Decoding Deceit: A Multivariate Exploration of Web Page Phishing Detection

António Figueiras
99402, MECD
Instituto Superior Técnico
Lisboa, Portugal

Carolina Filipe
89723, MECD
Instituto Superior Técnico
Lisboa, Portugal

Clara Pereira
99405, MECD
Instituto Superior Técnico
Lisboa, Portugal

Tomás Juhos
99446, MMAC
Instituto Superior Técnico
Lisboa, Portugal

***Abstract*—**
The increasing popularity of the Internet led to a substantial growth of e-commerce. However, such activities have main security challenges primarily caused by cyberfraud and identity theft. Therefore, checking the legitimacy of visited web pages is a crucial task to secure costumers' identities and prevent phishing attacks.
In this study, we employed multivariate analysis techniques, such as factor analysis and clustering, to unveil patterns and relationships within the data. Additionally, we conducted a comparative analysis of supervised and unsupervised learning classification methods within this context.
Our results showed that, after dimensionality reduction of the data,not only is it possible to identify distinct clusters of websites with similar profiles (although not perfectly separating between legitimate and phishing websites), but also accurately predict if a website is legitimate or phishing, through supervised learning.

***Index Terms*—Multivariate Analysis, Principal Component Analysis, Clustering, Classification**

I. Introduction

In the landscape of cyber threats, phishing stands out as a prevalent strategy used by cybercriminals to deceive individuals and obtain sensitive personal and financial information. The increasing reliance on the internet for daily activities unintentionally creates an environment conducive to the execution of targeted phishing attacks. The contemporary phishing landscape exhibits a notable level of sophistication, making these deceptive campaigns progressively challenging to identify.
This fact underscores the need for a deeper understanding of phishing dynamics and the development of effective countermeasures to strengthen our defenses. Consequently, this study explores the intricacies of phishing attacks, aiming to shed light on the tactics used by cybercriminals.
Our methodology employs advanced multivariate analysis techniques, such as factor analysis and clustering, to unveil latent structures within the data. These methods provide a nuanced perspective on the complex interplay of variables associated with web page legitimacy.
Simultaneously, we use machine learning models, conducting a comparative analysis between supervised and unsupervised learning classification methods. This approach seeks to assess the effectiveness of predictive models in distinguishing legitimate and fraudulent web pages, as well as uncovering intrinsic data structures that traditional classification paradigms may overlook.
The subsequent sections will detail our methodology, outlining the steps taken to dissect the dataset, and present a comprehensive analysis of the findings.

II. Materials and Methods

A. Computational Environment

All the experiments were conducted in R.

B. Dataset

For this study, the "Web page Phishing Detection Dataset" was used. The data was collected in May 2020, and is publicly available on Kaggle and on Mendeley Data [1].
Before preprocessing, the dataset was comprised of 11430 phishing and legitimate URL's with 1 categorical response variable (takes values "legitimate" or "phishing") and 87 extracted features (co-variables), from three distinct classes: 56 derived from the structure and syntax of URLs, 24 from the content of their corresponding pages, and 7 obtained through external service queries. Ensuring balance, the dataset presented an equal distribution of 50% phishing and 50% legitimate URLs.

C. Data Preprocessing

The data was preprocessed to ensure that it was suitable for analysis.
This involved, firstly, eliminating data that did not convey any information, for instance co-variables with constant values for all observations, which were deleted.
The categorical variables were transformed in order to be treated as such (with command **as.factor** from R). This included the response variable "status", which was transformed to take values 0 and 1 instead of "legitimate" and "phishing".
Furthermore, we filtered any incoherent observations/variables. The variables "Domain Age" and "Domain Registration Length", which are non-negative integers, presented some negative observations. These were removed from the dataset, together with their correspondent URL.
Remarkably, after preprocessing, the dataset remained balanced, with 4720 examples of phishing, and 4866 of legitimate URLS.
Finally, after processing, the dataset comprised of 9586 URL's, and 81 variables - 1 response variable and 80 co-variables.

D. Preliminary Data Analysis

1) Summary of Co-variables: To further our knowledge of the dataset, each variable was analysed in detail: for continuous variables (all numeric variables were considered to be continuous, even discrete ones, given their wide range), several descriptive measures were computed, namely the range of variables, the mean, median and quartiles.
Normality was also tested by plotting histograms for each co-variable. Since none of the plots showed the silhouette of a normal distribution, there was no need for further normality tests (Shapiro-Wilk test, for instance - see [2]).

2) Association between Co-variables: Identifying correlations between co-variables is crucial in multivariate analysis' efficiency [3]. Given two co-variables that are highly correlated, specifying the contribution of each co-variable to the response variable is a hard task. Additionally, including both variables in the model might not add new information - which also justifies the next step in the data processing: dimensionality reduction.
Since our dataset includes mixed type data, it is necessary to use adequate measures for each type.
We now describe each of the measures:

- *Pearson's Coefficient*: For continuous variables, the correlation measure chosen was Pearson's Coefficient (ordinary correlation), a scalar between -1 and 1. The larger its absolute value, the stronger the linear relationship between the two variables is. Negative values indicate a negative association, whereas positive values indicate a positive one.
- *Phi Coefficient*: For categorical variables with 2 levels, the association was assessed through Phi Coefficient - introduced by B. W. Mathews in 1975 (see [4]). The coefficient takes values between -1 and 1. In this case, there is no linear relationship assumption (as it would not make sense for categorical variables).
- *Cramer's V*: A measure of association between two categorical variables with two or more levels. It takes values between 0 and 1 - the larger the value, the greater the association [5].
- *Point-biserial Coefficient* This measure is a variation of Pearson's coefficient, designed for assessing the association between a numeric and a categorical variable with two levels. Takes values between -1 and 1, once again representing the strength and direction of the association between the variables [6].

3) Dimensionality Reduction: Before delving into classification, since some input variables might be irrelevant to the classification problem, dimensionality reduction techniques were employed. Besides that, dimensionality reduction is also important to decrease the computational cost of both supervised and unsupervised learning algorithms. The methods discussed in class were the classical PCA (Principal Component Analysis) and ROBPCA (Robust Principal Component Analysis). However, these methods are designed for continuous variables, and our dataset is formed of mixed data, containing both continuous and categorical variables. Due to the aforementioned characteristics of the data, we ended up choosing FAMD (Factor Analysis of Mixed Data). We can approximately say that FAMD works as PCA, for quantitative variables, and as MCA (Multiple Correspondence Analysis) for categorical ones. The method transforms the data in a way that both variables are on equal foot to determine the dimensions of the variability, where categorical variables are scaled using the specific scaling of MCA and continuous variables are scaled to unit variance. More details about FAMD in [7].
Note that continuous co-variables were standardized be-

fore applying FAMD to the dataset.

E. Classification

Following data preprocessing, supervised classification methods were applied to categorize each link as either phishing or legitimate.

1) Supervised Learning: After the dimensionalisty reduction step, the data was split in train (80%) and test (20%). The training set was further partitioned into training and validation subsets for a 5-fold cross-validation (CV) process aimed at tuning the hyperparameters of each classifier.
This technique involves randomly splitting the data into K mutually exclusive folds. The model is subsequently trained on $K-1$ folds and tested on the one that was left out. This process is iterated K times to ensure comprehensive validation and robust evaluation of the model's performance (Figure 1).

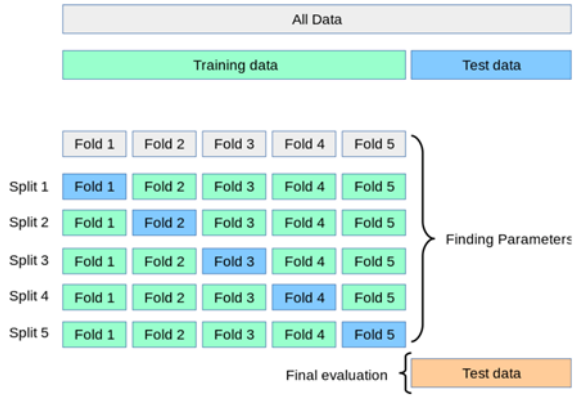

Fig. 1. 5-fold Cross-validation.

Then, four classifiers were trained: Support Vector Machines, Naive-Bayes, Random Forests and k-Nearest Neighbors. The SVM excels in identifying optimal hyperplanes for class separation. Naive Bayes operates on Bayesian principles, assuming co-variable independence, and demonstrates efficiency, particularly in high-dimensional datasets. Random Forests, as an ensemble method, combine multiple decision trees to improve accuracy and address overfitting concerns. k-Nearest Neighbors (k-NN), a non-parametric method, classifies data points based on the majority class among their k nearest neighbors, adapting to local patterns [8].

2) Unsupervised Learning: Ignoring the class variable, and also after the dimensionality reduction step, unsupervised methods were applied to our data, more specifically K-medoids. Agglomerative clustering could obtain satisfactory results, however this algorithm has a complexity of $\mathcal{O}(n^3)$, which makes the search for the appropriate dissimilarity a much harder task for datasets with a considerate number of observations, as it is in our case. K-medoids however has a complexity of $\mathcal{O}(n^2)$, allowing the testing of a more extensive number of distinct dissimilarity metrics. K-medoids is a partition method: it breaks the dataset into groups, in such a way that minimizes the distance between points belonging to a cluster, and the centre of that cluster. For K-medoids the centre of the cluster is its medoid, the point whose average dissimilarity to all points in the cluster is minimal.
The dissimilarity measures tested with K-Medoids were the following: Euclidean, Mahalanobis, Canberra and Gower.
Each of the chosen metrics had at least some advantage that we thought could improve results: Mahalanobis distance could ease the distance distortion caused by the linear combination of attributes done by the dimensionality reduction technique (FAMD); Canberra could result in accurate outcomes for medium-high dimensional datasets, specifically using K-medoids; Gower's distance [9] was thought to be a good approach since it can model different types of variables, using range-normalized Manhattan distance for quantitative variables, and using the Dice coefficient for categorical variables. Because of this ability to treat mixed type data, we not only applied K-medoids to the dataset with reduced dimensionality but also to our original dataset, that had both continuous and categorical variables.
To find the most appropriate number of clusters we used the Calinski–Harabasz index [10] and the Dunn index [11]. It is important to notice that to find the best number of clusters, it was decided to subsample our initial dataset and work with a random sample of 1000 observations. This choice was made since it was very costly to run K-medoids for a number of clusters that, based on our preliminary analysis, we thought to be large. Therefore, to calculate the CH and Dunn Indexes for all possible number of clusters and for all distinct dissimilarities we used a random subsample and later validated that the structure of the clustering could be generalized to the whole data-set, without loss of its properties.
To validate the choice of the ideal number of clusters, the dataset was partitioned into 10 equal random disjoint subsets, and optimal clustering numbers were investigated. If the indexes were indeed maximized with approximately the same number of clusters and had similar values in all the subsamples considered, we

would be able to validate the clustering chosen with the previous method.

3) Combining Supervised and Unsupervised Learning: Lastly, the classification process was repeated using supervised learning methods, this time using as classes the partition clusters obtained in the *Unsupervised Learning* section: instead of attempting to predict if a given link corresponds to a legitimate or phishing website, the classifier attempts to predict the cluster it belongs to. The link is then assigned to class 0 - "legitimate" or 1 - "phishing", according to their cluster's most frequent class. The supervised learning models used are the same as described in *Supervised Learning*'s section. However, the high number of clusters implied that some were very small sized, which revealed to be a problem in the Naive Bayes Classifier. For that reason, only the other 3 models were used - SVM, Random Forest, and K Nearest Neighbors.

4) Metrics: In order to evaluate and compare the performance of the different models, four metrics were used: Accuracy, Precision, Recall and F1-Score, defined as:

$$Accuracy = 1 - PMC = \frac{\hat{n}_{00} + \hat{n}_{11}}{N} \quad (1)$$

$$Precision(0) = P(Y = 0|\hat{Y} = 0) = \frac{\hat{n}_{00}}{\hat{n}_0} \quad (2)$$

$$Recall(0) = P(\hat{Y} = 0|Y = 0) = \frac{\hat{n}_{00}}{n_0} \quad (3)$$

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Where PMC refers to Probability of Misclassification, N refers to the total number of links, $\hat{n}_{00}$ refers to the number of correctly predicted legitimate links, $\hat{n}_{11}$ refers to the number of correctly predicted phishing links, $\hat{n}_0$ refers to the number of links assigned to class 0 (correctly and incorrectly) and n_0 refers to the number of links from class 0.

III. Results and Discussion

In this chapter, we present the results of the previously discussed methods, as well as discuss the implications of our findings in the real-life context of the problem.

A. Preliminar Analysis

Let's start by exploring association between continuous variables, using Pearson's Coefficient. In order to visualize results, a heatmap was plotted.

Note that the plot is a matrix, in which each entry i,j represents the correlation between variables X_i and X_j through a color scale: a brighter color implies a stronger correlation, red for positive and blue for negative, as specified in Figure 2. The map is symmetrical, since $cor(X_i, X_j) = cor(X_j, X_i)$, and the values in the main diagonal are always equal to 1, since $cor(X_i, X_i) = 1$.

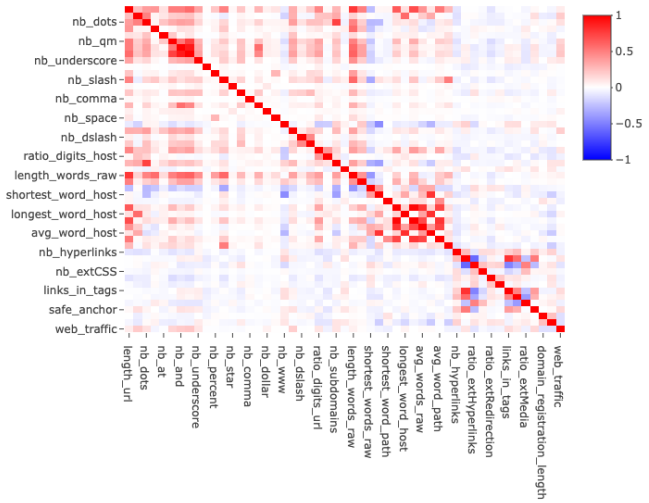

Fig. 2. Heatmap of correlation between numeric variables

As seen on Fig. 2, there are numeric variables showing a significant correlation coefficient. In terms of negative correlation, **Ratio of External Links** and **Ratio of Internal Links** show a significant value, having a stronger blue on the respective entry of the plot. This implies that a website with a higher ratio of internal links will probably have a lower ratio of external links, and vice-versa. Some co-variables show a high positive correlation, namely **Nb_and** and **Nb_eq**, **Longest_words_raw** and **Longest_word_path**, **Shortest_word_host**, which in the context of our problem indicates that the information of one of the members of each pair may suffice to express both their contributions to the response variable - i.e. it is not necessary to include both members of the pair in our analysis. Consequently, we expect that these variables will probably be collapsed into one or more in the process of dimensionality analysis.

Next, we searched for association between categorical variables with 2 levels, using Phi Coefficient.

Again, in order to visualize results, a heatmap was

plotted. Since both Pearson and Phi coefficient take values in the same range, we can analyse the plot in the same way.

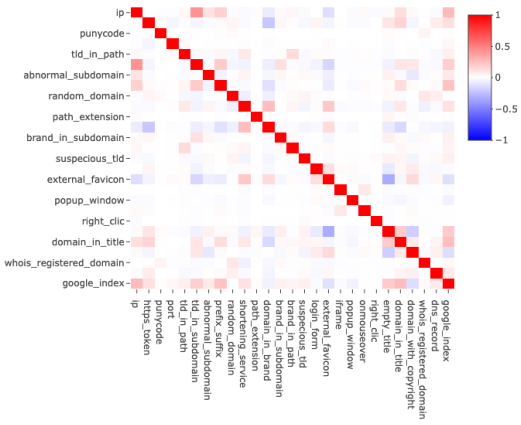

Fig. 3. Heatmap of correlation between categorical variables with two levels

Fig 3 shows low association between variables, as no entry portrays a bright color, except for the main diagonal, where the association is 1, representing the association between a variable and itself.
The remaining associations (categorical vs categorical with more than two levels, numeric vs categorical) were also computed, using the aforementioned coefficients. However, there were no results worth mentioning, so it was decided to leave them out of the report.
After association analysis of the data, we performed FAMD on our standardized dataset, obtaining the following scree plot:

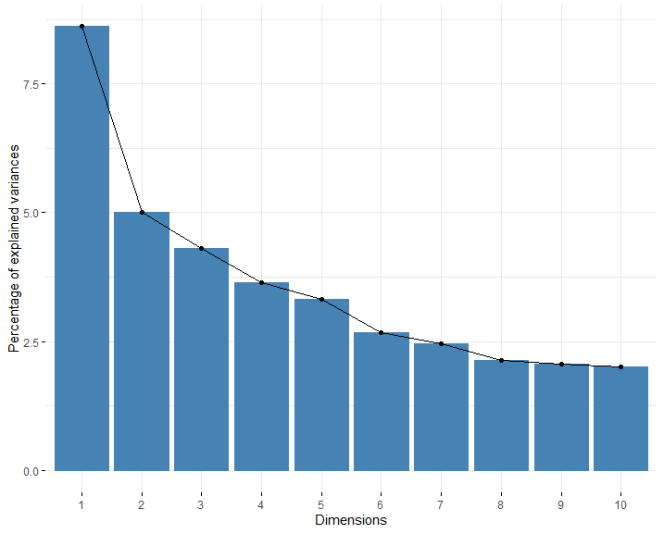

Fig. 4. Scree plot of the explained variance of each dimension

In Figure 4 we can see that our first dimension explains 8.61% of the population total variance, and the second dimension explains 5.00%, meaning that projecting the dataset onto the 2 most important dimensions would only preserve 13.61% of the population's total variance. We can see that for higher dimensions we also have a slow decrease on the percentages of explained variance. Also, to complement the information given by scree plot 4, we also created the following plot of the cumulative explained variance:

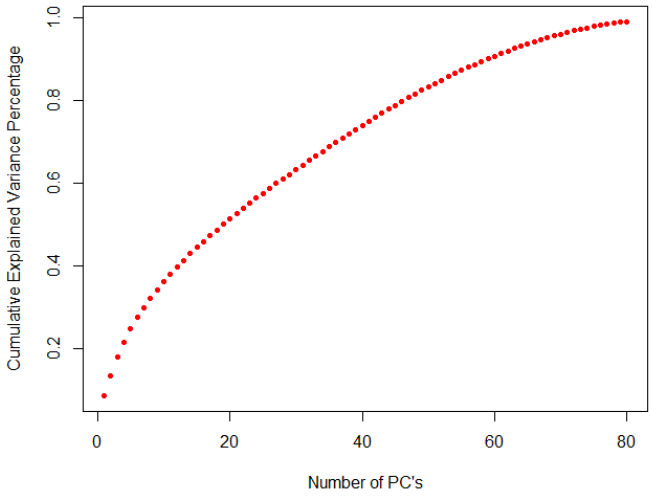

Fig. 5. Cumulative explained variance by dimension kept

By observing Figure 5, the obtained eigenvalues and the respective explained variance of each component, we concluded that in order to reach 50% of cumulative explained variance we would need to keep 19 PC's, to reach 75% we would need 41 PC's, and to reach 90% of cumulative explained variance we would need a total of 59 PC's. Also, after the 42th PC, the variance percent explained by each of the following PC's is less than 1%. We chose to keep a total of 45 PC's explaining a total of 78.89% of the total variance, a few units after the threshold of 75% was reached. We did not chose a higher threshold since as said before the increase of the cumulative variance explained starts to get really slow, needing many more principal components to observe a significant growth on total explained variance, highly increasing computational cost of the unsupervised and supervised learning algorithms.

B. Supervised Classification

The results of the supervised classification experiments are presented in Table I, where the classification models are evaluated based on performance metrics,

including accuracy (Acc), precision (P), Recall (R), and F1-score (F1). Additionally, the hyperparameters obtained through cross-validation tuning are discussed below for each specific model.

Hyperparameter tuning plays a pivotal role in optimizing the models' performance. For the SVM model, the cross-validation process revealed that maintaining the tuning parameter "C" at a constant value of 1 produced the most effective results. This choice indicated a balance between the desire for a smooth decision boundary and accurate classification of training points. Similarly, the Naive-Bayes model underwent a meticulous tuning process, where laplace smoothing, kernel density estimation usage, and probability adjustment were tunned. The final hyperparameter configuration included laplace = 0, usekernel = TRUE, and adjust = 1, underscoring the model's ability to leverage kernel density estimation and fine-tune probability estimates.

The Random Forest model's hyperparameter tuning process focused "mtry", the number of features considered for splitting at each node. Through cross-validation, it was determined that setting "mtry" to 2 yielded the optimal performance, emphasizing the importance of feature randomness in enhancing the model's robustness. Lastly, the k-NN model benefited from a careful exploration of the 'k' parameter during cross-validation, revealing that considering the five nearest neighbors offered the best compromise between accuracy and computational efficiency.

TABLE I: The comparison of different classification models

	Acc	P	R	F1
Support Vector Machines	0.94	0.94	0.95	0.94
Naive-Bayes	0.89	0.87	0.91	0.89
Random Forest	**0.96**	**0.95**	**0.96**	**0.96**
k-Nearest Neighbors	0.95	0.93	0.96	0.95

Random Forest outperformed the other models with the highest accuracy (0.96), precision (0.95), recall (0.96), and F1-score (0.96). Support Vector Machines also exhibited strong performance across all metrics, with accuracy, precision, sensitivity, and F1-score all exceeding 0.94.

K-Nearest Neighbors demonstrated competitive performance, particularly in accuracy (0.95) and F1-score (0.95). This highlights the effectiveness of the k-NN approach in identifying similar clusters of websites based on their features.

Naive-Bayes achieved a respectable overall performance, although it showed slightly lower metrics compared to Random Forest and k-Nearest Neighbors.

These results highlight the effectiveness of Random Forest for the specific classification task considered in this study.

Subsequently, the computation of feature contributions involved multiplying the FAMD feature loadings by the importance scores derived from the Random Forest model, identified as the top-performing model, followed by normalization for interpretability (see [12]).

In Figure 6, we explore the normalized contributions of individual features (co-variables) obtained from the Random Forest model for Web Page Phishing Detection. The top 20 features with the highest normalized contributions are emphasized, shedding light on the crucial factors influencing the effectiveness of the phishing detection model.

Each bar in the plot corresponds to a specific feature, and the height of the bar represents the normalized contribution of that feature to the overall importance in detecting phishing links. By concentrating on the top contributors, we gain valuable insights into the key elements significantly impacting the Random Forest model's ability to distinguish between legitimate and phishing web pages.

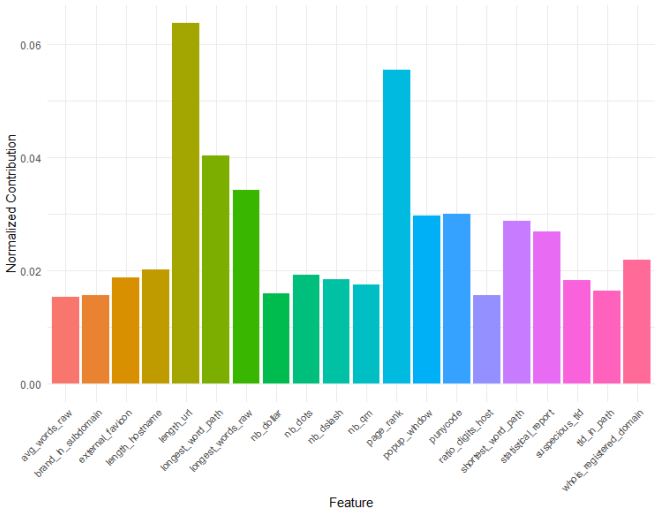

Fig. 6. Top 20 features with their normalized contributions in the Random Forest model for Web Page Phishing Detection.

Notably, Length of URL emerges as the most impactful feature, indicative of its significant role in the model's decision-making process. Page Rank's contribution is also noticeable. Other URL-related features, such as length and composition, play a substantial role, suggest-

ing that characteristics inherent in the structure of web addresses are pivotal in identifying potential phishing threats. Additionally, linguistic aspects, including the length of words in the URL path and content, contribute significantly to the model's predictive accuracy. These findings provide a nuanced understanding of the discriminative features within our Random Forest model, paving the way for enhanced accuracy and robustness in Web Page Phishing Detection.

C. Unsupervised Learning

Next we applied unsupervised methods, by ignoring our class variable. As stated before, the initial computation was executed over a random sample of 1000 observations and only later it was validated that the structure of the clustering stays the same, no matter which sample is chosen. After applying K-medoids the metric that led to the most satisfactory results in terms of the CH and Dunn Index was the Mahalanobis distance. The following plot contains the CH scores for the Mahalanobis distance for a range of clusters between 2 and 90:

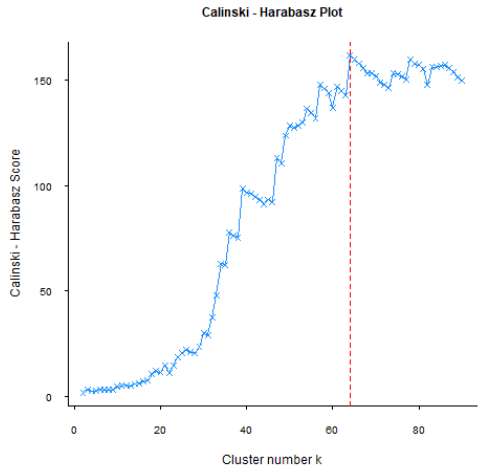

Fig. 7. CH: Clustering with K-medoids and Mahalanobis distance

We were able to achieve a maximum CH-score at 64 clusters with a value of 161.83. From the plot we can also see that the CH-score continued to increase until it reached it's maximum, and after that it stopped, being a good signal that choosing more clusters wouldn't add much information and wouldn't separate our data in a better way. To compare with Figure 7, the scores obtained by the euclidean distance were the following:

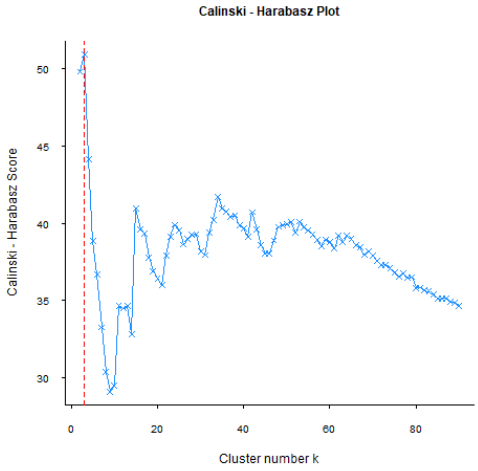

Fig. 8. CH: Clustering with K-medoids and Euclidean distance

As we can see in Figure 8, the maximum was reached at 3 clusters, with a value of 50.94, getting a very low CH index value when compared to the ones obtained using Mahalanobis distance. Besides having overall lower index values we also could conclude that euclidean distance would be a bad choice for the dissimilarity since the best values are obtained from a low amount of clusters, which is contradictory to the values obtained from other dissimilarities. Regarding Dunn indexes, the highest score was actually obtained using euclidean distance, reaching a maximum of 0.1127 at 73 clusters, contradicting the previously seen results from the euclidean CH indexes. The second best values were obtained using Mahalanobis distance, having a very similar behavior to the one seen with CH Indexes:

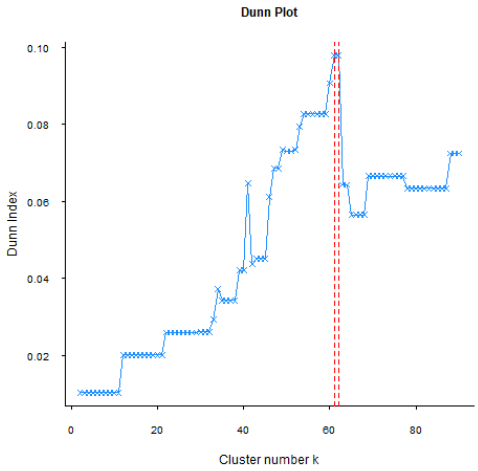

Fig. 9. Dunn: Clustering with K-medoids and Mahalanobis distance

A maximum of 0.0978 was reached at a number of clusters equal to 61 and 62, and therefore, jointly with the results from CH indexes, we concluded that for this chosen sample the number of clusters to choose should be located in the interval between 60 and 64 clusters.
Unfortunately, despite Gower's dissimilarity being adequate to datasets of mixed data, when applied to both the original dataset and the FAMD results, the CH and Dunn indexes were not satisfactory. When applied to the original dataset, although the maximum value of CH index reached a value of 171.9, it was only using a number of clusters equal to 2, decreasing rapidly after that value, having values overall smaller then the ones obtained by the Mahalanobis distance. Regarding Dunn indexes, they were also worse than the ones obtained in the range from 50 to 62 clusters, using Mahalanobis distances. We also discarded the Canberra distance since this metric had the worst scores of CH and Dunn Index, when compared to other tested dissimilarities.
Now with the most satisfactory distance metric picked, the Mahalanobis distance, we proceeded to attempt to validate that the best number of clusters for this dataset was indeed around 60 like the previous results suggested. To do this, just as mentioned in the methodology, the CH and Dunn indexes were computed for same size disjoint random samples of the dataset. The results in terms of the Calinski-Harabasz index were the following:

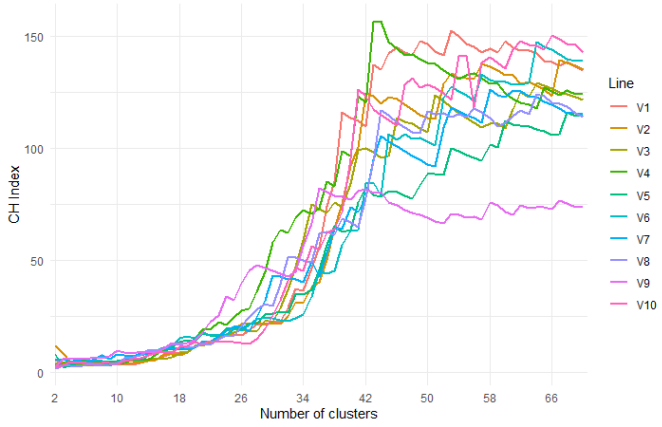

Fig. 10. CH index of the 10 partitions per number of clusters.

In Figure 10 we can clearly see a pattern emerge. The index sharply rises at around 30 to 40 clusters and stabilizes after that. This is in line with our previous result since 60 clusters is clearly above the threshold where the index becomes more stable. We could argue that possibly choosing around 43 to 47 clusters could yield similar or better results with a lower number of clusters, resulting in a better grouping of the data. To confirm that our initial hypothesis that approximately 60 clusters would yield better results, we proceeded to compute the CH index after clustering the whole dataset according to the number ranges that we wanted to compare (43 to 47 and 60 to 64 clusters). We verified that around 60 clusters the value of the index was much higher ($600 \sim 620$) than around 40 clusters ($340 \sim 370$), a satisfactory result since this is in line with our first hypothesis, the best number of clusters to pick being around 60.
Now for the Dunn index results in the partition we have the following plot:

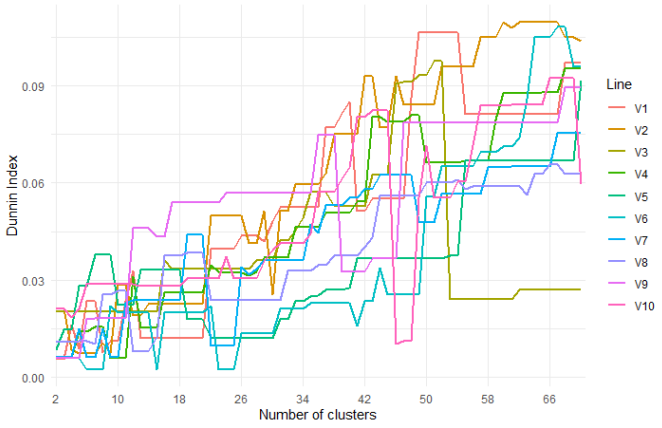

Fig. 11. Dunn index of the 10 partitions per number of clusters.

Despite the fact that we can still see a general uptrend in the Dunnin index along the number of clusters selected, the results are not as clear as in the previously analysed index. Furthermore, we note once again that index values stabilize around 40 clusters for most of the samples tested. Because of this we checked the same cluster number ranges as before for the whole dataset, arriving to the result that once again the values of the Dunn index are higher in the 60's range ($0.0025 \sim 0.0040$) than in the 40's range ($0.0020 \sim 0.0035$). Despite not being as strong evidence as the results with the CH index, this still supports the claim that the ideal number of clusters to pick for this dataset is somewhere slightly above 60.
After careful analysis the number of clusters selected was 62 as shown in Fig. 12, which represents the distribution of observations over the clusters, specifying their status - red for phishing and blue for legitimate. Directing our attention just to the distribution and ignoring class labels (status), we conclude that there are some small sized clusters. This could suggest that this number of

clusters could be just a bit too high for the dataset in question, leading to these small clusters and "overfitting" in the sense of unsupervised methods. We believe that the structure and nature of the dataset itself has a high impact in the number of clusters: phishing links do take extreme/unusual values, however the abnormalities are diverse and do not show consistently in a handful of variables - one phishing link may have an abnormal length, another an abnormal count of the character "=", etc. Keeping that in mind, we could hypothesize that many of the clusters try to capture one of these abnormalities, and since they are so diverse and unrelated, more clusters are needed to capture them.

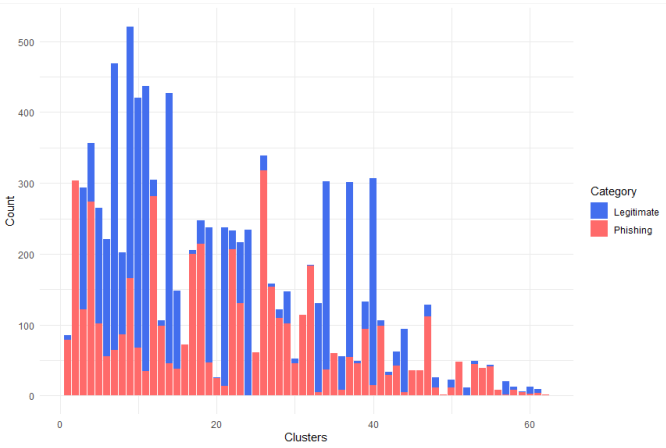

Fig. 12. Cluster distribution.

Now looking at the class labels in 12 we get satisfactory results. Despite there being very few "pure" clusters, most clusters are skewed towards one class or the other. Because the clusters are not "pure" in most cases, we cannot draw sure conclusions about the class of an observation just by the cluster it belongs to, but the imbalance in each of the individual clusters allows us to have a strong sense of which of the classes the observations in that cluster should belong too. Even though the results are not ideal, there is still room for improvement, however that kind of analysis goes beyond the scope of this study. Possible techniques/improvements to clustering methods will be later discussed in the final section.

D. Combining Supervised and Unsupervised Learning

After we were done with both supervised learning, using the true labels, and unsupervised clustering, using K-Medoids, we combined both methods using the cluster of each observation as its class and then, choosing the most probable class of the cluster as the method's final output. The scores tested for each method were the following:

TABLE II: The comparison of different classification models using most probable class from the predicted cluster

	Acc	P	R	F1
Support Vector Machines	0.84	0.80	0.90	0.85
Random Forest	0.83	0.79	0.91	0.85
k-Nearest Neighbors	**0.85**	**0.82**	**0.91**	**0.86**

As we can see, k-NN had the best overall score, outperforming Random Forest, which was the algorithm that obtained the best scores on the first classification task, using the observation's true labels.
Additionally, SVM achieved a very similar score to k-NN, which also happened on the first classification task, being lower by 0.01% on Accuracy, Recall and F1-Score, and by 0.02% on Precision.
When comparing with the previous classification task, the scores were overall lower than the ones obtained before, showing that using classification with the true labels is a better option than to use clustering and apply classification after.

IV. Conclusion

This project's goal was to explore supervised and unsupervised classification methods using the chosen dataset, about web page phishing detection. From the results obtained we can conclude that, in our case, since the supervised methods yielded exceptional results, they are far more suitable for this problem. As mentioned in the discussion of the unsupervised methods' results, we ended up with a very large number of clusters and most of them were not "pure". Both these factors bring down the quality of the predictions of the unsupervised methods, which further supports the choice of supervised methods to analyse the dataset. As for the combination of both techniques, we were not able to reach the same quality of predictions as the supervised methods by themselves. This is as expected due to the fact that supervised methods had exceptional performance and the mediocre performance of the unsupervised methods just brought the performance of the combination of both methods down. In conclusion, supervised methods are by far the best way to tackle this problem, out of the ones explored.
Despite the many challenges faced, we are still able to draw some satisfactory conclusions from the work.

Looking at the supervised methods' results, we can see that we can label a website as legitimate or phishing with a lot of certainty. This proves that this dataset combined with the previously referred classification methods could play a huge role in developing a tool to identify phishing websites that would protect its users and combat cybercrime.

As for the difficulties faced, starting from the data treatment there were problems. The dimensionality reduction did not simplify the problem that substantially, leading to additional difficulties and complexity in the following tasks. We could have compromised on the amount of explained variance, settling for a lower value and simplifying the problem a lot instead of slightly, but we chose to maintain around 80%. We also did not touch any type of outlier analysis which could be interesting future work as we can hypothesize that most outliers should be phishing websites. As mentioned in the unsupervised methods section of the results, we could have tried to cluster the dataset in a lower number of groups to try to eliminate the small classes, despite the analysis conducted not pointing to that as strongly. Interesting future work could be to explore even more unsupervised techniques, since this was the part that fell short in comparison to the others as far as performance is concerned, with the goal of improving the already outstanding results of the supervised methods by combining them with better unsupervised methods.

V. References

[1] A. Hannousse, "Web page phishing detection," Jun 2021. [Online]. Available: https://data.mendeley.com/datasets/c2gw7fy2j4/3

[2] S. S. SHAPIRO and M. B. WILK, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3-4, pp. 591–611, dec 1965. [Online]. Available: https://doi.org/10.1093/biomet/52.3-4.591

[3] T. Kyriazos and M. Poga, "Dealing with multicollinearity in factor analysis: The problem, detections, and solutions," *Open Journal of Statistics*, vol. 13, 06 2023. [Online]. Available: https://www.scirp.org/journal/paperinformation?paperid=125846

[4] B. Matthews, "Comparison of the predicted and observed secondary structure of t4 phage lysozyme," *Biochimica et Biophysica Acta (BBA) - Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0005279575901099

[5] H. CRAMÉR, *Mathematical Methods of Statistics (PMS-9)*. Princeton University Press, 1999. [Online]. Available: http://www.jstor.org/stable/j.ctt1bpm9r4

[6] S. Gupta, "Point biserial correlation coefficient and its generalization," *Psychometrika*, vol. 25, no. 4, pp. 393–408, December 1960. [Online]. Available: https://ideas.repec.org/a/spr/psycho/v25y1960i4p393-408.html

[7] J. Pagès, *Multiple Factor Analysis by Example Using R*, ser. Chapman & Hall/CRC The R Series. CRC Press, 2014. [Online]. Available: https://books.google.pt/books?id=BR5jDAAAQBAJ

[8] A. C. Müller and S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media, 2016.

[9] J. C. Gower, "A general coefficient of similarity and some of its properties," *Biometrics*, vol. 27, no. 4, p. 857, 1971.

[10] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics*, vol. 3, no. 1, pp. 1–27, 1974. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/03610927408827101

[11] J. C. Dunn, "A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973. [Online]. Available: https://doi.org/10.1080/01969727308546046

[12] A. Palczewska, J. Palczewski, R. M. Robinson, and D. Neagu, "Interpreting random forest models using a feature contribution method," in *2013 IEEE 14th International Conference on Information Reuse Integration (IRI)*, 2013, pp. 112–119. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6642461