FINAL TASK FOR ANALISI DEI DATI – 2023/24

1. Introduction

Anomaly detection broadly refers to the process of identifying outliers in data. Essentially outliers are the data points that stand out among all other data points, and whose values are somewhat unexpected and deviate significantly from the standard patterns. The identification of outliers has traditionally been aimed at obtaining a more faithful representation of the statistical properties of the data, and consequently at improving the effectiveness of methods built on such data. Recent applications in anomaly detection are for instance developed in predictive maintenance in manufacturing facilities, to prevent incidents and reduce downtime cost. The aim is to detect any deviations from the equipment's normal performance long before the machine fails.

Sometimes, in certain families of models, the detection task can be easily accomplished through theoretical results. For instance, in the context of linear regression, there is the well established notion of *leverage*, which is a measure of how far away the values of a data point are from those of the others. This is accomplished by computing the degree to which the model coefficients would change when removing a single data point. As it is usually the case in linear regression, it is not necessary to fit as many models as the number of data points, because leverage can be computed by a suitable manipulation of the design matrix. In general, this method may prove impractical also, but not only, because of the computational cost.

2. Goals

Your task is to explore possible ways of developing methods of anomaly detection in classification problems. You are not restricted to using a single classification method, and you can, in principle, develop different assessments of anomaly for different algorithms. When developing your method(s), try to consider

- the computational complexity of your method,
- that the notion of anomaly strongly depend on the nature of the data analyzed.
- above all, that the results should be as much informative as possible and able to identify not only *which* data points are outliers, but also *why* data points are outliers.

Date: June 19, 2024.

3. Data

Data provided are of different kind and we strongly encourage to choose only one of the datasets as the subject of your analysis. All datasets are available in the following shared folder:

https://drive.google.com/drive/folders/1mIJKT6sDKi036YsmEnYNog-fc_9PAs01

3.1. **MNIST.** The first dataset (`mnist.csv`) has been obtained by the original MNIST dataset of handwritten digits by sampling at random 5000 digit-zero (considered as inliers) and 500 digit-six (considered as outliers). Moreover, only 150 out a total of 784 features have been randomly selected.

3.2. **Power Demand.** The second dataset (`ipd.csv`) records the hourly electrical power demand of a small Italian city for about three years.

3.3. **Network intrusions.** The third dataset records LAN traffic, and contains connections with malicious intents, called intrusions or attacks, and normal connections. There are two dataset, one for training the model (`net_train.csv`), and one for testing the model (`net_test.csv`). Please be aware that the test set contains intrusions which are not present in the train set. This is intentional.

4. Submission

The results of your own analysis and ideas must be summarised in a report which explains how you have planned to tackle the problem and the possible strategies you have tried to solve the problem. The emphasis is not on the performances of the final method(s) proposed, but on the way you have dealt with the problem.

You are not only allowed but actually encouraged to read up on the subject. In order to be complete and fair, you are required to cite all sources of research material you have used (books, scientific papers, etc.).

This final assignment is a personal piece of work and must not be done in groups. Discussions with colleagues or experts, although discouraged, should be reported for fairness.

Your report can be uploaded on the *e-learning* website. The deadline is by **August 25, 2024** (but early submissions are appreciated). If, for some reason, you need to complete the exam before the scheduled date, for instance for degree completion or as an Erasmus student, please contact us.

You should add, at the end of your report, the link to a script (R or Python, or any other programming language of your choice) containing the implementation of the *final* method(s) proposed, based on the analysis developed. The script must be shared via a *notebook* on Google Colab. Obviously the script must not contain any errors. Please add a link to the notebook in your report.

It is not necessary (and in fact useless) for the script to contain the entire analysis. The recommendation is that the output of your scripts will be a

detailed account of your conclusions. The numbers, without any explanation about their meaning, are not really helpful.

Appendix A. Details on the network intrusion dataset

feature name	description
duration	length (number of seconds) of the connection
protocol_type	type of the protocol, e.g. tcp, udp, etc.
service	network service on the destination, e.g., http, telnet, etc.
src_bytes	number of data bytes from source to destination
dst_bytes	number of data bytes from destination to source
flag	normal or error status of the connection
land	1 if connection is from/to the same host/port; 0 otherwise
wrong_fragment	number of "wrong" fragments
urgent	number of urgent packets

Table 1. Network intrusions dataset: basic features of individual TCP connections.

feature name	description
hot	number of "hot" indicators
num_failed_logins	number of failed login attempts
logged_in	1 if successfully logged in; 0 otherwise
num_compromised	number of "compromised" conditions
root_shell	1 if root shell is obtained; 0 otherwise
su_attempted	1 if "su root" command attempted; 0 otherwise
num_root	number of "root" accesses
num_file_creations	number of file creation operations
num_shells	number of shell prompts
num_access_files	number of operations on access control files
num_outbound_cmds	number of outbound commands in an ftp session
is_hot_login	1 if the login belongs to the "hot" list; 0 otherwise
is_guest_login	1 if the login is a "guest"login; 0 otherwise

Table 2. Network intrusions dataset: content features within a connection suggested by domain knowledge.

feature name	description
count	nr. of connections to the same host as the current one in the past 2"
serror_rate[1]	% of connections that have "SYN" errors
rerror_rate[1]	% of connections that have "REJ" errors
same_srv_rate[1]	% of connections to the same service
diff_srv_rate[1]	% of connections to different services
srv_count[1]	nr. of connections to the same service as the current one in the past 2"
srv_serror_rate[2]	% of connections that have "SYN" errors
srv_rerror_rate[2]	% of connections that have "REJ" errors
srv_diff_host_rate[2]	% of connections to different hosts

Table 3. Network intrusions dataset: traffic features computed using a two-second time window.

[1] These features refer to these same-host connections.

[2] These features refer to these same-service connections.