SAKI SS 2021 Homework 1

Author: Mads Jordt, 22726829

Program code: https://github.com/madsjordt/saki21_madsjordt

Summary

The first homework in the area of machine learning required to prepare a given dataset and to train, evaluate and visualize it with the Gaussian Naive Bayes supervised learning algorithm. For this purpose, the dataset had to be preprocessed accordingly and the most important features had to be evaluated in order to classify the labels correctly. The problem was supposed to be solved by using the scikit-learn library in Python. This open source library offers efficient tools for predictive data analysis.

Dataset description and data preprocessing

The dataset itself has different characteristics. Overall, it contains eleven attributes and 209 tuples which makes it fairly small. I split the dataset into 70% training and 30% testing data to train and test the final classifier accordingly. Taking less testing samples might lead to overfitting due to the limited size of the dataset which should be avoided.

I concatenated the relevant features (Buchungstext, Verwendungszweck, Beguenstigter/Zahlungspflichtiger) to one unified string since they contain most information for predicting the labels. Other attributes such as BLZ, Betrag, and Waehrung are not that important due to their limited informative value. After also applying other feature combinations to the model, I came to the conclusion that the three relevant features just mentioned above improved the accuracy the most. Moreover, for preprocessing I converted the transaction data into a matrix of token counts. In addition to that, all characters were converted to lowercase before tokenizing. Also, I decided to remove german stopwords which is why I had to handle umlauts and special characters. Additionally, I generated n-grams and general character normalization was performed in order to remove noise and improve performance.

Classifier and performance scores

I applied the Gaussian Naive Bayes algorithm since it works generally well for classification problems due to its support for continuous valued features. Moreover, I implemented a pipeline that assembles several steps including data preprocessing and matrix transformation so that they can be cross-validated accordingly. For cross-validation, I performed 10 iterations to be able to compare the performance outcomes and eventually to detect outliers. The rather equal performance distribution without any major outliers confirmed my model's accuracy.

Finally, a classification report and a confusion matrix heatmap are plotted to visualize the classifier's performance. More information will follow in the evaluation section.

Evaluation

The main results are displayed in the classification report in the screenshot section. It shows that the Gaussian Naive Bayes classifier performed very well with an overall accuracy of 0.92 (92%). The f1-score indicates the mean of precision and recall. Thus, the number of false positives is very low compared to the number of correctly predicted

true positives.. Generally, the classifier performed well for all six labels. The accuracy was below 0.90 only for the labels leisure and living, but above 0.90 for all four other labels. One reason for this might be an imbalance in the dataset when it comes to label distribution. However, this is not that bad due to the good performance in general.

A graphical illustration of the corresponding metrics results can be found in the confusion matrix screenshot. The heatmap provides an overview over the distribution of the actual labels and the predicted labels. The predicted testing labels are colored depending on their occurence. The labels leisure and standardOfLiving have the highest support, while the support for the other labels is rather equally distributed.

Besides Gaussian Naive Bayes, I also applied the Multinomial Naive Bayes algorithm to have a comparison. I used Gaussian Naive Bayes in my final solution because this homework expected the implementation of this specific algorithm. However, the Multinomial Naive Bayes would also have been a good choice as the transaction data are discrete numbers and hence not necessarily normally distributed.

Screenshot

Resulting metrics: classification report and confusion matrix heatmap.

	precision	recall	f1-score	support
finance	1.00	0.88	0.93	8
income	1.00	1.00	1.00	5
leisure	0.82	0.93	0.87	15
living	0.90	0.82	0.86	11
private	0.88	1.00	0.93	7
standardOfLiving	1.00	0.94	0.97	17
accuracy			0.92	63
macro avg	0.93	0.93	0.93	63
weighted avg	0.93	0.92	0.92	63