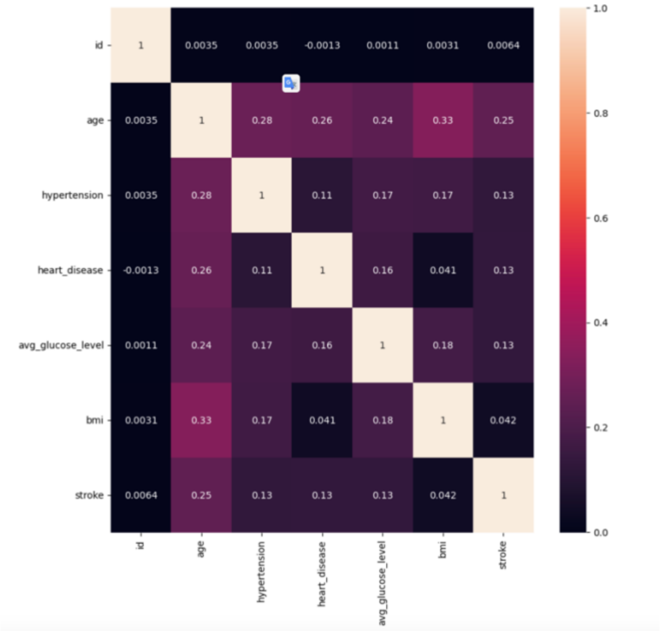
- the id column is not useful for us and we will drop it
- the age seems evenly distributed across all age levels
- both hypertension and heart_disease are categorical boolean features (already translated into 0 and 1), and we already see that they are both highly imbalanced with a majority value 0
- we will most likely see some outliers in the avg_glucose_level and bmi
- bmi contains some NaNs, which we will need to deal with
- the target variable stroke is also highly imbalanced, with majority value 0
- in general, we will deal with a lot of categorical features in this dataset. Only age, avg_glucose_level and bmi are numerical (continuous) variables

Correlation between numeric columns

We can see that some features are clearly correlated with having stroke or not - hypertension and other heart disease, high avg. glucose levels and age.

There is also a high correlation between age and all the other variables, and also noticing specifically bmi, which seems to not be directly correlated with stroke despite our assumption, but is very highly correlated with age.

This overall correlation with age is in line with the assumption that the older the person is, the higher likelihood of these diseases.



Stroke(target)¶

✓ *Initial Assumption*: As with these kinds of diseases, usually the sample is imbalanced with majority consisting of people with no stroke.

Summary of findings: In our sample, stroke occurs in around 4.9% of the observations. This is a highly imbalanced dataset. We will have to use certain balancing techniques to make the prediction for stroke == 1 more accurate.

Gender¶

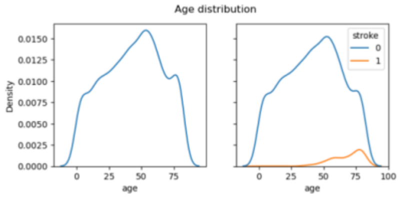
⊗ *Initial Assumption*: Gender plays a role in stroke. Males are more likely to get a stroke than women.

Summary of findings: Gender doesn't seem to have an impact on the fact if a person has a stroke or not. There is only 1 observation in the category 'Other' with stroke == 0, so I will drop this row during the feature engineering to reduce dimensionality.

Age¶

✓ *Initial Assumption*: Age plays a role in getting a stroke. The older the person, the more likely they are to get a stroke.

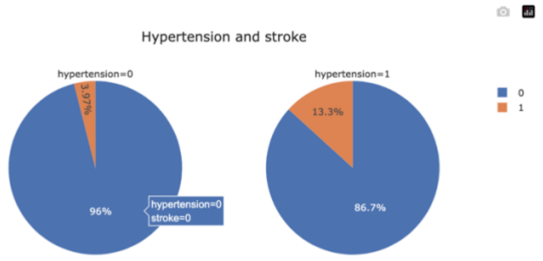
Summary of findings: The age distribution of this sample is from babies to elderly age groups. The most dominant group is between 40-60 years. The age distribution from the stroke perspective is very different for people with stroke and without. For people with stroke, the peak is at a higher age around 80.

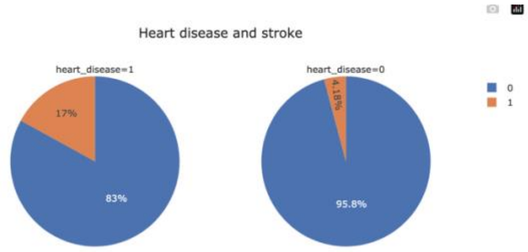


Heart-related issues - heart disease and hypertension

✓*Initial Assumption*:: Heart diseases and hypertension is a factors in developing stroke. People with one of these conditions are more likely to get a stroke

Summary of findings: Only 10% of the overall number of observations have hypertension and 4.5% heart disease. For both of these conditions, the % of people with stroke is 3-4 times higher than for people without these conditions (3x for hypertension, 4x for heart disease)
A higher value of glucose level indicates hypertension or heart disease or stroke, and highest when a combination of heart-related issues and stroke is present.





Marriage

✓*Initial Assumption*: Marriage contributes to a more stressful life and therefore can be a factor in getting a stroke.

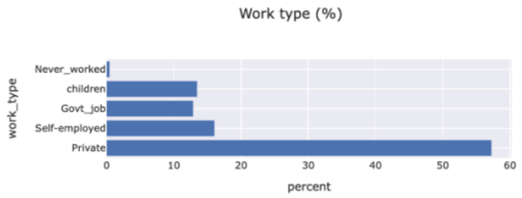
Summary of findings: There are twice as many ever-married people in our observation than never married. In our overall sample set, 66 % are people who have ever been married. 4x higher percentage of people who have ever been married had a stroke compared to those who have never been married (6.6% vs 1.6 %).
The proportion of people who have been married vs. not married is the same for males and females, 63% for males and 66% for females.

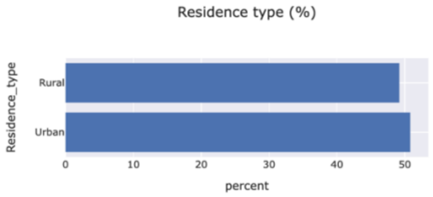
Work and living¶

✓✗*Initial Assumption*: Self-employed people, people working in the private sector and residing in urban environments have higher levels of stress and so a higher probability of developing stroke.

Summary of findings: Most of the people in the sample are employed in the private sector (57%). Logically, almost none of the children segment had a stroke. On the other hand, the % of self-employed with stroke is higher than the other types of work. We don't see the same for private-sector workers.

The distribution of rural and urban dwellers in our sample is the same. The % of urban dwellers with stroke is slightly higher than for the segment with no stroke.

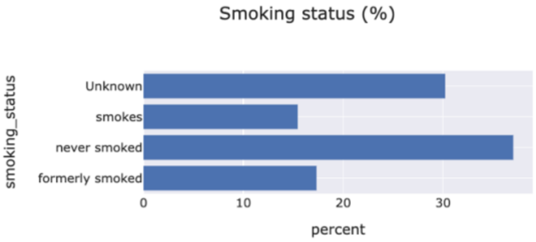


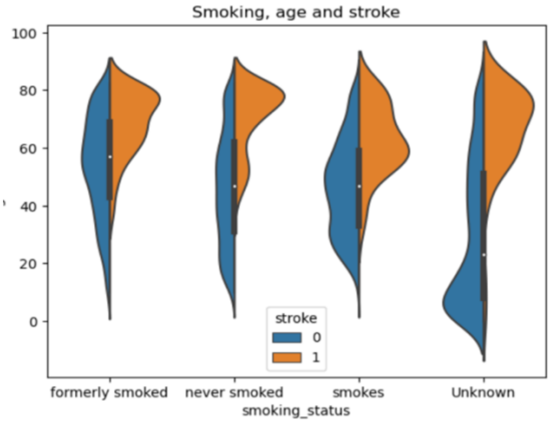


Smoking

✓*Initial Assumption*: Smoking is a factor in developing stroke. Men are more likely to smoke than women.

Summary of findings: 37% of the sample has never smoked. 30% of the sample hasn't answered this question - this is quite a large %, we are however not able to answer why nor are we able to replace the value base with the status based on other variables.
There is a larger % of females who have never smoked (41% females vs 31% males).
For those who have formerly smoked, but already stopped, the % of having a stroke is the highest. My assumption is that deteriorating health might be a factor in their decision to stop smoking. Interestingly, people who currently smoke have a similar % of strokes as those who have never smoked and had a stroke. However, when looking at the age distribution, for those who have never smoked, the mean age is much higher than for those who smoke.





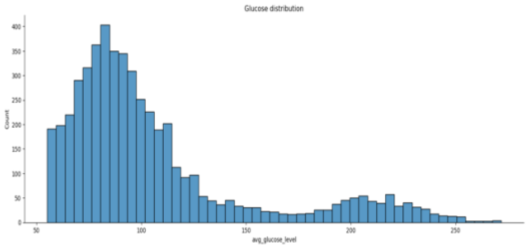
Other health indicators - BMI and glucose level¶

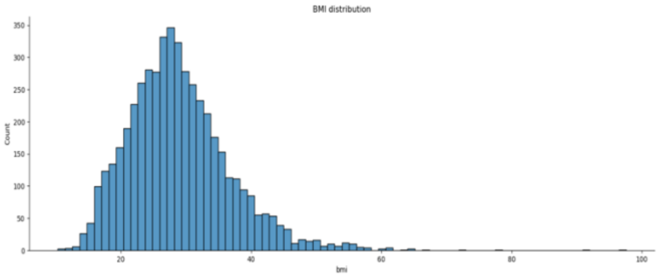
Assumption: Higher BMI and glucose levels are factors in developing stroke.

Summary of findings: The distribution of BMI seems to be in line with the bmi distributions I found on the internet. The generally accepted levels:

- <18 underweight
- 18-25 normal
- 25-30 overweight
- >30 obese - very very very obese

We also see that the distribution is slightly skewed to the right. Due to this when doing feature engineering, we will replace the NA values of BMI with the median (and not the mean).





Interestingly, there are 2 peaks in the distribution of avg_glucose_levels. The distribution is right-skewed.

Age is a huge factor in the probability of getting a stroke. With growing age so do the observations with higher glucose levels grow (>150) and there is a much higher concentration of people with stroke at these levels. It seems that glucose level will have an impact on the prediction of stroke.

What is also clear, what we already know from the correlations, is that age and glucose level are somewhat correlated.

Basic feature engineering and selection

In this part, we have performed the following:

- **remove the column 'id'**, because this is not a feature that can help us in the prediction
- **fill in the NaN values in the BMI** column with a reasonable estimate – the missing values were replaced with the median based on the variables gender and ever_married.
    - this selection is based on the fact that the distribution is slightly skewed to the right
    - with the boxplot, we saw that the median is different from the values of ever_married and gender
- turn the **categorical variables into features** that the models can work with – I have used vectorization in this part
    - split into categorical binary (drop_first = True) and categorical non-binary

I don't see a reason to remove any of the additional features from the set:

- we don't have a huge set of features
- they all seem to be able to bring some value to the prediction. I will however experiment with feature selection libraries during the modelling part

Splitting into testing and training

For the split, I used the standard 20% testing 80% training split, with 'stratify=y' to make sure we keep the balance of the stroke categories as we are dealing with a highly imbalanced set

Model training and evaluation¶

In this part I started with the following modelling techniques and then compared the results:

- **Without balancing**
- **Adding sample balancing and pipeline**
- **Adding more advanced feature engineering and selection**
- **Removing outliers in numeric columns**
- **Hyperparameter finetuning on our best performant models**

In all the parts, I have applied the most famous classification models which all have their strengths and weaknesses:

- **Logistic Regression**
- **Multilayer Perceptron**
- **Decision Tree**
- **Random Forest**
- **K-nearest Neighbours**
- **Support Vector Classification**
- **Gaussian Naive Bayes**
- **AdaBoost**
- **GradBoost**
- **XGBoost**

Modelling without sample balancing¶

Approach: I first tried very quickly the most famous classifier models:

1. **use stratified fold cross-validation on the balanced training set**
2. **fit the model,** make predictions for the testing set
    - create **classification report**
    - **confusion matrix on the testing set**
3. **analyze the results**
    - decide on the best one from this set of models
    - decide on what could be the next steps towards a better-performing model

Results:
All these models are performing weakly and are not usable in real life. While on stroke == 0 they perform well, they are not able or are very weak in predicting stroke == 1. From all the models I tried, only a few were able to do at least some prediction for stroke ==1 (f1 !=0) and only the Gaussian Naive Bayes was able to predict stroke, with an F1 score of 0.16 (recall of 0.9) for stroke == 1.

In this case, I believe that **recall for stroke == 1 is the more important aspect of the f1 score** because I consider the **consequences of missing a stroke (false negatives) more severe and harmful than making a prediction that a patient will have a stroke when they won't have it** (which would be **false positives** in this situation). Of course, there needs to be a balance, but in general I will be looking into making the prediction in general, and recall for stroke == 1 in particular, better.

Modelling with sample balancing and pipeline

Approach: We will use **SMOTE oversampling and undersampling** to achieve a **better-balanced set in terms of stroke == 1.** We will also use a pipeline which improves the performance and results of using balancing techniques with a model. We will select the sampling strategy in SMOTE to 0.1 and oversampling to 0.5. Based on the recommendation from the data science community, combining undersampling with oversampling in this proportion is a good starting point and is optimal in terms of results.
The way we will apply this, similarly to the modelling without sample balancing:

1. create pipeline
    - the under/oversampling will happen on the training set
2. fit the model and use stratified fold cross-validation on the balanced training set
3. make predictions for the testing set and analyze the results
    - create classification report
    - confusion matrix on the testing set

Results: **Using balancing has significantly improved the performance of certain models**, where previously most models were not able to do a prediction for stroke == 1 in general and **recall for stroke == 1 in particular.** We can conclude that **including balancing has made a huge difference and we will keep balancing in the further experiments we will try.**

It is difficult to select the best-performing model, as all of them are still weak in terms of performance. From all these models, I would select the SVC as the best-performing model, because of the relatively high recall for stroke == 1 (compared to other models) (0.6), while keeping a decent performance for stroke == 0. While the Gaussian NB still has a very high recall for stroke == 1, it performs poorly in the other metrics (e.g. recall for 0 = 0.41) and also has a very low precision for stroke == 1. Even though there are some models that are performing better in the other metrics, because of the recall for stroke == 1, this is what I would select.

Note: I have also tried different balancing sampling strategies, but this seems to work best.

Adding more advanced feature engineering and selection¶

Approach: Compared to the previous version of the models, we will play around with column transformation in a bit more advanced way. We will keep the following:

- preprocessing: removing id, removing row with 'Other' as gender, replacing NaN in BMI with values from our BMI_guesser function
- SMOTE oversampling, undersampling with the same parameters as before (this seems to work best for our set)
- working with pipeline
- cross-validation on the training set

We will add:

- creating **ColumnTransformer and adding it to the pipeline**
    - scaler on numerical values (StandardScaler)
        - Note: I also tried with MinMaxScaler but the results were slightly worse
    - one hot encoding on categorical values
    - **feature selection** with SelectPercentile(chi2, percentile=50)

Results: **Using a standard scaler with oversampling and undersampling helped significantly in certain models in terms of recall for stroke == 1.** This is good news. From these models, I would select Logistic Regression and SVC as the best-performing. They are both performing much better in terms of recall for stroke == 1 (0.74 for both on my run) which is a huge improvement compared to the previous results. At the same time, they also perform much better in terms of recall for stroke == 1 (0.18 for log reg, 0.17 for svc on my run - still weak though) and keep a decent performance in terms of prediction for stroke == 1.

Removing outliers

First experiment with RobustScaler

Approach: We have seen some outliers in the BMI and avg_glucose_levels and we will **try with the RobustScaler, which normally handles outliers better.**

Results: Robust scaler has not improved the results.

Manual removal of outliers

Approach: I've designed a **function to replace the outliers in the 'BMI' and 'avg_glucose_level'** with a lower value based on IQR.

Results: **This experiment hasn't brought improvements to the model.**

Experiment with PCA¶

Approach: In this experiment, I have tried to **reduce the dimensionality by using PCA.**

Results: **PCA hasn't improved the results in our best-performing models.** I tried with both versions, with and without outlier handling and the outlier handling seems to contribute to better results when in combination with PCA.

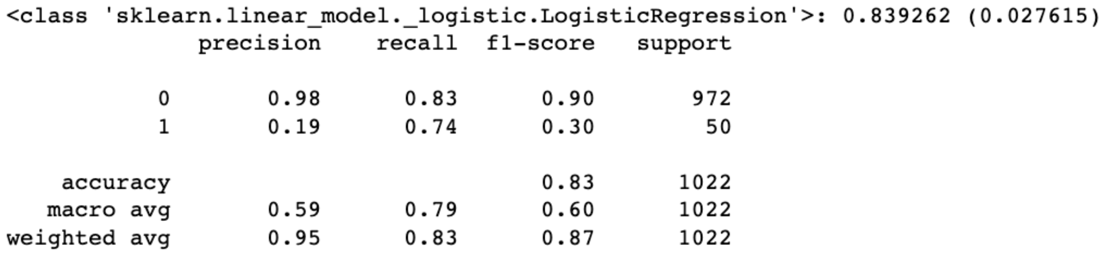
Hyperparameter finetuning on our best performant models

Approach: In this part, I will try to **finetune the hyperparameters with the use of GridSearchCV** of **the 2 best performing models: logistic regression and SVC,** which produced the best results so far.

Results: As expected, hyperparameter tuning hasn’t brought better results. In the case of SVC, the hyperparameter tuning finetunes the model towards higher accuracy, which in this case as we work with a very imbalanced dataset, means focusing on class stroke == 1 which is the majority.

Conclusions and next steps

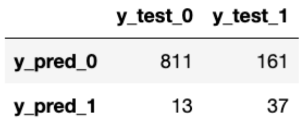
Based on the exploratory analysis, feature selection and modelling, we have achieved the best results with the following model:



```
<class 'sklearn.linear_model._logistic.LogisticRegression'>: 0.839262 (0.027615)
              precision    recall  f1-score   support

           0       0.98      0.83      0.90       972
           1       0.19      0.74      0.30        50

    accuracy                           0.83      1022
   macro avg       0.59      0.79      0.60      1022
weighted avg       0.95      0.83      0.87      1022
```



	y_test_0	y_test_1
y_pred_0	811	161
y_pred_1	13	37

The work done here is not a comprehensive and there are **further activities** that can be done:

- parameter fine-tuning the other models to see if better results can be achieved
- further experiments with balancing techniques
- dive deeper into the domain and collect other features that are assumed to have an impact on getting a stroke

The model that I have reached is still weak and **has been tuned for achieving as good predictions for stroke == 1** as possible, specifically **focusing on recall**, as I **consider the risk of false negatives (predicting no stroke when it should be yes) much higher than for false positives (predicting stroke when there is none**) as focusing on prevention and regular check-ups is not causing any harm, while not predicting a stroke has severe consequences.

My overall conclusion and so **the ultimate next step would be to collect further samples of patients with stroke == 1 as the sample given is highly imbalanced and only contains 249 samples for stroke == 1. As our goal is to predict when a person will get a stroke, this is definitely not enough to build a high-accuracy model.**