Anomaly detection on a letter recognition dataset

Daniela Trenzinger, Mea-Maria Leinonen, Florentina Hager
Knowledge Discovery and Data Mining 2

Problem understanding

This project's objective is to test, develop and compare algorithms for anomaly detection. In general, three different types of anomaly detection problems can be distinguished: fully labelled data, labelled normal data and unlabelled data.

In our project we used the original letter recognition set[1,2]. This dataset consists of 20,000 single data points, which were artificially created by combining 20 different fonts. Basically, each stimulus presents a letter on a 45 x 45 pixel image, and each pixel can have 2 states: "on" or "off". In the next step, each image was converted into 16 different features, such as the letter's **width** and **height**, or **the average number of edges**.

For the evaluation we artificially created 4.16% outliers, which are 3 standard deviations away from the mean. However, we addressed the data as unlabelled and only used the labels for evaluation.

Data inspection

We first started by doing the data inspection, including:

- Check for missing values and incorrect data types
- Statistical description of each feature in the dataset
- Correlation matrix
- Tests for gaussian distribution (kolmogorov-smirnov, histograms)

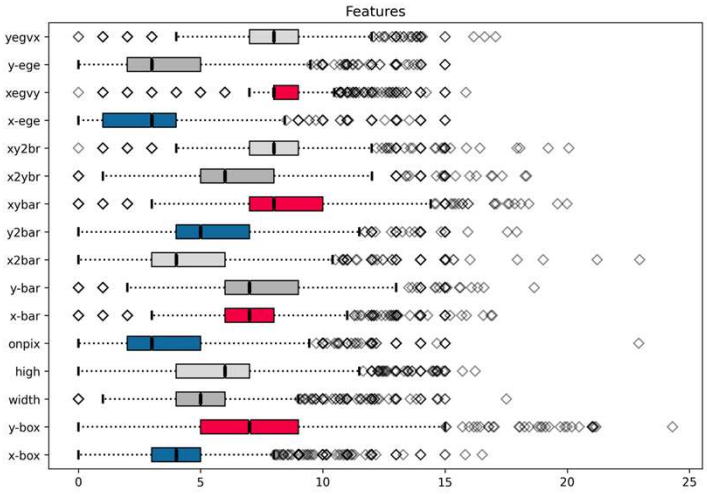

Figure 1: Feature inspection of the outlier dataset

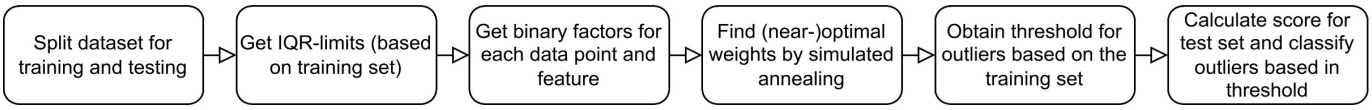

Figure 2: Description of the implementation of the own algorithm

Own algorithm

Our own algorithm is based on the idea to score data points according to their entries for the features and then split the dataset into normal data and outliers. We separate the group at the largest gap between scores and try to maximize this gap by applying simulated annealing. Our algorithm takes the following hyperparameters:

- *IQR-Factor*: factor to distinguish outliers from normal data for individual features
- *T*: starting temperature for simulated annealing
- *α*: decreasing rate for T
- *iter*: number of iterations after which the temperature is updated

Results

We obtain varying results for the different algorithms. For the Local Outlier Probability, as well as our own algorithm and the Isolation Forest, we used hyperparameter testing an chose the best performing parameters. The Local Outlier Probability performs best with a F1 score of 0.77, followed by the DBSCAN (0.30), our own algorithm (0.10) and the Isolation Forest (0.08).

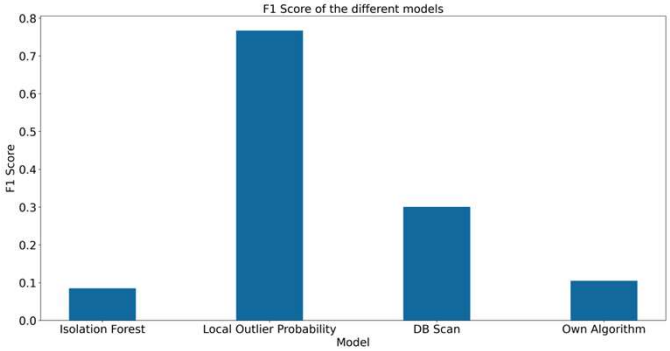

Figure 3: This chart shows the evaluation of the different algorithms. The own algorithm is on rank 3

Approach

We used 3 existing algorithms for unlabelled data:

- Local outlier probability
- Isolation forest
- DBSCAN

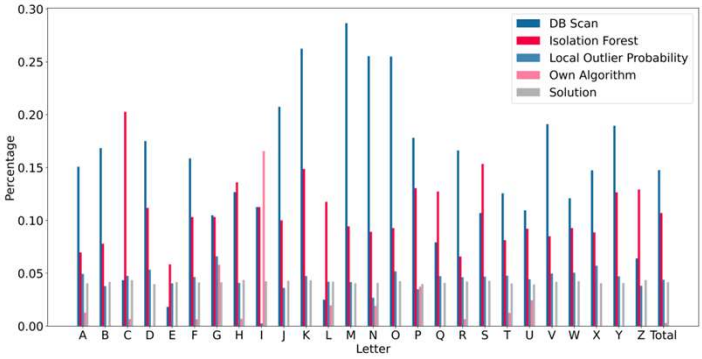

Figure 4: Results per algorithm and letter

Conclusion & Limitations

In our project we implemented three existing algorithms and presented a new algorithm to the letter dataset. We obtained varying results across the different algorithms, indicating that not only the hyperparameters but also the choice of the underlying algorithm is crucial for the quality of the results. For our own algorithm, we are planning the following steps:

- Improving the determination of the largest gap
- More hyperparameter tuning to balance the quality and the runtime
- Application to different datasets

Literature

[1] Frey, P. W., & Slate, D. J. (1991). Letter recognition using Holland-style adaptive classifiers. *Machine learning*, *6*(2), 161-182.
[2] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

TU Graz – *Institute of Interactive Systems and Data Science*
8010 Graz, Inffeldgasse 13, Austria, Tel.: +43 316 873-0000
https://www.tugraz.at/institute/isds/home