RESEARCH

Introduction to Focus Areas in Bioinformatics Project Week 4

Sina Glöckner[*†], Christina Kirschbaum[†], Swetha Rose Maliyakal Sebastian[†] and Gokul Thothathri[†]

[*]Correspondence:
sina.gloeckner@fu-berlin.de
Institute for Informatics, Freie Universität Berlin, Takustr. 9, Berlin, DE
Full list of author information is available at the end of the article
[†]Equal contributor

Abstract

Goal of the project: The goal of the project was to process and sample the data, apply the data to two different classifiers, and afterwards compare and evaluate the results.

Main results of the project: The sampled data sets performed better for the decision trees and the logistic regression.

Personal key learnings:
Sina: To use PySpark to read, analyze, impute, and sample data
Christina: Different imputation methods, overview of the potential of PySpark
Swetha: Fundamentals of PySpark, understood how to handle imputed data and how a decision tree classifier works
Gokul: Overview of basic packages in PySpark and regression classifier for a pipeline

Estimation of the time:
Sina: 9 hours
Christina: 6 hours
Swetha: 5 hours
Gokul: 5 hours

Project evaluation: 3

Number of words: 1227

1 Scientific Background

The death from stroke as well as the frequency of new and recurrent strokes decreased over the past decade [1]. Nonetheless, about 85% more strokes might be preventable according to new data [2]. Considering that strokes are also a main cause for disabilities, the aging population, and that through the changing lifestyle over the past decades younger people will be affected by increasing stroke rates further stresses the importance of effectively preventing strokes [3].

The major risk factor in the medical context is hypertension, which can be modified, but treatment of cardiac diseases like atrial fibrillation and carotid stenosis help to prevent strokes too. Lifestyle factors like smoking or body weight in combination with healthy nutrition and physical activity are relevant too. [1, 3]

Based on the medical report for patients, computational methods are able to predict strokes by taking all these risk factors into account, like it was shown in the McKinsey Hackathon in Healthcare.

2 Goal

This project recreated the tasks in that Hackathon. The goal was to analyze the given data, impute the missing values and sample the data. Afterwards two classifiers were applied to the original as well as the oversampled and undersampled data set. The results were evaluated and compared.

3 Data

The data is from McKinsey Stroke Dataset, which includes information about 60,001 patients. It was already split into a training set with 43,400 samples and a test set with 18,601 samples.

The data set defines 12 attributes. For every patient, an identifier, age, and sex are given. Medical details for the average glucose level, the BMI, the presence of hypertension, or heart diseases. Additionally, the data holds information about lifestyle factors such as if the patient was married, in which field he is working, whether he lives in rural or urban areas, and the smoking status. The target is the stroke value, which is encoded with 0 for no stroke and 1 for stroke.

4 Results

The data processing and the predictions were implemented with PySpark [4].

4.1 Data Exploration

First, the data was checked for missing values, which were detected in the columns BMI and smoking status. To fill in the missing values for BMI, the mean overall BMI values was computed and added. The smoking status is a categorical feature, so no mean could be calculated. A random category was chosen, in this case 'never smoked', and filled in for the missing values.

Afterwards, the data was analyzed. The distribution of all attributes was considered with the help of the histograms in Figure 1. The histograms shown were created with pandas [5] because the PySpark [4] histograms were not as legible.

Figure 1 illustrates that the data set includes more females than males and that only a small amount had hypertension, heart diseases, or a stroke. The patients mainly worked in the private sector, most of them never smoked and many stopped smoking by then, nearly twice as many have been married than not married, and an equal amount lives in the rural and urban areas. Age is normally distributed, while average glucose level and BMI have a skewed-right distribution.

For the numerical features, the boxplots in Figure 2 were generated to further look into their distribution. The boxplots for average glucose level has several outliers ranging from 160 to 300, while the outliers for BMI range from 50 to 100.

Additionally, the correlation between the three attributes was computed (Figure 3). Age and average glucose level had a coefficient of 0.24, age and BMI 0.35, average glucose level, and BMI 0.18. According to these results, there is no correlation between the numerical features.

Last, the data was sampled. The original data set contains 783 samples for patients with strokes and 42,617 for patients without stroke. With only about 1.8% strokes, the data set is quite unbalanced. Two more balanced sets were created. For one set, the majority class was undersampled to 824 and in the other set, the minority class was oversampled to 42,282.

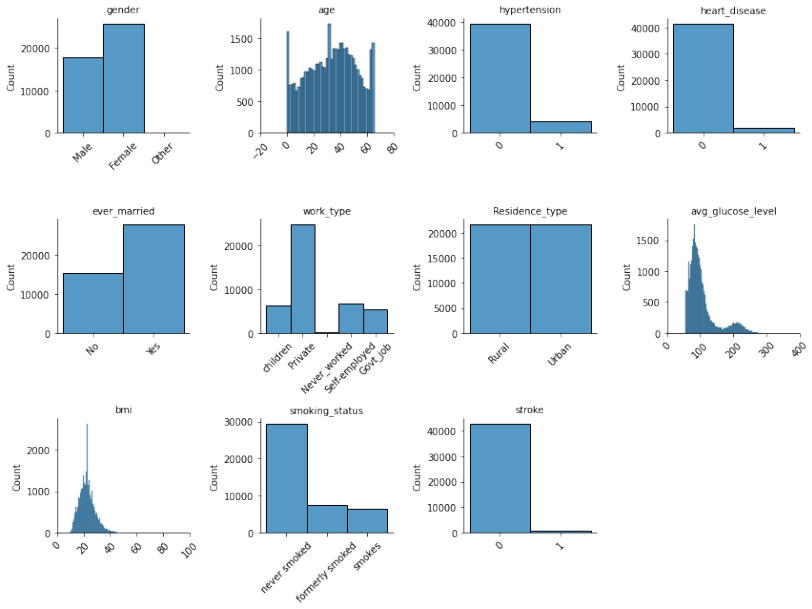


Figure 1: **Histograms for all attributes.** The histograms show the distribution of all 12 attributes given in the data set.

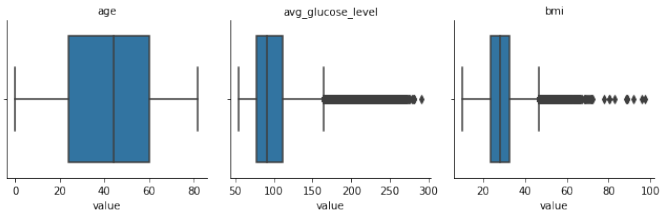


Figure 2: **Boxplots for numerical attributes.** The distribution for the three numerical attributes age, bmi and average glucose level can be seen in the boxplots.

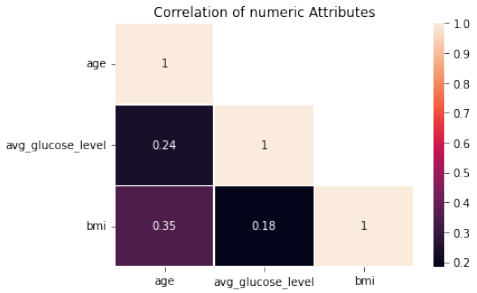


Figure 3: **Correlation of numerical attributes.** The heatmap visualizes the correlation between the three numerical attributes age, bmi and average glucose level.

4.2 Prediction

For the prediction, decision trees and logistic regression were used.

Decision trees are popular classification methods due to their ability to show the structure that leads to the decision in a tree format. The classifier constructs a tree where the leaves are labeled with values from the target class and the inner nodes hold information about the features. The children of the inner nodes hold the corresponding values of the features, so in every instance, from the root to the leaf, a decision is made depending on these. [6]

Logistic regression models the chance for an outcome based on the individual features given. The logarithm of the chance is modeled by the formula below, with π being the probability of an event, β_i being regression coefficients, x_i are the features, and n is the number of features. [7]

$$\log\frac{\pi}{1-\pi} = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n$$

The accuracy, precision, recall, and F-measure were calculated for both methods.

$$Accuracy = \frac{\Sigma \text{ True Positives} + \Sigma \text{ True Negatives}}{\Sigma \text{ Positives} + \Sigma \text{ Negatives}}$$

$$Precision = \frac{\Sigma \text{ True Positives}}{\Sigma \text{ True Positives} + \Sigma \text{ False Positive}}$$

$$Recall = \frac{\Sigma \text{ True Positives}}{\Sigma \text{ True Positives} + \Sigma \text{ False Negatives}}$$

$$F - measure = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

For the unbalanced data set, F-measure and recall obtained low values with 0.26% as well as 0.13% for decision tree and 0.00% in both cases for the logistic regression. The results significantly increased with the use of the sampled data sets, they both performed equally well for the classifiers. The detailed results of the classifiers are shown in Table 1.

Table 1: Evaluation metrics for both classifiers on the three different data sets.

Model	Dataset	Accuracy	F-Measure	Precision	Recall
Decision Tree Classifier	unbalanced	98.20%	0.26%	100.00%	0.13%
	undersampled	79.53%	79.78%	76.90%	82.89%
	oversampled	77.85%	77.63%	78.12%	77.14%
Logistic Regression	unbalanced	98.20%	0.00%	0.00%	0.00%
	undersampled	77.16%	77.30%	74.94%	79.82%
	oversampled	78.08%	78.89%	75.79%	82.25%

5 Discussion

The sampled data sets performed much better than the original data set. An accuracy of 98.20% for both models is good on the first view but it resulted from the overfitting of the models. Only the negative samples were classified correctly and

with a share of 1.8% positive samples, this results in high accuracy. The 100.00% precision of the decision tree also derives from only one true positive sample.

The sampled data sets performed well in both cases, the undersampled set being slightly better for the decision tree and the oversampled set for the logistic regression.

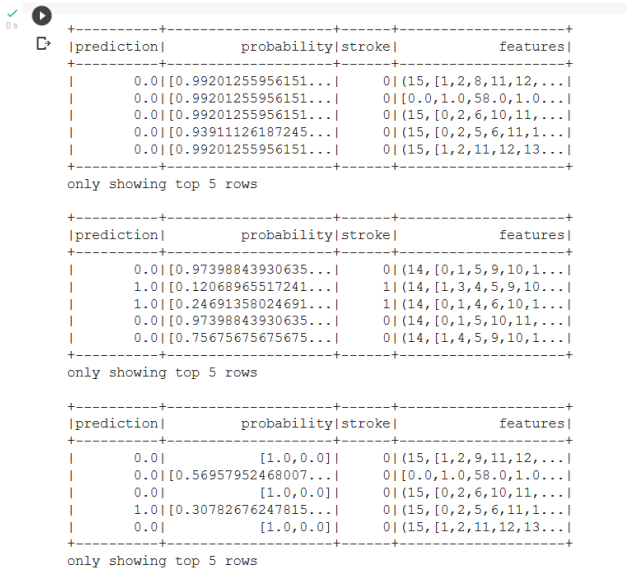
This week's project focused on the data processing and transformation tasks a data scientist encounters in his job. These were complemented with the statistics for the analysis of the data and machine learning to compare the results of the original and the sampled data sets. The project highlighted the importance of screening and preparing the data before starting to work with it.

Competing interests
The authors declare that they have no competing interests.

Author's contributions
Sina Glöckner implemented the data exploration. Swetha Rose Maliyakal Sebastian and Gokul Thothathri implemented the predictors. Christina Kirschbaum informed about the scientific background and wrote the report.

References
1. Guzik, A., Bushnell, C.: Stroke epidemiology and risk factor management. CONTINUUM: Lifelong Learning in Neurology **23**, 15–39 (2017). doi:10.1212/CON.0000000000000416
2. O'Donnell, M., Xavier, D., Liu, L., Zhang, H., Chin, S., Rao-Melacini, P., Rangarajan, S., Islam, S., Pais, P., McQueen, M.: Risk factors for ischaemic and intracerebral hemorrhagic stroke in 22 countries (the interstroke study): a case control study. The lancet **376**, 112–123 (2010)
3. Sarikaya, H., Ferro, J., Arnold, M.: Stroke prevention - medical and lifestyle measures. European neurology **73**, 150–157 (2015). doi:10.1159/000367652
4. PySpark: An Interface for Apache Spark in Python. https://spark.apache.org/docs/latest/api/python/index.html
5. Reback, J., McKinney, W., jbrockmendel, den Bossche, J.V., Augspurger, T., Cloud, P., gfyoung, Sinhrks, Hawkins, S., Roeschke, M., Klein, A., Petersen, T., Tratner, J., She, C., Ayd, W., Naveh, S., Garcia, M., Schendel, J., Hayden, A., Saxton, D., Jancauskas, V., McMaster, A., Battiston, P., Seabold, S., patrick, Dong, K., chris-b1, h-vetinari, Hoyer, S., Gorelli, M.: Pandas-dev/pandas: Pandas 1.1.5. doi:10.5281/zenodo.4309786. https://doi.org/10.5281/zenodo.4309786
6. Stiglic, G., Kocbek, S., Pernek, I., Kokol, P.: Comprehensive decision tree models in bioinformatics. PloS one **7**, 33812 (2012). doi:10.1371/journal.pone.0033812
7. Sperandei, S.: Understanding logistic regression analysis. Biochemia medica **24**, 12–8 (2014). doi:10.11613/BM.2014.003

Figures

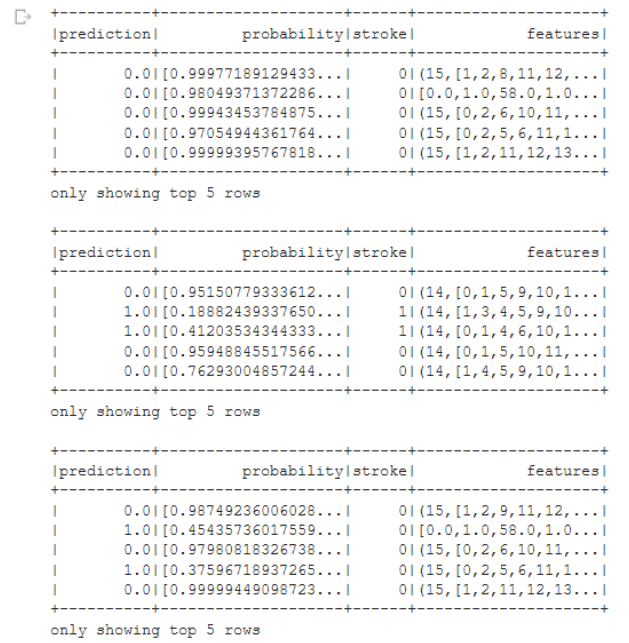


```
+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.99201255956151...|     0|(15,[1,2,8,11,12,...|
|       0.0|[0.99201255956151...|     0|[0.0,1.0,58.0,1.0...|
|       0.0|[0.99201255956151...|     0|(15,[0,2,6,10,11,...|
|       0.0|[0.93911126187245...|     0|(15,[0,2,5,6,11,1...|
|       0.0|[0.99201255956151...|     0|(15,[1,2,11,12,13...|
+----------+--------------------+------+--------------------+
only showing top 5 rows

+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.97398843930635...|     0|(14,[0,1,5,9,10,1...|
|       1.0|[0.12068965517241...|     1|(14,[1,3,4,5,9,10...|
|       1.0|[0.24691358024691...|     1|(14,[0,1,4,6,10,1...|
|       0.0|[0.97398843930635...|     0|(14,[0,1,5,10,11,...|
|       0.0|[0.75675675675675...|     0|(14,[1,4,5,9,10,1...|
+----------+--------------------+------+--------------------+
only showing top 5 rows

+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|          [1.0,0.0]|     0|(15,[1,2,9,11,12,...|
|       0.0|[0.56957952468007...|     0|[0.0,1.0,58.0,1.0...|
|       0.0|          [1.0,0.0]|     0|(15,[0,2,6,10,11,...|
|       1.0|[0.30782676247815...|     0|(15,[0,2,5,6,11,1...|
|       0.0|          [1.0,0.0]|     0|(15,[1,2,11,12,13...|
+----------+--------------------+------+--------------------+
only showing top 5 rows
```

Figure 4: **Screenshot Decision Tree.** The screenshot shows the results for the classifier decision tree with the original data (top), the undersampled data (middle) and the oversampled data (bottom).



```
+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.99977189129433...|     0|(15,[1,2,8,11,12,...|
|       0.0|[0.98049371372286...|     0|[0.0,1.0,58.0,1.0...|
|       0.0|[0.99943453784875...|     0|(15,[0,2,6,10,11,...|
|       0.0|[0.97054944361764...|     0|(15,[0,2,5,6,11,1...|
|       0.0|[0.99999395767818...|     0|(15,[1,2,11,12,13...|
+----------+--------------------+------+--------------------+
only showing top 5 rows

+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.95150779333612...|     0|(14,[0,1,5,9,10,1...|
|       1.0|[0.18882439337650...|     1|(14,[1,3,4,5,9,10...|
|       1.0|[0.41203534344333...|     1|(14,[0,1,4,6,10,1...|
|       0.0|[0.95948845517566...|     0|(14,[0,1,5,10,11,...|
|       0.0|[0.76293004857244...|     0|(14,[1,4,5,9,10,1...|
+----------+--------------------+------+--------------------+
only showing top 5 rows

+----------+--------------------+------+--------------------+
|prediction|         probability|stroke|            features|
+----------+--------------------+------+--------------------+
|       0.0|[0.98749236006028...|     0|(15,[1,2,9,11,12,...|
|       1.0|[0.45435736017559...|     0|[0.0,1.0,58.0,1.0...|
|       0.0|[0.97980818326738...|     0|(15,[0,2,6,10,11,...|
|       1.0|[0.37596718937265...|     0|(15,[0,2,5,6,11,1...|
|       0.0|[0.99999449098723...|     0|(15,[1,2,11,12,13...|
+----------+--------------------+------+--------------------+
only showing top 5 rows
```

Figure 5: **Screenshot Linear Regression.** The screenshot shows the results for the classifier linear regression with the original data (top), the undersampled data (middle) and the oversampled data (bottom).