Revue RAMReS – Sciences Appliquées et de l'Ingénieur
RAMReS 2023 - *Published Online 2023* - http://publication.lecames.org/
Vol. 5(1), pp. 91-95, Online December 2023

Vol. 5
ISSN 2630-1164
December 2023

Full Length Research Paper

Modélisation de l'identification des familles de vulnérabilités logicielles par l'Intelligence Artificielle à partir des données de 2021 de CVEdetails.com

Doffou Jérôme DIAKO[1*], Melaine Odilon ACHIEPO [2]
[1]ESATIC, LASTIC, LARIT – Abidjan, Cote d'ivoire
[2]UVCI, UREN– Abidjan, Cote d'ivoire

Received July 2023 – Accepted October 2023

**Corresponding author. Jerome.diako@esatic.edu.ci*

Résumé:
De nos jours, la prolifération des vulnérabilités logicielles représente une menace grandissante pour les systèmes informatiques des entreprises, des organisations gouvernementales et des agences. Dans cet article, nous avons proposé une approche novatrice permettant de regrouper les potentielles vulnérabilités logicielles qui pourraient se propager à l'avenir. Pour atteindre cet objectif, nous avons développé une méthodologie innovante en combinant l'analyse des correspondances multiples (MCA), la procédure Elbow et l'algorithme Kmeans. Pour valider notre approche, nous avons procédé à une simulation basée sur un ensemble de données conséquentes comprenant 20153 observations. Les résultats de cette simulation nous ont permis d'identifier et de classifier différentes familles de vulnérabilités futures. Afin de mesurer l'efficacité de notre modèle, nous avons utilisé l'indice de silhouette, fournissant ainsi une évaluation de leur approche. En somme, cette recherche met en lumière une méthode prometteuse pour anticiper et mieux comprendre les vulnérabilités logicielles émergentes, ouvrant la voie à des mesures préventives plus efficaces pour protéger les systèmes informatiques des menaces potentielles.

Mots clés/Keyword: **Vulnérabilités, ACM, Apprentissage non supervisé, Kmeans, CVSS,**

Cite this article:
Doffou Jérôme DIAKO, Melaine Odilon ACHIEPO (2023). Modélisation de l'identification des familles de vulnérabilités logicielles par l'Intelligence Artificielle à partir des données de 2021 de CVEdetails.com. Revue RAMReS – Sci. Appl. & de l'Ing., Vol. 5(1), pp. 91-95. ISSN 2630-1164.

1. Introduction

De nos jours, l'omniprésence de la technologie numérique a transformé le monde en un vaste écosystème interconnecté.

Les entreprises, les organisations gouvernementales et les agences dépendent désormais largement des systèmes informatiques pour mener à bien leurs activités essentielles [1]. Cependant, cette dépendance s'accompagne également d'une menace croissante : les vulnérabilités logicielles.

Les vulnérabilités logicielles, ou failles de sécurité, sont des défauts ou des faiblesses présentes dans les logiciels qui peuvent être exploitées par des acteurs malveillants pour accéder, perturber ou endommager les systèmes informatiques [2]. Ces vulnérabilités peuvent entraîner des conséquences désastreuses, telles que des pertes financières considérables, la divulgation d'informations sensibles ou encore des atteintes à la réputation des organisations.

Malheureusement, le nombre de vulnérabilités logicielles ne cesse d'augmenter, rendant la tâche de protection des systèmes informatiques de plus en plus complexe [3]. Il est donc crucial de développer des méthodes avancées pour anticiper et contrer ces menaces émergentes.

Dans cette optique, le présent article se concentre sur la proposition d'une approche novatrice visant à regrouper les futures vulnérabilités logicielles susceptibles de se propager. Pour ce faire, nous combinons l'analyse des correspondances multiples (ACM), la procédure Elbow et l'algorithme Kmeans pour identifier et classifier les différentes familles de vulnérabilités.

L'objectif principal de cet article est de fournir aux professionnels de la sécurité informatique un outil puissant pour mieux comprendre les vulnérabilités émergentes et développer des mesures préventives plus ciblées et efficaces. En utilisant une approche basée sur des techniques statistiques avancées et une évaluation présentée à l'aide de l'indice de silhouette, nous contribuerons à renforcer la résilience des systèmes informatiques face à ces menaces croissantes.

Dans la suite de cet article, nous présentons en détail notre méthodologie, en décrivant les étapes clés de l'analyse des correspondances multiples (ACM), de la procédure Elbow et de l'algorithme Kmeans. Nous détaillerons également le jeu de données utilisé pour la simulation et présenterons les résultats obtenus, mettant en évidence les familles de vulnérabilités futures.

2. Analyse des correspondances multiples (ACM)

L'Analyse des Correspondances Multiples (ACM) est une technique statistique multivariée utilisée pour analyser des données catégorielles ou nominales. Elle permet d'explorer et de visualiser la structure des relations entre différentes variables catégorielles dans un tableau de contingence [4].
Les données catégorielles sont des données qualitatives qui peuvent prendre des valeurs nominales (sans ordre) ou ordinales (avec un ordre). Le but de l'ACM est de transformer ces données en coordonnées numériques pour représenter graphiquement les relations entre les différentes catégories. Cela permet de visualiser les associations ou les regroupements entre les variables catégorielles et de mettre en évidence les structures sous-jacentes dans les données [5].
Le processus de l'ACM implique généralement les étapes suivantes :

1. Construction du tableau de contingence : Les données catégorielles sont organisées dans un tableau de contingence croisant les différentes variables.

2. Calcul des fréquences : Les fréquences des différentes catégories dans le tableau de contingence sont calculées.

3. Standardisation des données : Les données sont transformées pour supprimer l'effet de taille des variables et leur donner une distribution comparable.

4. Calcul des valeurs propres et des vecteurs propres : L'ACM utilise une décomposition en valeurs propres pour extraire les axes principaux (composantes principales) qui représentent la variance des données.

5. Projection des données sur les axes principaux : Les données sont projetées sur les axes principaux pour obtenir les coordonnées des observations et des catégories dans un nouvel espace.

6. Visualisation des résultats : Les résultats sont généralement représentés graphiquement, en utilisant des graphiques comme les biplans ou les cercles de corrélation pour mon.

2.1 Définition

On considère p variables qualitatives ($p > 2$) notées, $\{X_j; j = 1, ...p\}$ possédant respectivement c_j modalité avec

$$c = \sum_{j=1}^{p} c_j$$

On suppose que ces variables sont observées sur n individus, chacun affecté du poids $\frac{1}{n}$.

Soit $X = [X_1 | .... | X_p]$ le tableau disjonctif complet des observations $(X..est..n * c)$.

On appelle Analyse des Correspondances Multiples (ACM) des variables $(X_1, ..., X_p)$ relativement à l'échantillon considéré, l'Analyse Factorielle des Correspondances (AFC) réalisée soit sur la matrice X . On note $n_k^j (1 \le j \le p, 1 \le k \le c_j)$ l'effectif de k ième modalité

$D_j = \frac{1}{n} diag(n_{(1)}^j, .., n_{(c_j)}^j)$ et $\Delta = diag(D_1, ...D_P)$

On note : Δ est une matrice diagonale d'ordre c et D_j est la matrice diagonale d'ordre c_j $1 < j \le p$

2.2 Tableau disjonctif complet

Soit X une variable qualitative à c modalités. On appelle variable indicatrice de la k-ième modalité de $\boldsymbol{x}$ (k = 1, ..., c), la variable $X_{(k)}$ définie par

$$\mathbf{X_{(k)}(i)} = \begin{cases} \mathbf{1 \ si \ X(i) = x_k} \\ \mathbf{0 \ \ sinon} \end{cases}$$

où i est un individu, dans notre cas une vulnérabilité et $\boldsymbol{x_k}$ est la k–ième modalité de X.
On notera $\boldsymbol{n_k}$ l'effectif de $\boldsymbol{x_k}$. On appelle matrice des indicatrices des modalités de $\boldsymbol{X}$ et l'on notera X, la matrice n×c de terme général : $\boldsymbol{x_i^k = X_{(k)}(i)}$.
Considérons maintenant $\boldsymbol{p}$ variables qualitatives $\boldsymbol{X_1}, \ldots\ldots, \boldsymbol{X_p}$. On note $\boldsymbol{c_j}$ le nombre de modalité de $\boldsymbol{X_j}$, $\boldsymbol{c = \sum_{j=1}^{p} c_j}$ et $\boldsymbol{X_j}$ la matrice des indicatrices de $\boldsymbol{X^j}$ .On appelle alors tableau disjonctif complet la matrice X, n×c, obtenue par concaténation des matrices
X_j : $X = [X_1 | .... | X_p]$ [6].

3. Méthodes Elbow et Kmeans

3.1 Méthode ELBOW

3.1.1 Definition

Dans le clustering, la méthode Elbow ou méthode du coude est une heuristique utilisée pour déterminer le nombre optimal k de clusters dans un ensemble de données. La méthode consiste à tracer la variation expliquée en fonction du nombre de clusters, et à choisir le coude de la courbe comme le nombre de clusters à utiliser. Elle permet de déterminer cette valeur optimale de k

3.1.2 Principe

En clustering, l'inertie est la somme des carrées des distances entre chaque centroïde d'un cluster et les différentes observations inclues dans le même cluster. La méthode ELBOW cherche à trouver donc un nombre **k** de clusters de telle sorte que les clusters retenus minimisent l'inertie intra classe dans le même cluster. La variance des clusters se calcule comme suit:

$$\boldsymbol{wcss_k} = \sum_{j} \sum_{x_i \to c_j} \boldsymbol{d}(\boldsymbol{c_j^k x_i})^2$$

$\boldsymbol{c_j^k}$: Le centre du cluster **k** (le centroïd)

$\boldsymbol{x_i}$: La ième observation dans le cluster ayant pour centroïd $\boldsymbol{c_j^k}$

$\boldsymbol{d}(\boldsymbol{c_j^k}, \boldsymbol{x_i})^2$: La distance (euclidienne ou autre) entre le centre du cluster et le point $\boldsymbol{x_i}$.

3.2 Méthode KMEANS

3.2.1 Définition

K-means est une méthode de quantification vectorielle. C'est une méthode de minimisation alternée qui, étant donné un entier **k**, va chercher à séparer un ensemble **X** d'observations en **k** clusters.

3.2.2 Description

Étant donné un ensemble de $\boldsymbol{n}$observations, $(\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n})$, où chaque observation est un vecteur réel de dimension d, l'algorithme des k -means vise à partitionner les n observations en $\mathbf{k} (\leq \mathbf{n})$ ensembles $\boldsymbol{S_1}, \boldsymbol{S_2}, \ldots, \boldsymbol{S_k}$ de manière à minimiser la somme des carrés intra-cluster (WCSS). Formellement, l'objectif est de trouver :

$$\arg\min_{S} \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2 = \arg\min_{S} \sum_{i=1}^{k} |S_i| Var S_i$$

où $\boldsymbol{S_i} = \boldsymbol{S_1}, \boldsymbol{S_2}, \ldots, \boldsymbol{S_k}$

$\boldsymbol{\mu_i}$ est la moyenne des points de $\boldsymbol{S_i}$

4. Etat de l'art

4.1 Détection des vulnérabilités par les techniques d'apprentissage non supervisé.

Des chercheurs en sécurité informatique ont publié des travaux concernant cette technique.

K. Kumar et al [7] ont présenté une approche pour identifier les vulnérabilités stockées dans les weblogs. Ils présentent une nouvelle approche basée sur l'algorithme K-Means pour analyser les données en utilisant différents attributs comme le protocole, le numéro de port, etc. afin de détecter des vulnérabilités. Dans ce processus, ils ont utilisé les techniques du prétraitement pour éliminer les attributs indésirables des données des weblogs.

Gupta et al [8] proposent une approche de la détection des vulnérabilités. Basée sur la combinaison de l'algorithme K-Means et des règles d'association. Les expériences ont été effectués les données KDD CUP 99. Cette approche permet de déterminer un bon taux de détection uniquement dans le cas d'une attaque par déni de service (DOS), mais est limitée dans le cas des autres types de vulnérabilités.

Zhengjie et al [9] proposent une méthode basée sur la combinaison de l'algorithme K-means et l'algorithme d'optimisation par essaims de particules (Kmeans-OEP). Les expériences ont été effectués sur la base de données KDD CUP 99. Elles ont montré l'efficacité de la méthode proposée et montrent également que la méthode a un taux de détection plus élevé et un taux d'erreur détection plus faible.

G. Schaffrah et al [10], ont effectué des recherches dans le domaine de la détection de vulnérabilité basés sur le flux. Ce travail fournit une classification des techniques d'attaque et de défense et montre comment les techniques basées sur les flux peuvent être utilisées pour détecter les scans, les vers, les botnets et les attaques de déni de services (DoS).

4.2 Limites des travaux existants

Les études antérieures pour l'identification des vulnérabilités ont utilisé la base de données KDD CUP 99. Cependant, cette base de données ne contient que des données quantitatives, ce qui limite notre capacité à évaluer les niveaux de vulnérabilité pour des données contenant des descripteurs qualitatifs.

Par la suite, nous avons remarqué que les travaux de Gupta et al. ne permettent d'identifier que les attaques par déni de service. Pour surmonter ces limitations, nous nous sommes tournés vers l'utilisation d'une base de données provenant de cevdetail.com. Cette base de données décrit les vulnérabilités en se basant principalement sur des variables qualitatives.

Une autre particularité de notre approche réside dans le fait que peu de recherches ont été réalisées sur cette base de données pour la découverte de nouvelles vulnérabilités. Nous évitons donc une approche de

modélisation par apprentissage artificiel qui combine l'Analyse des Correspondances Multiples, la méthode Kmeans et la méthode Elbow pour identifier les familles potentielles de vulnérabilités logicielles qui n'ont pas encore été découvertes.

5. Nouvelle Approche

5.1 Principe

Le jeu de données d'étude est la base de données recueillie sur le site web de recherche https://www.cvedetails.com/vulnerability-list/year-2021/vulnerabilities.html. Nous nous sommes appuyé sur des données de 2021 de CVEdetails.com.

Cette base de données ne contient que des variables qualitatives. Le modèle que nous avons élaboré va permettre d'identifier les vulnérabilités potentielles et les vulnérabilités inconnues dans les applications. Pour atteindre ces objectifs, nous avons écrit un algorithme dénommé **IdsoftVulNew** . Cet algorithme suit les étapes suivantes :

- Etape 1 : Transformer notre base de données qui contient uniquement des variables qualitatives par la technique de l'ACM ;
- Etape 2 : Appliquer l'Algorithme Kmeans sur la base de données transformée ;
- Etape 3 : Appliquer la méthode Elbow pour déterminer le nombre optimal de cluster ;
- Etape 4 : Sélection automatique du K optimal
- Etape 5 : Procéder à l'évaluation du modèle par le score de silhouette.

ALGORITHME : IdSoftVulNew
Entrée
D : Base de Données de vulnérabilités avec des variables qualitatives
$\boldsymbol{\Delta_{ACM}}$:Base de données ayant été transformée par une ACM
NCluster, **ScoreSilhouette**, **i**: Entier

DEBUT
// Importer les données qualitatives
Importer (**D**)
$\boldsymbol{\Delta_{ACM}} \leftarrow$ **ACM**(**D**)
Lire ($\boldsymbol{\Delta_{ACM}}$)
// Elaboration du modèle
NCluster ← **1**
Pour i ← **NCluster** à **11 faire**
$\boldsymbol{Vuln} \leftarrow$ **KMEANS**(**NCluster**[**i**])
$\boldsymbol{IdVulnew} \leftarrow \boldsymbol{Vuln.fit}(\boldsymbol{\Delta_{ACM}})$
Appliquer la Méthode Elbow sur l'inertie de $\boldsymbol{Vuln}$
Afficher le nombre de cluster par la méthode Elbow
Sélection automatique du K optimal
FinPour
// Evaluation du modèle par le Score de Silhouette
Pour i ← **1** à **11 faire**
$\boldsymbol{ScoreSilhouette} \leftarrow \boldsymbol{SILHOUETTE}(\boldsymbol{IdVul}, \boldsymbol{\Delta_{ACM}})$
Afficher ($\boldsymbol{ScoreSilhouette}$)
FinPour
FIN

6. Résultats et discussion

En utilisant notre Algorithme **IdsofVulNew**, nous avons réussi à transformer la base de données de vulnérabilités de 2021 provenant de cvedetails.com en la recodant en données numériques. Après cela, en utilisant la méthode ELBOW, la figue 1 démontre que le nombre de clusters k est de quatre (04), comme le montre l'illustration ci-dessous.

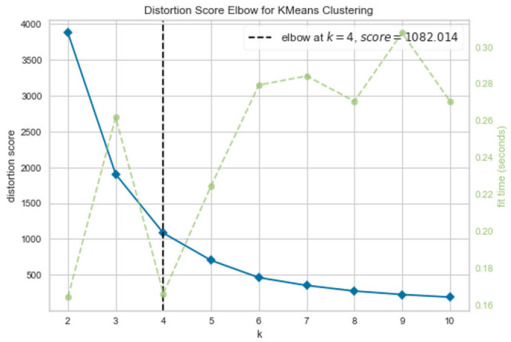

Figure 1 : Méthode Elbow

Enfin, **IdsofVulNew** nous permet d'identifier des familles de vulnérabilités logicielles inconnues pour lesquelles, les entreprises ou les organismes doivent faire attention.

Avec K=4, ce cluster représente quatre familles de vulnérabilités inconnues. Ils sont évalués par l'indice de silhouette avec un score moyen de silhouette de 0,58 représenté par le tracé vertical rouge. Comme le confer la figure 2.

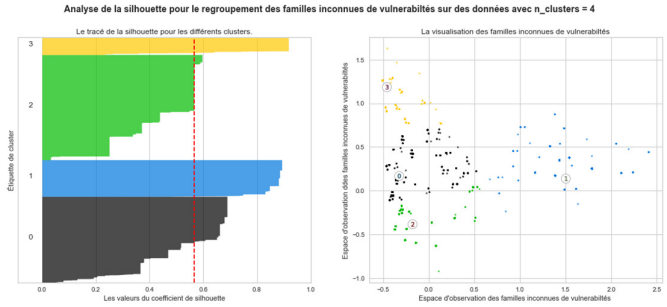

Figure 2 : Familles de vulnérabilités inconnues pour k=4

4. Conclusion

Le score moyen de la silhouette $\boldsymbol{\epsilon\ [0,51\ ,\ 1]}$. Selon l'Echelle de silhouette, cela signifie que l'identification des familles de vulnérabilités logicielles découvertes sont de bonnes qualités, comme illustré dans le tableau ci-dessous.

Nombre de cluster	Score moyen de silhouette	Nature de la structure
4	0.58	Forte

Tableau 1 : Performance de IdSoftVulNew

En somme, nous pouvons dire que l'algorithme **IdSoftVulNew** est un bon modèle d'identification de vulnérabilités logicielles.

Acknowledgements

Nos remerciements vont à l'endroit du LARIT et du LASTIC ainsi que tous les enseignants chercheurs de l'Ecole Supérieur Africaine de Technologie (ESATIC) ainsi que son Directeur général Prof Konaté Adama.

REFERENCES

[1] R. Akrout, « Analyse de vulnérabilités et évaluation de systèmes de détection d'intrusions pour les applications Web », p. 155.

[2] DIAKO Doffou Jérôme, ACHIEPO Odilon Yapo M., MENSAH Edoete Patrice, « Analysis_of_Software_Vulnerabilities_Using_Machine_Learning_Techniques ». » e-Infrastructure and e-Services for Developing Countries, pp.30-37, FEVRIER 2020.

[3] F. Rustam et al., « COVID-19 Future Forecasting Using Supervised Machine Learning Models », IEEE Access, vol. 8, p. 101489-101499, 2020, doi: 10.1109/ACCESS.2020.2997311.

[4] R. Rakotomalala, « Analyse des Correspondances Multiples – ACM », p. 67.

[5] L. Dhanabal et D. S. P. Shantharajah, « A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms », vol. 4, no 6, p. 7, 2015.

[6] Wikistat, « Analyse factorielle multiple des correspondances (AFCM) ». 2016. [En ligne]. Disponible sur: https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-explo-afcm.pdf

[7] U. Kumar, C. Joshi, et N. Gaud, « Information Security Assessment by Quantifying Risk Level of Network Vulnerabilities », Int. J. Comput. Appl., vol. 156, no 2, p. 37-44, déc. 2016, doi: 10.5120/ijca2016912375.

[8] C. Gupta, A. Sinhal, et R. Kamble, « Intrusion Detection based on K-Means Clustering and Ant Colony Optimization: A Survey », Int. J. Comput. Appl., vol. 79, no 6, p. 30-35, oct. 2013, doi: 10.5120/13747-1555.

[9] Zhengjie Li et al, « «Anomaly Intrusion Detection Method Based on K-means Clustering Algorithm with Particle Swarm Optimization,» », International Conference of Information Technology, Computer Engineering and Management Sciences, 2011.

[10] G.Schaffrath et al, « An Overview of IP Flow-Based Intrusion Detection Communications Surveys & Tutorials », IEEE, 2010.

