# Projet BDA : Anomaly Detection in Network Traffic with K-means Clustering

## IntroductionBDA\_Hochet\_Kopp\_Silvestri

Dans le cadre du cours Big Data Analytics, un projet d’apprentissage est réalisé. La technologie utilisée pour sa réalisation est SparkML. Ce document présente la description du projet ainsi que les résultats obtenus.

## Partie 1 : Description du dataset

Le dataset contient un ensemble de 4,9 millions de connexions réseau, et à été généré par rapport à des données de 1999. Les informations contenues ont été prétraitées, afin d’avoir uniquement un résumé de celle-ci (un paquet réseau peut avoir une structure complexe et pas forcément toujours uniforme) afin d’obtenir des données facilement utilisables dans un contexte de machine learning. Au final, la taille totale du dataset est de 708 MB.

Chaque entrée du dataset est représentée au format csv, et contient 41 features, comme par exemple les protocoles des différentes couches OSI (tcp/udp, http, ftp, …), ou encore la quantité de données échangées. La plupart des features correspondent à des compteurs, ou à une valeur binaire (0 ou 1). Il existe également certaines features correspondant à un ratio, et donc représenté avec des valeurs décimales comprises entre 0 et 1.

Enfin, la dernière feature est un label, catégorisant le type d’attaque correspondant à la connexion. Dans notre cas, l’apprentissage est non supervisé, et donc cette valeur sera très certainement ignorée.

Une description plus détaillée des features est présentée dans un document en ligne[[1]](#footnote-1) :

|  |  |  |
| --- | --- | --- |
| *feature name* | *description* | *type* |
| duration | length (number of seconds) of the connection | continuous |
| protocol\_type | type of the protocol, e.g. tcp, udp, etc. | discrete |
| service | network service on the destination, e.g., http, telnet, etc. | discrete |
| flag | normal or error status of the connection | discrete |
| src\_bytes | number of data bytes from source to destination | continuous |
| dst\_bytes | number of data bytes from destination to source | continuous |
| land | 1 if connection is from/to the same host/port; 0 otherwise | discrete |
| wrong\_fragment | number of ``wrong'' fragments | continuous |
| urgent | number of urgent packets | continuous |
| hot | number of ``hot'' indicators | continuous |
| num\_failed\_logins | number of failed login attempts | continuous |
| logged\_in | 1 if successfully logged in; 0 otherwise | discrete |
| num\_compromised | number of ``compromised'' conditions | continuous |
| root\_shell | 1 if root shell is obtained; 0 otherwise | discrete |
| su\_attempted | 1 if ``su root'' command attempted; 0 otherwise | discrete |
| num\_root | number of ``root'' accesses | continuous |
| num\_file\_creations | number of file creation operations | continuous |
| num\_shells | number of shell prompts | continuous |
| num\_access\_files | number of operations on access control files | continuous |
| num\_outbound\_cmds | number of outbound commands in an ftp session | continuous |
| is\_host\_login | 1 if the login belongs to the ``hot'' list; 0 otherwise | discrete |
| is\_guest\_login | 1 if the login is a ``guest''login; 0 otherwise | discrete |
| count | number of connections to the same host as the current connection in the past two seconds | continuous |
| srv\_count | number of connections to the same service as the current connection in the past two seconds | continuous |
| serror\_rate | % of connections that have ``SYN'' errors | continuous |
| srv\_serror\_rate | % of connections that have ``SYN'' errors | continuous |
| rerror\_rate | % of connections that have ``REJ'' errors | continuous |
| srv\_rerror\_rate | % of connections that have ``REJ'' errors | continuous |
| same\_srv\_rate | % of connections to the same service | continuous |
| diff\_srv\_rate | % of connections to different services | continuous |
| srv\_diff\_host\_rate | % of connections to different hosts | continuous |
| dst\_host\_count | Count of connection having same dest hot | continuous |
| dst\_host\_srv\_count | Count of connection having the same destination host and using same service | continuous |
| dst\_host\_same\_srv\_rate | % of connection having the same destination host and using same service | continuous |
| dst\_host\_diff\_srv\_rate | % of different service on the current host | continuous |
| dst\_host\_same\_src\_port\_rate | % of connection to the current host having same src port | continuous |
| dst\_host\_srv\_diff\_host\_rate | % of connection to the same service coming form different host | continuous |
| dst\_host\_serror\_rate | % of connection to the current host that have ``SYN'' error | continuous |
| dst\_host\_srv\_serror\_rate | % of connection to the current host and specified service that have ``SYN'' error | continuous |
| dst\_host\_rerror\_rate | % of connection to the current host that have ``REJ'' error | continuous |
| dst\_host\_srv\_rerror\_rate | % of connection to the current host and specified service that have ``REJ'' error | continuous |

Voici enfin un exemple de connexions normale :

***0,tcp,http,SF,181,5450,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.00,0.00,9,9,1.00,0.00,0.11,0.00,0.00,0.00,0.00,0.00,normal.***

Ici on a l’exemple d’une connexion http, avec 181 bytes envoyé (la requête GET/POST) et 5450 (le contenu de la page). On peut aussi voir que les pourcentage d'erreurs valent 0.

Puis une connexion catégorisée comme une attaque (ici l’attaque smurf, une variante de DDOS) :

***0,icmp,ecr\_i,SF,1032,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,smurf.***

Dans ce cas le protocole est icmp, et il y a 0 byte dans le réponse (le principe de l’attaque smurf est d'envoyer un ping avec l’ip de la victime comme source, de façon à ce que le serveur réponde directement vers la victime). On peut voir ici un grand nombre de connexions au même port (255 connexions), qui nous montre déjà une certaine anomalie (la requête précédente n’avait que 9 connexions).

## 

## Partie 3 : Questions d’analyse

Cette première approche du projet soulève certaines questions, dont nous espérons pouvoir obtenir des réponses à la fin de ce projet.

* Est-ce que l’utilisation de toutes les features disponibles donne de meilleurs résultats ? -> Quelle ensemble de features donnent les meilleurs résultats ? Est-ce que certaines ne sont pas nécessaires au clustering ?
* Est-ce que l’algorithme K-mean est l’algorithme le plus adapté à notre problème ? -> Y-a-t-il des algorithmes de clustering plus adaptés que K-means pour le problème ?
* Est-il possible d’identifier les différentes attaques au moyen d’un algorithme non supervisé ou une distinction “normal/anormal” est-elle la seule possible ?

1. <http://kdd.ics.uci.edu/databases/kddcup99/task.html> [↑](#footnote-ref-1)