DATA MINING PROJECT
Report

Summary

For the implementation of the knowledge extraction model, the result/prediction of each vaccine's company was used. As for the algorithms, two specific classification algorithms, three clustering algorithms, and one association rule extraction algorithm were utilized. Specifically, the classification algorithms used were Random Forest Classifier and K-Nearest Neighbors Classifier, while the clustering algorithms used were K-Means, DBSCAN, and Agglomerative Clustering. Lastly, the Apriori algorithm was used to extract association rules. Data preprocessing was necessary, and a wide range of preprocessing techniques were employed. Steps were taken to avoid overfitting, and the results were evaluated.

Introduction

The categorization was done with the aim of finding the accuracy with which we can predict from which company each age group received the vaccine. Furthermore, clustering was performed to group the characteristic data of the set-in relation to the vaccine received by the given age groups. As for the association rules, association rules were derived between age groups and vaccines.

Data Preprocessing

In the data preprocessing stage using Python libraries, we initially removed the columns FirstDoseRefused, ReportingCountry, UnknownDose, and YearWeekISO with the goal of eliminating data that does not assist us in knowledge extraction or removing columns that do not contribute to the accuracy or quality of our model. In the next step, the data in the Vaccine column was assigned to five basic numerical categories. Furthermore, any feature that contained categorical content was converted into numerical content using methods such as get_dummies and map. Next, the dataset was expanded using the aforementioned methods, which increased its dimensions. To address this, dimensionality reduction techniques known as Principal Component Analysis (PCA) were employed, which helped us extract and create nine principal components that characterize our dataset. Additionally, normalization was applied to simplify the values of our dataset's features. Finally, the train_test_split technique was utilized to randomly split our dataset into two sets: the train set, which serves as the training data for the algorithms, and the test set, which serves as the evaluation data for our final model.

Algorithms

Random Forest Classifier: Since the classification we are going to perform refers to a dataset that has and we want to have multiple and different values as results/predictions, this classifier will be quite accurate compared to other classifiers such as logistic regression, support vector machines, etc.

K-nearest-neighbors: Like the random forest classifier, this classifier defines the outcome by creating regions with characteristics that lead to the same predicted value, ranging from 1 to 5. Additionally, due to the extensive and multidimensional nature of the model, the classifier faces classification time issues, but it produces satisfactory accuracy.

K-Means : This algorithm allows us to create a number of clusters that we desire based on the desired potential outcomes. In other words, it allows us to create five different clusters corresponding to five different types of vaccines, which seems to perfectly match the manner and philosophy of the preprocessing that was conducted. This is achieved by calculating the Euclidean distance from each initially created center.

DBSCAN: This algorithm calculates the points at which there is a difference in data density and creates clusters based on the maximum density diameter that we have specified, that is, the epochs.

Agglomerative Clustering: This algorithm creates clusters based on the methodology of calculating the distance between the input points. In our case, the Euclidean distance was used, along with the method of selecting the maximum distance. The number of clusters created is five, as we want to group our set into five different vaccine groups, as previously mentioned.

Methodology

The Scikit-Learn library was used, specifically the train_test_split method, to create random training and testing sets. In other words, we created two datasets as mentioned in the preprocessing of our dataset. In general terms, the model created is a unified model that includes all the requirements of the laboratory exercise and is implemented in 83 seconds, which is less than two minutes. Given the volume of information and the use of multiple algorithms, we believe that this is a satisfactory implementation time.

More specifically, the classifiers were implemented based on the x_train and y_train data, and their effectiveness was calculated based on the results they produced and the y_test set.

For the implementation of the Random Forest classifier, the corresponding library was used, and it was simply assigned to a variable to perform the fit_transform operation on the information. However, for the implementation of the K-Nearest Neighbors classifier, a loop structure was used, which allowed us to select the best number of neighbors based on the maximum accuracy we could achieve with the input dataset. Then, the algorithm was implemented with the most efficient number of neighbors.

Regarding the implementation of the clustering algorithms, the K-Means and Agglomerative Clustering algorithms only required the declaration of the desired number of clusters to be created. However, the DBSCAN algorithm requires the declaration of the eps parameter, which was determined by creating a graph where the maximum curvature indicates the value that the eps parameter should take.

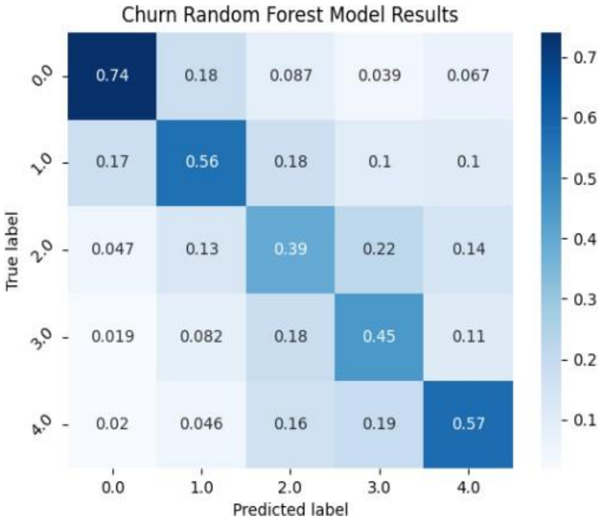
The last algorithm used is the Apriori algorithm, for which separate data preprocessing was performed. This preprocessing includes selecting data where the value of FirstDose is different from 0 and assigning all these data to the df3 dataset. Then, the TargetGroup and Vaccine columns are imported into a variable z, and they are combined into a single column named Total. Finally, the corresponding words are separated by commas using the split method.

Ultimately, the apriori algorithm is used to find frequent itemsets and itemset relationships between vaccines and age groups with min_support = 0.02. Then, rules are extracted with min_threshold = 0.02 and confidence as the metric.

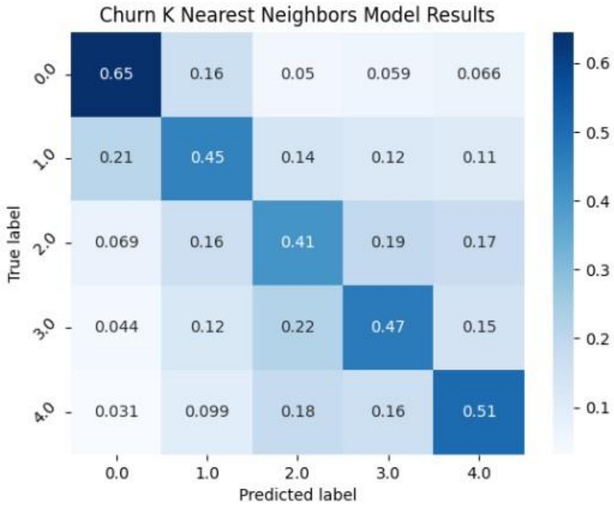
EXPERIMENTAL EVALUATION

CLASSIFIERS:

Random Forest Classifier:

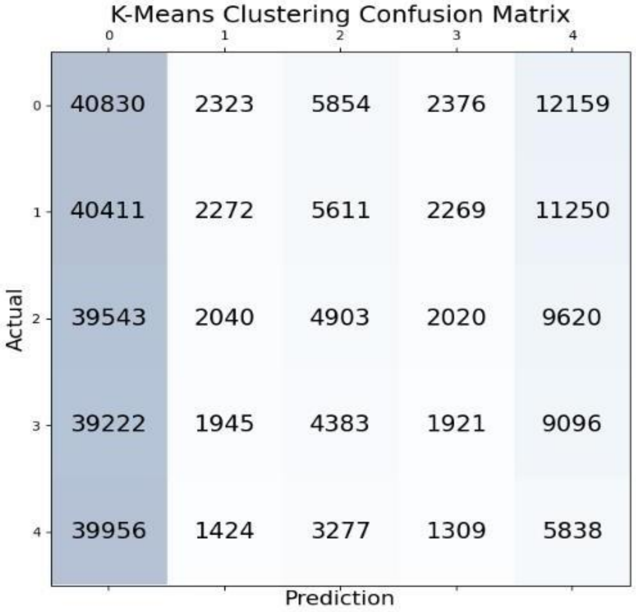


K-Nearest Neighbors:

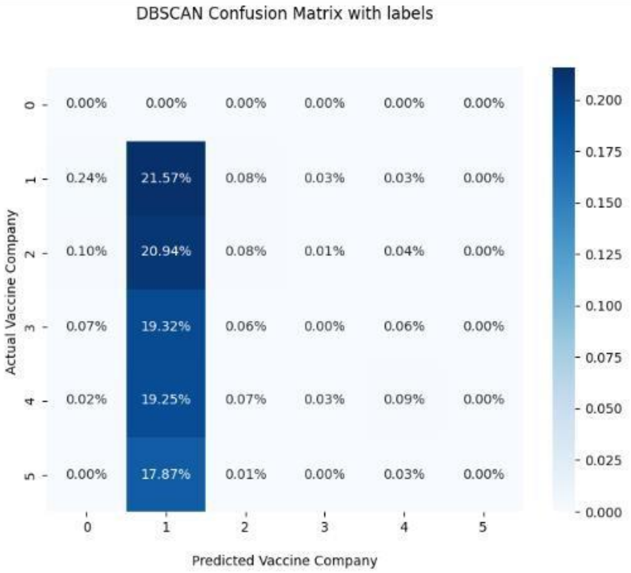


CLUSTERING ALGORITHMS:

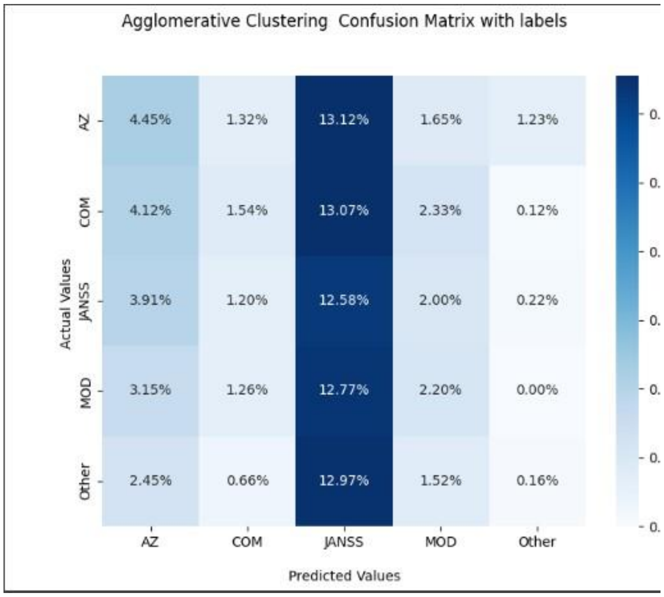
K-MEANS CLUSTERING:



DBSCAN CLUSTERING:



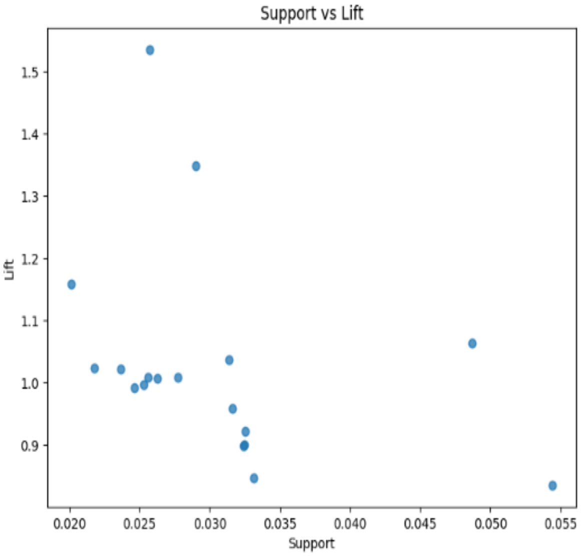
AGGLOMERATIVE CLUSTERING:



Agglomerative Clustering Confusion Matrix with labels

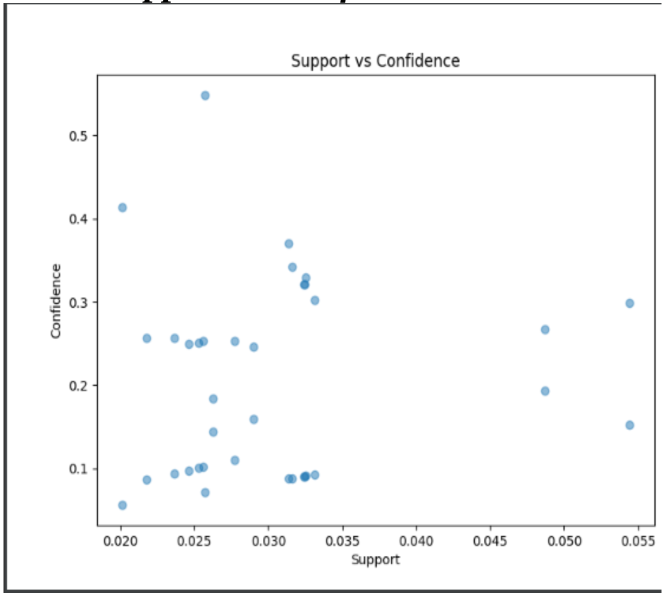
	AZ	COM	JANSS	MOD	Other
AZ	4.45%	1.32%	13.12%	1.65%	1.23%
COM	4.12%	1.54%	13.07%	2.33%	0.12%
JANSS	3.91%	1.20%	12.58%	2.00%	0.22%
MOD	3.15%	1.26%	12.77%	2.20%	0.00%
Other	2.45%	0.66%	12.97%	1.52%	0.16%

Support vs Lift

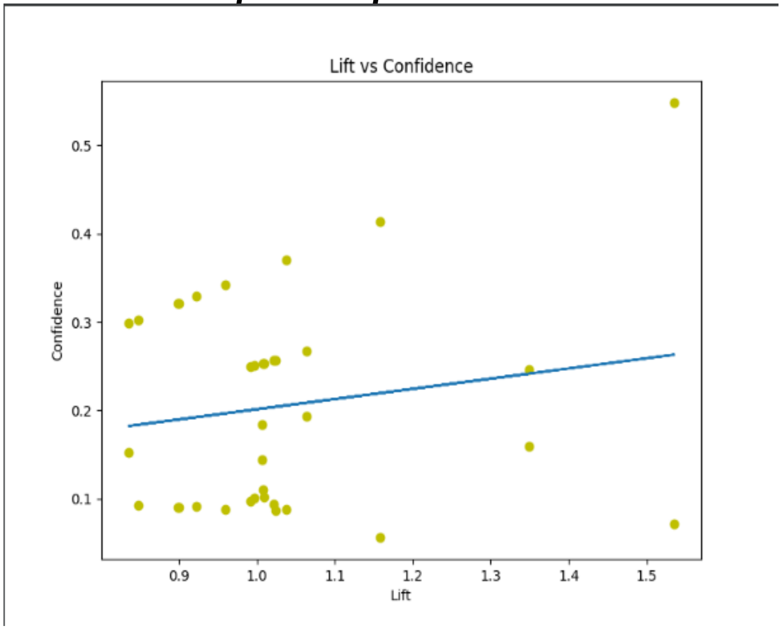


Apriori algorithm Diagrams:

Support vs Confidence

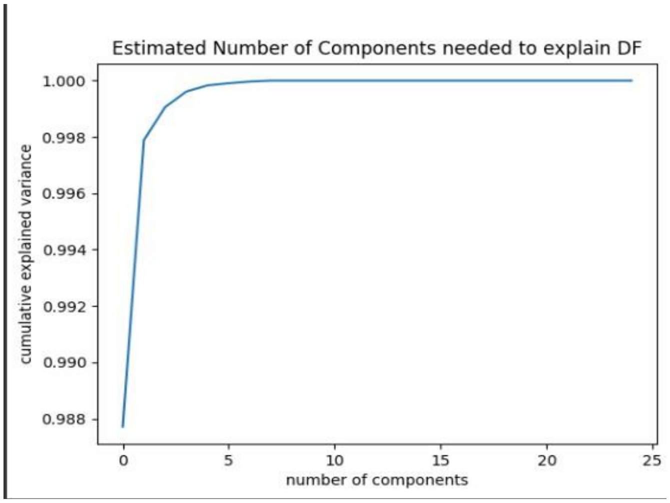


Lift vs Confidence



Select number of key features:

Below is the number of principal components that would be better to use based on the dataset we have. Therefore, nine different principal components were used during the dataset transformation using the dimensionality reduction method PCA.

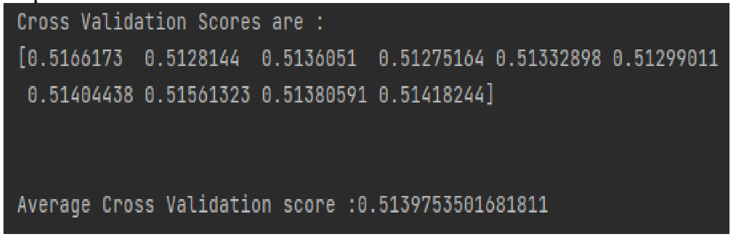


General observations

Initially, to avoid data disclosure before the implementation of the algorithms from previous implementation, the shuffle function has been used, as well as the random_state where necessary and allowed. Regarding the categorization, it seems that even with the preprocessing techniques that were used, we couldn't surpass the accuracy of 0.54. The main reason is that the prediction we ask from our model is the categorization of basic characteristics into five main categories. This difficulty hampers and reduces the efficiency of the categorization algorithm. An attempt was made to use only two vaccine companies where our data would be categorized, and we would achieve an accuracy reaching 0.9. This is desirable, but we would lose significant information, and the quality of the extracted information would be low. However, it seems that clustering the data into multiple clusters was a good choice to group our data, due to the way the information is distributed and the results we want are better represented by clustering techniques rather than categorization. Finally, regarding rule extraction, only those age groups that had received at least one dose were selected because it wouldn't make sense to extract rules between age groups and vaccines under different conditions (i.e., without dosage, there is no quality in the extracted knowledge in our opinion).

Additional material within the code

Regarding the avoidance of overfitting or underfitting during the categorization process, the K-Folds method was used. This method allows us to divide the dataset into n different subsets and apply the algorithms we are interested in analyzing for overfitting or underfitting separately to each subset. The implementation of the algorithm's data is found in a code comment section. Finally, the model seems not to detect any of the aforementioned problems. Below is an indicative implementation of K-Folds:



```
Cross Validation Scores are :
[0.5166173  0.5128144  0.5136051  0.51275164 0.51332898 0.51299011
 0.51404438 0.51561323 0.51380591 0.51418244]


Average Cross Validation score :0.5139753501681811
```

In which the relative accuracy of classification is stated, as well as the average value of the classification accuracy, which seems to have no significant differences, so we conclude that the aforementioned problems have been addressed.

REFERENCES

- https://scikit-learn.org/stable/
- https://www.datacamp.com/
- https://analyticsindiamag.com/
- https://realpython.com/
- https://www.youtube.com/