Network Attack Outlier/Anomaly Detection

BY: Matan – Ben Nagar, ID: 206240301

Data exploration- what have you learned?

The simplest approach to identifying irregularities in data is to flag the data points that deviate from common statistical properties of a distribution, including mean, median, mode, and quantiles.
In statistics, outliers are data points that don't belong to a certain population. It is an abnormal observation that lies far away from other values. An outlier is an observation that diverges from otherwise well-structured data. For Example, you can clearly see the outlier in this list: [20,24,22,19,29,18,**4300**,30,18]

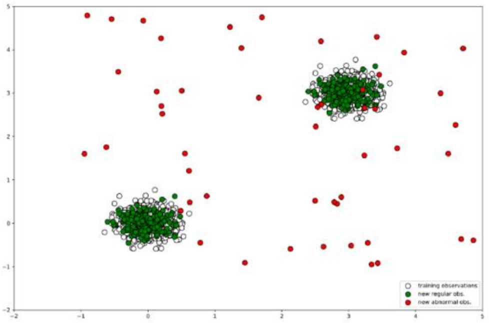
Algorithms Group that are suitable for this task and why:

- **Standard Deviation:** if you have any data point that is more than 3 times the standard deviation, then those points are very likely to be anomalous or outliers.
- **Boxplots:** Box plots are a graphical depiction of numerical data through their quantiles. It is a very simple but effective way to visualize outliers.
  https://www.youtube.com/watch?v=mhaGAaL6Abw&ab_channel=TheOrganicChemistryTutor
- **Clustering Algorithms:**
  - DBScan Clustering: The downside with this method is that the higher the dimension, the less accurate it becomes. You also need to make a few assumptions like estimating the right value for eps which can be challenging.
- **Isolation Forest:** This approach is different from all previous methods. All the previous ones were trying to find the normal region of the data then identifies anything outside of this defined region to be an outlier or anomalous. This method works differently. It explicitly isolates anomalies instead of profiling and constructing normal points and regions by assigning a score to each data point.

Please create a report that will explain how you solved the problem.

What is the approach you tried? Why them?

Since I am new to the world of data-sciense, prediction algorithms, deep-learning, etc… I mostly found myself roaming the web looking for new approaches as to how to detect anomalies in a given set of information. I read statistical articles, looked through github repositories of Anomaly-Detetction projects, watched youTube videos and so on…



I finally narrowed my testing to a specific algorithm:

<u>Isolation Forest</u> who seemed very promising, easy to use and comprehend. In principle, outliers are less frequent than regular observations and are different from them in terms of values (they lie further away from the regular observations in the feature space). Isolation Forest explicitly identifies anomalies instead of profiling normal data points.

After deciding to test out the Isolation algorithm, there was the matter of understanding how the parameters effect the results. In my jupyter notebook I left some of my testing. For exmaple I wanted to see the influence of the paramter "n_estimator" which refers to the number of base estimators or trees in the ensemble, i.e. the number of trees that will get built in the forest… My results showed me that the number of n_estimators didn't make the result change by much.

I than tested the variable "contamination" : With isolation forest we had to deal with the contamination parameter, which sets the percentage of points in our data to be anomalous.

But how would I decide if the results were good or bad? That I answer in the next paragrph.

How do you know the algorithm is good?

Other than the fact that at the end I compared my results with the csv answer file that you provided us , I mostly got to the conclusion that the algorithm works well based on the 2D plot. Contamination is an important parameter in the Isolation Tree Algorithm and I have arrived at its value based on trial and error on validating its results with outliers in 2D plot. Again, I had to scan the web to figure out how to visualize the dataset that had been given to me. Through this I was able to estimate when the algorithm performed okay and when it was way-off.

Conclusions

Detecting anomalies in a given dataset is not an easy task. Being able to determine "how much" a deviation from the majority that is "good", is a tricky thing. Picking the wrong parameters might cause you to classify a good behaviour as bad. It is important to understand the concept of mean, median and the deviation from "normal" using mathematical terms. Testing and comparing different algorithms is also an important step since Deep Learning is never 100% accurate.

References & Links

1. https://towardsdatascience.com/5-ways-to-detect-outliers-that-every-data-scientist-should-know-python-code-70a54335a623
2. https://www.youtube.com/watch?v=5p8B2Ikcw-k&ab_channel=PyData
3. https://www.youtube.com/watch?v=TP3wdwD8JVY&t=632s&ab_channel=DecisionForest
4. https://towardsdatascience.com/outlier-detection-with-isolation-forest-3d190448d45e