DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Stroke prediction using machine learning models

Alberto Ursino, Marco Mariotto, and Fabio Marangoni

University of Padova

1 INTRODUCTION

Strokes are a life-threatening cerebrovascular disease estimated to be the second leading cause of death in the world; worldwide statistics on mortality of this medical condition show that a stroke results in the death of the patient within a year from the event for 53% of the cases [1]. By the moment of the manifestation of symptoms of a stroke, promptness in its treatment is crucial, since it can reduce brain damage and the likelihood of death or long-term disability. The majority of stroke survivors have to deal with permanent physical impairments, inflicting challenging life changes to them and their families and causing depression in some cases. Being such a traumatic and potentially fatal condition, it is important for people to better know what a stroke is, and how it manifests itself. In particular, awareness of one's own propensity to the disease would be extremely helpful: this acknowledgement would push people to consult a doctor for getting a medical advice for proper life habits or a healthier diet, with the purpose of lowering the probability to get a stroke.

Every individual is exposed to stroke risk, but certain demographic and life-style factors can accentuate or reduce it. In this project we decided to define a binary classification problem and to base the training of some machine learning models on these features in order to accomplish the task of predicting the potential of a person to have a stroke. Then we planned to evaluate them and to compare their results, highlighting which one is the most accurate.

1.1 Related work

In the literature different aspects of stroke prediction have been investigated: from the identification of risks factors which fall into categories such as demographic, lifestyle, medical or functional to algorithms which can predict stroke risk from potentially modifiable risk factors [2]. We can find works based on ML techniques, in particular decision trees, deep neural networks and Naive Bayes predictors [3], which also make use of electronic health record databases, which clearly generalize better than our models, since these databases consist of hundred of thousands of records [4, 5]. The results from the various techniques show that multiple factors can affect the result of any study: these factors include the method of data collection, the approaches used to clean data, the handling of missing values, the standardization phase, etc.. Therefore other papers have directed their attention towards examination of relevant features through feature selection, based on the rationale that it's important to first understand how all features in a database record are related to each other, and how much they can impact on the final accuracy (in fact many studies have shown that the identification of important features can have a huge impact on the final results, in different areas of ML). In addition, prior to the development of ML algorithms for stroke prediction, an extensive research have been carried out in the alike field of heart disease prediction, with very promising results and similar approaches.

2 DATASET

Fedesoriano Stroke Prediction Dataset [6] is the dataset we used to train our models. It contains just over 5000 entries, each consisting of 10 features and the corresponding label: 1 if the subject has had a stroke, 0 otherwise.

Before using the dataset we have transformed all the entries into numerical values: this is because strings such as "Female" are not usable by our models, so in this case we have translated "Female" into a 1 and "Male" into a 0.

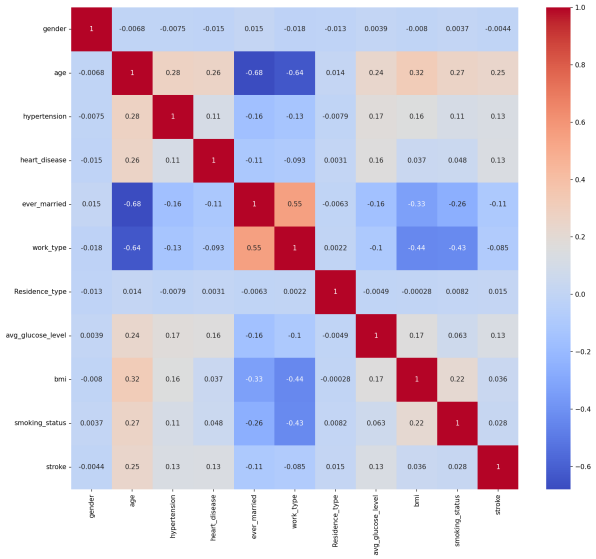

Figure 1: *Heatmap of the dataset*

After pre-processing the dataset we decided to do some preliminary analysis. As it can be seen from the last row of the heatmap graph in the img.1, the feature most correlated with "stroke" is "age", with a score of 0.25. The next most correlated features are "hypertension", "heart problems" and "average glucose level" with a score of 0.13. A positive score means having a direct proportionality: it means, for example, that as you get older you are more likely to have a stroke (see img.2).

Finally we decided to do two important operations on the dataset to make the training more consistent and accurate: an over sampling and a standardization. The over sampling is useful to enlarge the dataset and was performed thanks to the SMOTE function of Scikit-learn that implements the work presented in [7]. SMOTE basically tries to create more samples for the minority classes in order to balance a dataset; there are many different variations of this technique: the original idea consists of picking a random point on the segment between a sample point and one of its k nearest neighbors chosen at random (samples are vectors in the Euclidean space $\mathbb{R}^n$, n being the number of features). The standardization is necessary to have all the features at the same weight and was performed using the StandardScaler provided by Scikit-learn.

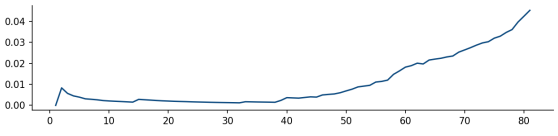

Figure 2: *Risk of having a stroke by age*

3 MODELS

3.1 Logistic Regression

Logistic regression is commonly used to predict binary variables, although it can be further generalized to handle multi labeled variables. We basically try to optimize the vector $\boldsymbol{w}$ in the following prediction formula:

$$p(\boldsymbol{x}) = \frac{1}{1+e^{-<\boldsymbol{w},\boldsymbol{x}>}}$$

so that the cross entropy between $(p(\boldsymbol{x}), 1-p(\boldsymbol{x}))$ and the actual distribution $(y, 1-y)$ (y being 1 if stroke happened to patient $\boldsymbol{x}$, 0 otherwise) on the two element space (stroke, no stroke) is minimized. It is easily seen this is equivalent to maximize the log likelihood function. This model assumes that the log of the odds ratio is equal to a linear combination of the explanatory features.

3.2 Neural Network

A neural network (NN) is a machine learning tool characterized by a brain-like structure organized into interconnected layers of nodes called "neurons", whose composition is defined by proper hyperparameters. Neural networks can be used for several purposes, one of them being the binary classification problem. More in

detail, for this task the NN is able to determine the class label for a given input data. This skill comes after a previous phase of supervised learning: adopting a sufficiently large training dataset, weights of all the links between each neuron and the next layer are iteratively adjusted in order to enable the network to recognize patterns in these data and to predict the correct output for other input data.

In this project we used a multi-layer perceptron (MLP), a particular feedforward neural network that is fully connected; its implementation was provided by the Scikit-learn library. Looking at the hyperparameters returned by grid-search cross-validation, we selected the most suitable architecture: 3 hidden layers, each one consisting of 60 neurons. The training phase started with a learning rate of 0.1 and for weights updating we ran the stochastic gradient descent (SGD) algorithm.

3.3 Random Forest

Random forest is an ensemble learning method that is widely used for different tasks, such as regression or classification, which works by creating a variety of decision trees at training time. For classification tasks, the final output is the class most selected by all trees. A decision tree is a tree structure where each leaf represents a class for the classification task, while branches represent conjunctions of features that lead through some rules (decisions) towards the leaves.

Different techniques can be employed to grow a tree: the Scikit-learn library uses an optimized version of CART algorithm. Decision trees are fairly easy to interpret, they can handle both numerical and categorical data, and prediction cost is logarithmic in the number of points used to train the tree. However, this comes at the price that learners usually tend to create over complex trees that do not generalize well in practice; this is called overfitting and can be mitigated setting a maximum depth for the tree, or the minimum number of samples at a leaf which stops the splitting phase. Decision trees also tend to be unstable when data are slighlty changed, i.e. small variations can cause the learners to create completely different trees; this problem can be mitigated using an ensamble. Another problem is that learners tend to create biased trees if some class dominates, thus it is important to balance the dataset prior to training: we did this using SMOTE oversampling technique.

To build the best model we performed grid search cross-validation on different parameters, such as maximum depth, total number of estimators, bootstrap.

3.4 Support Vector Machines

Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression and outliers detection. In our project we used this method for the classification task. One of the strengths of SVMs is that they work well with high-dimensional samples. As we have already mentioned in the dataset section 2, the samples we used are composed of a set of 10 features: this is why we decided to implement this model.

For our purpose we used a linear SVM that, given a training set of n points:

$$(\boldsymbol{x_1}, y_1), ..., (\boldsymbol{x_n}, y_n)$$

where y_i is either 1 or 0 and denotes what class the point $\boldsymbol{x_i}$ belongs to and each $\boldsymbol{x_i}$ is a vector of 10 dimensions, aims at finding the "plane with maximum margin" that divides the two sets of points $\boldsymbol{x_i}$ for which y_i is 1 from those for which y_i is 0.

To find good parameters for the model, we used the cross validation technique provided by Scikit-learn. We tested different models with different parameters affecting the regularization and the kernel of the SVM and finally took the model that performed best.

4 RESULTS

The results we obtained from the training and testing phase can be seen in table 1. At first glance, we can say that the models perform quite well and all manage to classify with an accuracy greater than 80%; the best models even exceed 90% accuracy. The accuracy is the main metric in order to estimate the performances of a model; it is calculated computing the ratio between the number of samples classified correctly and the total number of the samples on which the predictions have been made:

$$Accuracy = \frac{Number\ of\ correct\ prediction}{Total\ number\ of\ predictions}$$

From the table you can see that we have used two other parameters to analyze model performances: the precision and recall score. The precision is the ratio

$$\frac{tp}{tp+fp}$$

where tp is the number of true positives and fp the number of false positives. The precision is intuitively the ability of the classifier not to label as positive a sample that is negative. The recall is the ratio

$$\frac{tp}{tp+fn}$$

where tp is the number of true positives and fn the number of false negatives. The recall is intuitively the ability of the classifier to find all the positive samples. All models have both these parameters above 80% while the precision of the logistic regression is relatively low.

Another metric that allows us to estimate in a more compact way the performances of the various models is the Receiver Operating Characteristic (ROC) curve 3. The ROC curve is a graphical plot used to show the diagnostic ability of binary classifiers. Classifiers that give curves closer to the top-left corner indicate a better performance. As a baseline, a random classifier is expected to give points lying along the diagonal.

To compare different classifiers, it can be useful to summarize the performance of each classifier into a single measure; one common approach is to calculate the area under the ROC curve, which is abbreviated to AUC. According to this approach, the model that performs best is Random Forest with an AUC of 0.96.

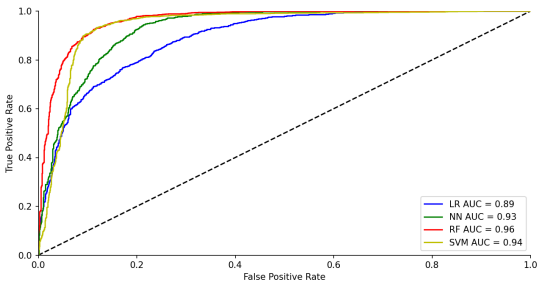

Figure 3: *Receiver operating characteristic curve*

Training scores	*Accuracy*	*Precision*	*Recall*
Logistic Regression	0.800	0.782	0.830
Neural Network	0.870	0.827	0.935
Random Forest	0.920	0.889	0.960
Support Vector Machines	0.942	0.931	0.955

Test scores	*Accuracy*	*Precision*	*Recall*
Logistic Regression	0.803	0.788	0.835
Neural Network	0.864	0.819	0.938
Random Forest	0.896	0.858	0.951
Support Vector Machine	0.901	0.868	0.949

Table 1: *Models performances on the training and test set*

5 CONCLUSIONS

As can be concluded from the satisfying results that we achieved, all the ML models that we applied to our classification task are fairly reliable and they can provide a valid stroke risk evaluation. This application finds its true usefulness in sensitizing people to their propensity to the cerebrovascular condition, indicating when it is appropriate to be concerned and to seek for medical advice.

As Terence Mills states in his article published by Forbes [8], artificial intelligence is increasingly being used in hospitals, not only to speed up simple tasks, but with new technologies also to help diagnose diseases. Always Terence Mills: ”Machine learning is changing the world of medicine for the better every day. It helps doctors diagnose more accurately, nurses admit patients more quickly, and pharmacists develop drugs more efficiently and safely.”.

All the work we have done and that has been described in this paper can be found in our repository [9].

REFERENCES

[1] Feigin, V. L. *et al.* Global, regional, and national burden of stroke and its risk factors, 1990–2019: A systematic analysis for the global burden of disease study 2019. *The Lancet Neurol.* **20**, 795–820 (2021).

[2] Min, S. N., Park, S. J., Kim, D. J., Subramaniyam, M. & Lee, K.-S. Development of an algorithm for stroke prediction: A national health insurance database study in korea. *Eur. Neurol.* **79**, 214 – 220 (2018).

[3] Jeena, R. S. & Kumar, S. Stroke prediction using svm. In *2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 600–602 (2016). DOI 10.1109/ICCICCT.2016.7988020.

[4] D., T. Towards stroke prediction using electronic health records. *BMC Med Inf. Decis Mak* (2018). DOI 10.1186/s12911-018-0702-y.

[5] CY, H., CH, L., TH, L., GS, P. & CC., L. Development of an intelligent decision support system for ischemic stroke risk assessment in a population-based electronic health record database. *PLoS One* (2019). DOI 10.1371/journal.pone.0213007.

[6] Fedesoriano. Stroke prediction dataset. URL `https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset`.

[7] Bowyer, K. W., Chawla, N. V., Hall, L. O. & Kegelmeyer, W. P. SMOTE: synthetic minority oversampling technique. *CoRR* **abs/1106.1813** (2011). URL `http://arxiv.org/abs/1106.1813.1106.1813`.

[8] Mills, T. Ai for health and hope: How machine learning is being used in hospitals (2022). URL `https://www.forbes.com/sites/forbestechcouncil/2022/02/16/ai-for-health-and-hope-how-machine-learning-is-being-used-in-hospitals/?sh=b68e1ce55beb`.

[9] Ursino, A., Mariotto, M. & Marangoni, F. Stroke prediction (2022). URL `https://github.com/d-u-d-e/stroke_prediction_ML.git`.