Tabular Data Science Final Project - **Outliers Detection**

Michael Ternovsky
Roei Ben Zeev

20 March 2022

1 Introduction

The problem we aimed to solve is outliers' detection. Our solution is to detect those outliers that their deviate by 2 standard deviations from the mean or the median. Our experiment was successful, and showed significantly more beneficial learning over data sets with high number of numerical features, between those with outliers' removal and those without.

2 Problem Description

In data analysis, anomaly detection (also referred as "outlier detection" and sometimes "novelty detection") is generally understood to be the identification of rare items, events or observations which deviate significantly from the majority of the data and do not conform to a well-defined notion of normal behavior. Such examples may arouse suspicions of being generated by a different mechanism, or appear inconsistent with the remainder of that set of data.

Outlier detection is often an important step in data pre-processing to provide the learning algorithm a proper dataset to learn on. This is also known as Data cleansing. After detecting anomalous samples classifiers remove them, however, at times corrupted data can still provide useful samples for learning. A common method for finding appropriate samples to use is identifying Noisy data.

3 Solution overview

As mentioned, sometimes a dataset can contain extreme values that are exceed the expected range and are unlikely belong to the dataset, or in our paper, out-

liers. In the field of machine learning modeling and in model skill in general it is important to understand and even sometimes to remove these outlier values, in order to improve their well-suited learning.
The goal in our project is to identify and remove the outliers from our dataset, which is the data over which we perform and trained our machine learning algorithm.
We would like to automate the data pipeline and make a pre-processing mechanism to handle outliers and notify the user that those samples can affect the model result and distribution.
Here we detect the outliers using two different methods and simply remove them from the dataset, although removal is not necessarily the only or the best way to deal with outliers, it's most definitely the simplest one, and therefore we chose it.
the methods are:

•Removal of all samples that deviated by at least two standard-deviations from the mean of all samples.

•Removal of all samples that deviated by at least two standard-deviations from the median of all samples.

4 Experimental plan

We tried different methods to identify the outliers and checked each method in order to discover the method over which our machine learning algorithm will provide the best results over data set with high number of numerical features

We evaluated our results based on comparison between the model result to the outlier samples and the results after cleaning the outliers with the two different methods.

We split our data into a training set and test set. on the same training set, we ran the two different methods to remove outliers, and ran the examined the two different resulting models over the test set. We did the same with the full training set, without any outliers' removal, and determined which technique has given the best learning results.

Then, we compared the result of the experiment to sklearn library outliers handling function (e.g IsolationForest, EllipticEnvelope, LocalOutlierFactor) and determine whether one of our approaches and methods has given us a better result.

5 Experimental evaluation

As mentioned above, we compared the results by running the same model on the data set after we removed the outliers using two different methods, and ran the model on different sklearn outliers' detection methods. we noticed that for data set with a relatively large number of numerical features, we achieved an outstanding result as we can see in the comparison we made on the house pricing dataset that which contains 36 numerical features.

	Method	Model's Performances
1	STD Median	0.978315
0	STD Mean	0.977289
2	EllipticEnvelopen	0.912819
5	None	0.911485
3	LocalOutlierFactor	0.909727
4	IsolationFactor	0.908610

As we can see above the models used our two methods achieved a significantly better result than other sklearn outlier detection methods and, unsurprisingly compared to the same dataset without any outlier detection method.

Furthermore, as can be seen in the images attached in the project folder, the difference in the feature distribution field, we can clearly notice that the removal of the outliers and by resulted in a much more accurate data with less "noisy" data that can affect our data set and learning model. we can see that most of the data from the "edges" has deleted meaning we left most of the data around the "center" of our data set distribution.

As for data sets with a quite smaller number of numerical features. As we can see in the car-prices data set below. our methods gave us a better result than not using any outlier removal method however other sklearn methods of outliers removal has ended up giving a better result.

	Method	Model's Performances
4	IsolationFactor	0.965940
3	LocalOutlierFactor	0.826687
2	EllipticEnvelopen	0.817243
1	STD Median	0.767888
0	STD Mean	0.766859
5	None	0.743349

6 Related work

In this article we can see an explanation of Global outliers. As mentioned there, a data point is considered a global outlier if its value deviates the entirety of the data in which it is found:
"A global outlier is a measured sample point that has a very high or a very low value relative to all the values in a dataset. For example, if 99 out of 100 points have values between 300 and 400, but the 100th point has a value of 750, the 100th point may be a global outlier."

According to this, the outliers we detect using our two proposed methods are considered "global outliers". we measured out outliers which deviated by two standard deviations from the mean or the median of the features.

The article explains the idea of global outliers with visualization examples. We improved the technique to an automated mechanism which detects the outliers and removes them.

7 Conclusion

In conclusion, we noticed that in the case where our data contains a large number of numerical features our method of outliers' detection and removal improves the model results as can be seen in the histograms in the attached images, we can notice that the removal of the outliers has improved the feature data distribution and resulted in a better result than Sklearn outliers detection methods. In data sets with a small number of numerical features we notice our method did not affect the model results in a more significant way than not using it but some other sklearn outlier detection methods. Furthermore, other outliers' detection

methods were better and had better results. According to the data above we can conclude that our methods were very good, and did not harm the results adversely but improved them.

We had a great time implementing this project idea, we learned a lot about the effect of data preprocessing and how it can improve our results. Furthermore, we saw how it can improve our data distribution.

Thanks for the semester, Dr. Somech, and we wish you all the best.