Mining frequents itemset and association rules in diabetic dataset

Youssef FAKIR[0000000213911052], Abdelfatah MAAROUF[0000-0003-3404-9403],

Rachid ELAYACHI[000000191440316]

Sultan Moulay Slimane University, Faculty of Sciences and Technics, Beni Mellal, Morocco

Email:info.dec07@yahoo.fr,abdelfatahmaarouf06@gmail.com, rachid.elayachi@usms.ma

Abstract— Data mining is a field of science to extract and analyses the information from large dataset. One of the most techniques is association rule mining. It aim is to find the relationship between the different attributes of data. Several algorithms for extracting data have been developed. Among the existing algorithms the FP-Growth algorithm is one of well-know algorithm in finding out the desired association rules. The aim of this paper is the extraction of association rules by FP-Growth algorithm and its variants using a diabetic dataset, which are the CFP-Growth and ICFP-Growth. Experimental results show that the ICFP-Growth is more accurate than CFP-Growth and FP-Growth.

Keywords— Data mining, Association rules, frequent patterns, FP-Growth, CFP-Growth, ICFP-Growth.

1. Introduction

Extraction of knowledge form large databases is important items in datamining. During the past decades several algorithms have been developed [1,2,3,4,5,6. In this paper, we are interested in the association rules algorithms, especially the FP-Growth [7,8] based algorithms and its variants such as CFP [9,10] and ICFP-Growth[11,12,13], by making a comparative study between these three algorithms. The ICFP is an improved version of the CPF-growth algorithm which consist of three steps: the construction of the Multiple Item Support Tree (MIS-Tree) [14,15], the extraction of the compact MIS-tree and mining the compact MIS-tree. The three algorithms FP-growth, CFP-Growth and ICFP-growth are implemented in order to compare their performances using Python 3 as programming language, vs code as IDE and windows 10 machine with 1.8 GHz and 8GB memory as environment. The dataset is the female's diabetes dataset (https://www.kaggle.com/mathchi/diabetes-data-set). It is divided into two '.csv' files the first for train dataset, and the other for the test dataset. The two '.csv' files contain 8 features:

- Pregnancies: Number of times pregnant.
- Glucose: Plasma glucose concentration 2 hours in an oral glucose tolerance test.
- Blood Pressure: Diastolic blood pressure (mm Hg).
- Skin Thickness: Triceps skin fold thickness (mm).
- Insulin: 2-Hour serum insulin (mu U/ml).
- BMI: Body mass index (weight in kg/(height in m)^2).
- Diabetes Pedigree Function: Diabetes pedigree function.
- Age: Age (years).

The paper is organized as follows. Section 2 describes the processes of database transformation. Section 3 deals with the extraction rules extraction. Performances evaluation of the algorithm are given in section 4 while section 5 concludes the paper.

2. Database transformation

The dataset contains just numerical values. The FP-Growth, CFP-Growth and ICFP-Growth accept the transactional datasets. The diabetes dataset (numerical datasets) is transformed into a transactional dataset. To do this transformation each feature is visualized in order to know how it change next to the number of individuals and to divide each feature into domains that regroups several individuals. The first feature is the age; the result of visualization is illustrated in Fig.1.

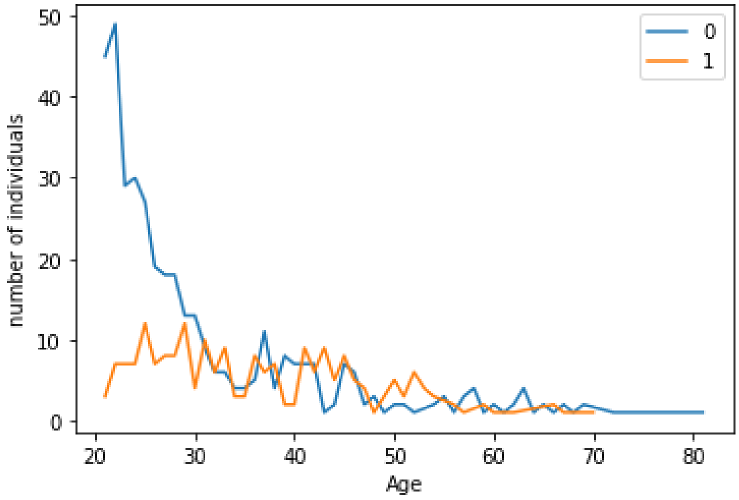


Fig.1: Age Vs number of individuals

As we can see in the range [20, 30] we have, hight number of no diabetes in comparison to the numbers of individuals with diabetes, and for the range [30, 80] we have allmost the same number of individuals for both classes 0 and 1 (0: without diabetes , 1: with diabetes), so we can just divide the range of the feature into two domains: A1 : [0, 30] and A2 : [30, 80].

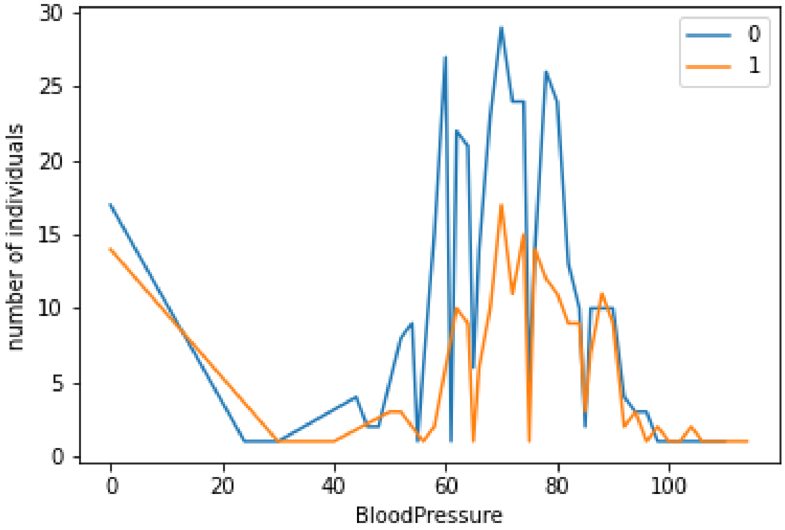


Fig.2: Blood Pressure Vs number of individuals

The second feature is the blood pressure, the result shown in the Fig.2. As we can see in the graph of the blood pressure, in the range [0, 40] we have the same variation for the two classes 0 and 1, and in [40, 90] the class 0 is highest than the 1 class, and in the range [90, 120] also we have the same variation of the classes 0 and 1, so we divide this feature to three domains: B1: [0, 40]; B2 : [40, 90]; B3: [90, 120]. The third feature is BMI, the result of visualization shown in Fig.3.

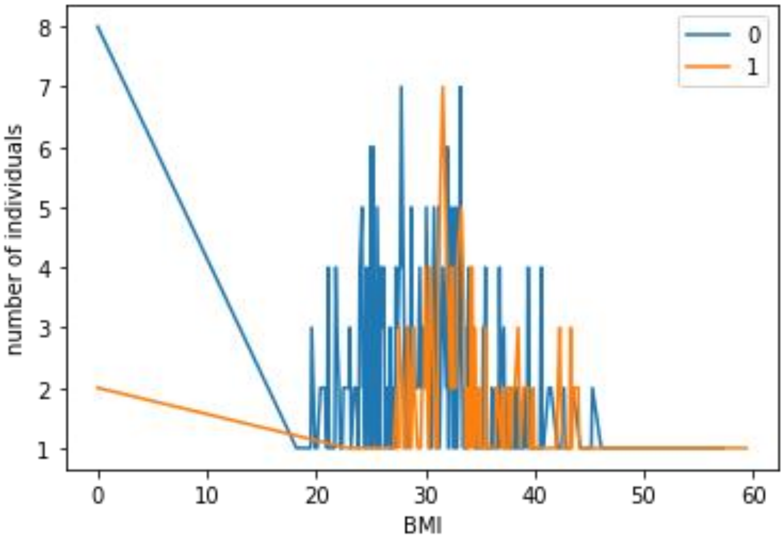


Fig.3: BMI vs number of individuals

As we see in this graph, we can divide the range of the BMI feature into two domains the first BMI1: [0, 30], where we have individuals' number of the class 0 highest than the 1 class, and the second is BMI2: [30, 60] where the two classes have almost the same variation.

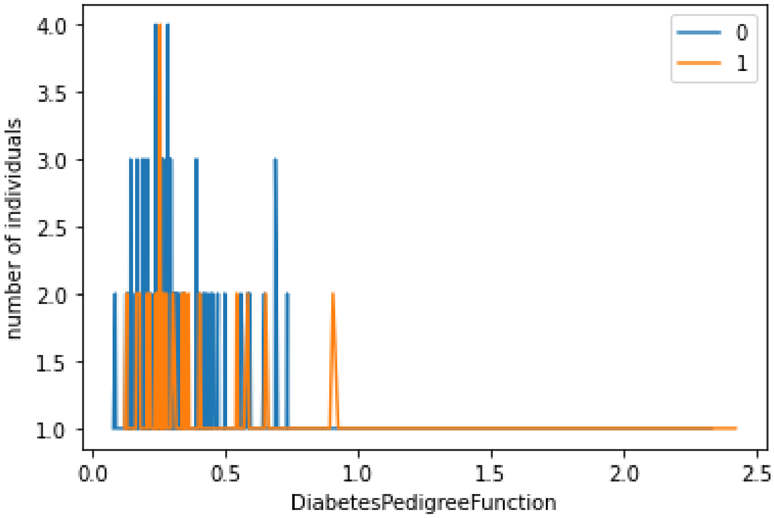


Fig.4: Diabetes PredgreeFunction vs number of individuals

The fourth feature is the Diabetes Pedigree Function, the visulisation is in the Fig.4. In this figure we can see in [0, 0.8] the 0 class have almost the highest number of individuals than the 1 class, and for the range [0.8, 2.5] the opposite, the class 1 have the highest number of individuals, therefore we can divide the feature into two domains: D1: [0, 0.8] and D2: [0.8, 2.5]. The fifth feature is the Glucose, the result of visualization illustrated in the Fig.5.

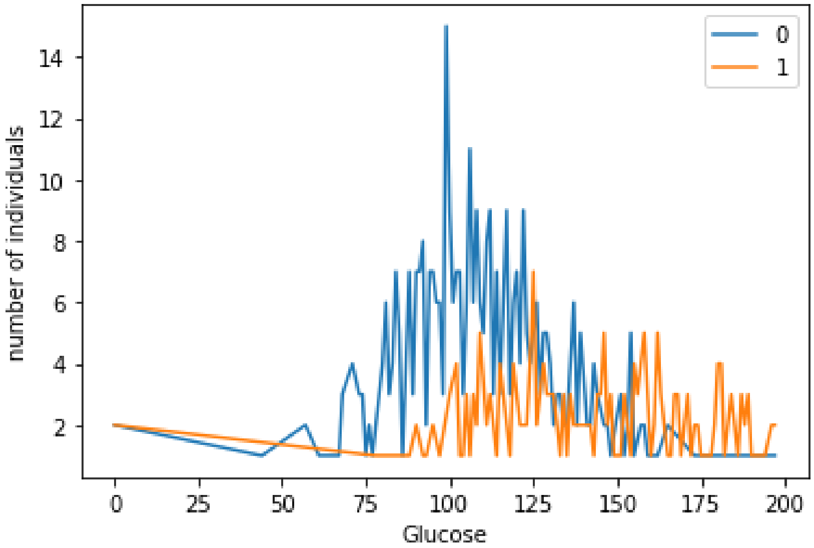


Fig.5: Glucose Vs number of individuals

As we can see in the range [0, 125] we have, hight number of the class 0 in comparison to the numbers of individuals of the class 1, and for the range [125, 200] we have allmost the same number of individuals for both classes 0 and 1, so we can divide the range of the feature into two domains: G1 : [0, 125] and G2 : [125, 200].

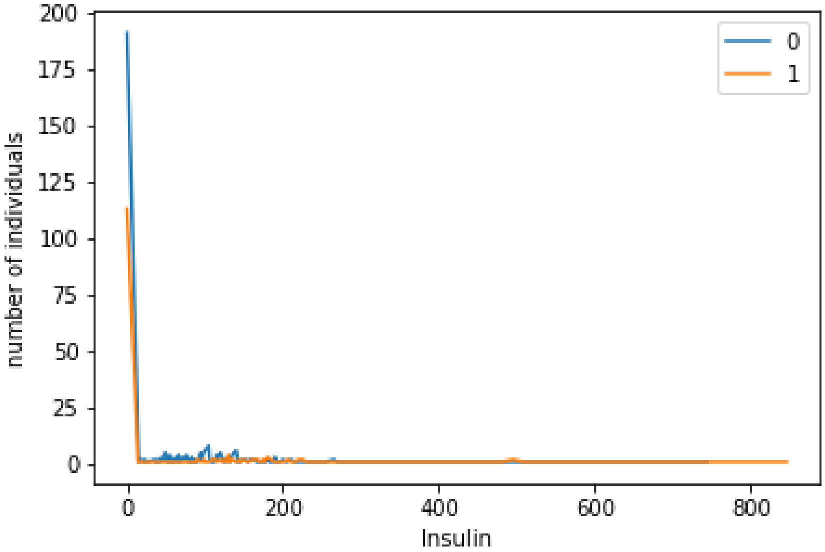


Fig.6: Insulin Vs number of individuals

The sixth feature is the Insulin, the result shown in the Fig.6. In the graph of the insuline vs number of individuals in the range [0, 30] we have almost the same variation for the two classes 0 and 1, and in [30, 150] the class 0 is highest than the 1 class, and in the range [150, 800] also we have the same variation of the classes 0 and 1, so we divide this feature to three domains: I1: [0, 30]; I2 : [30, 150]; I3: [150, 800]. The seventh feature is the Pregnancies, the result of visualization is shown in the Fig.7.

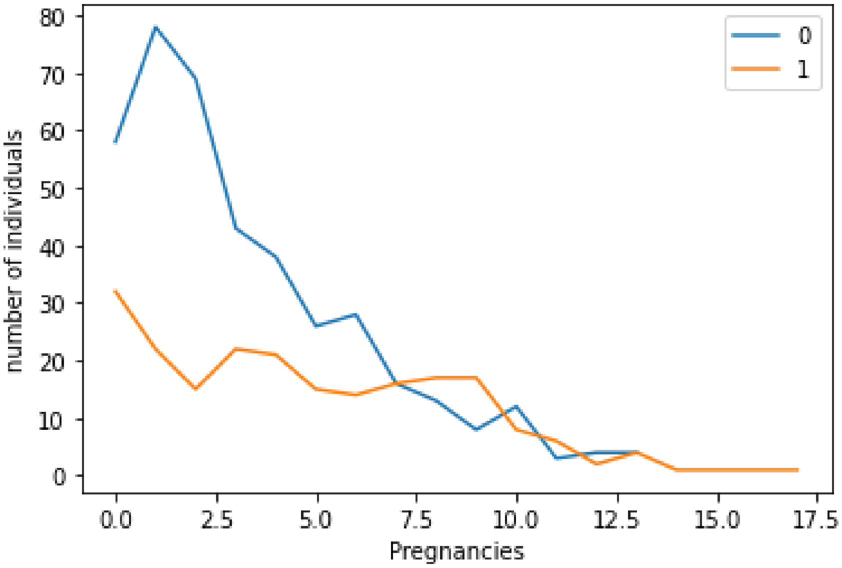


Fig.7: Pregnancies VS number of individuals

As we see in this graph, we can divide the range of the Pregnancies feature into two domains the first P1: [0, 7], where we have individuals' number of the class 0 highest than the 1 class, and the second is P2: [7, 17] where the two classes have almost the same variation. The last feature is the SkinThickness, the visulisation is in the Fig.8.

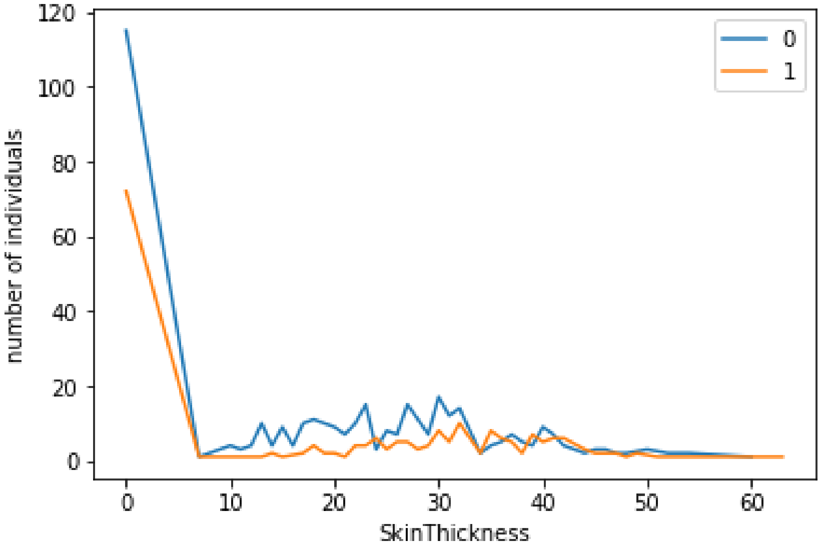


Fig.8: Skin thickness VS number of individuals

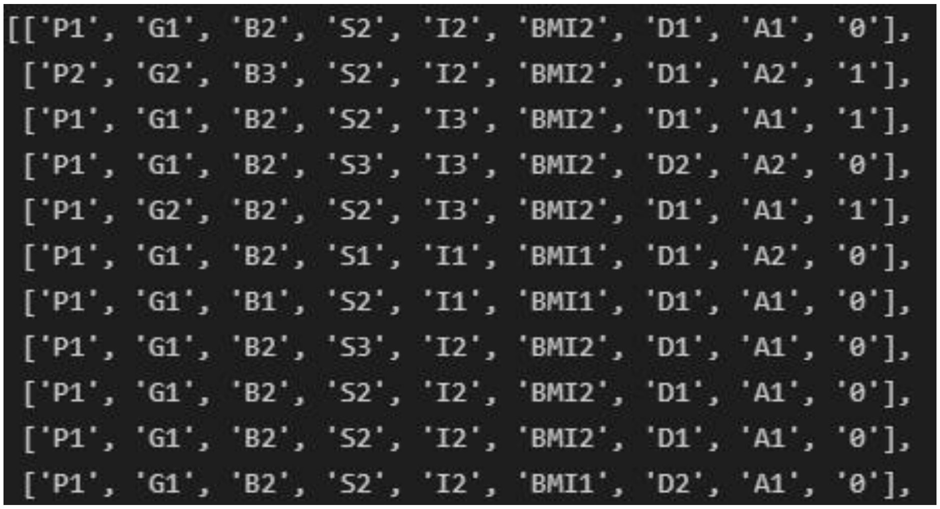
In this figure we can see in [0, 8], the proximately the same variation for the two classes, and in [8, 45] the 0 class have almost the highest number of individuals than the 1 class. In addition, for the range [45, 60] also we have the same variation for the both classes, therefore we can divide the feature into three domains: S1: [0, 8], S2: [8, 45] and S3: [45, 60].

After all these analysis, we can briefly resume all the information in the Table 1.

Table 1: result of transformation

feature	domains
Age	A1: [0, 30]; A2: [30, 80]
Pregnancies	P1: [0, 7]; P2: [7, 17]
Glucose	G1: [0, 125]; G2: [125, 200]
Blood Pressure	B1: [0, 40]; B2: [40, 90]; B3: [90, 120]
Skin Thickness	S1: [0, 8]; S2: [8, 45]; S3: [45, 60]
Insulin	I1: [0, 30]; I2: [30, 150]; I3[150, 800]
Diabetes Pedigree Function	D1: [0, 0.8]; D2: [0, 2.5]
BMI	BMI1: [0, 30]; BMI2: [30, 60]

Now we can use the domains to transform the dataset to a transactional dataset. As we can see in the Fig.9, we have a part of result that we obtained from the transformation.



```
[['P1', 'G1', 'B2', 'S2', 'I2', 'BMI2', 'D1', 'A1', '0'],
 ['P2', 'G2', 'B3', 'S2', 'I2', 'BMI2', 'D1', 'A2', '1'],
 ['P1', 'G1', 'B2', 'S2', 'I3', 'BMI2', 'D1', 'A1', '1'],
 ['P1', 'G1', 'B2', 'S3', 'I3', 'BMI2', 'D2', 'A2', '0'],
 ['P1', 'G2', 'B2', 'S2', 'I3', 'BMI2', 'D1', 'A1', '1'],
 ['P1', 'G1', 'B2', 'S1', 'I1', 'BMI1', 'D1', 'A2', '0'],
 ['P1', 'G1', 'B1', 'S2', 'I1', 'BMI1', 'D1', 'A1', '0'],
 ['P1', 'G1', 'B2', 'S3', 'I2', 'BMI2', 'D1', 'A1', '0'],
 ['P1', 'G1', 'B2', 'S2', 'I2', 'BMI2', 'D1', 'A1', '0'],
 ['P1', 'G1', 'B2', 'S2', 'I2', 'BMI2', 'D1', 'A1', '0'],
 ['P1', 'G1', 'B2', 'S2', 'I2', 'BMI1', 'D2', 'A1', '0'],
```

Fig. 9: Part of transformation

3. Extraction of association rules

First, we have to initialize the minsupport for FP-Growth [17], and MIS-values for CFP-Growth and ICFP-Growth. To assign MIS-values for CFP-Growth, we use the equation (2).

$$MIS(i) = maxinum(\beta f(i), LS) \quad (2)$$

- MIS (i) is the MIS-value of the item "i".
- $\beta \in [0, 1]$ is a parameter that controls how the MIS values for items should be related to their frequencies.

- $f(i)$ is the frequencies value for the item “i”.
- LS is a use specified value, represent the least minimum support allowed.

In addition, for the ICFP-Growth we use the equation (3).

$$MIS(i) = \begin{cases} M(i) \ \ if \ M(i) > LMS \\ LMS \ else \ if \ M(i) < LMS \ and \ S(i) > LMS \\ LMIS \ else \end{cases} \qquad (3)$$

with

$$M(i) = S(i) - SD$$

where

- $SD \in$ [0, 1] is a user specified value.
- $S(i) = \frac{f(i)}{N}$ is the support of the item “i”.
- $f(i)$ is the frequencies value for the item “i”.
- N represent the number of transactions in the dataset.
- LMS is a user specified value, stand for lowest minimum support, and represent lowest MIS value of a frequent item
- LMIS is also a user specified value, stand for least minimum item support, and represent the lowest MIS value among all items in the transaction dataset.
- The LMIS value should always be less than or equal to LMS.

In this experiment, we define the minsupport of the FP-Growth equal to 40, for the CFP-Growth the β equal to 0.1 and *LS* equal to 40, and for the ICFP-Growth, the *SD* value is 0.1, LMS equal to 50 and LMIS equal to 40.
The result of the MIS-values generation, for the CFP-Growth is shown in table 2. In addition, for ICFP-Growth the result of MIS-values initialization is given in the table 3.

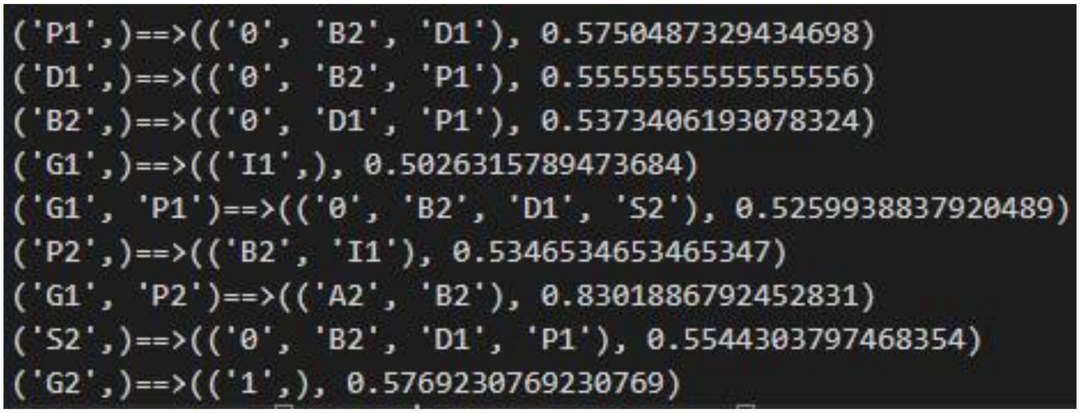
Table 2: MIS value for CFP-Growth algorithm

item	P1	G1	B2	S2	I2	BMI2	D1	A1	0	P2	G2	B3	A2	1	I3	S3	D2	S1	BMI1	B1
MIS-value	51	40	55	40	40	40	53	40	40	40	40	40	40	40	40	40	40	40	40	40

Table 3: MIS value for ICFP-Growth algorithm

item	P1	G1	B2	S2	I2	BMI2	D1	A1	0	P2	G2	B3	A2	1	I3	S3	D2	S1	BMI1	B1
MIS-value	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40	40

After we apply the three algorithms on our dataset, we obtain three models that contains the association rules as shown in Fig.10.



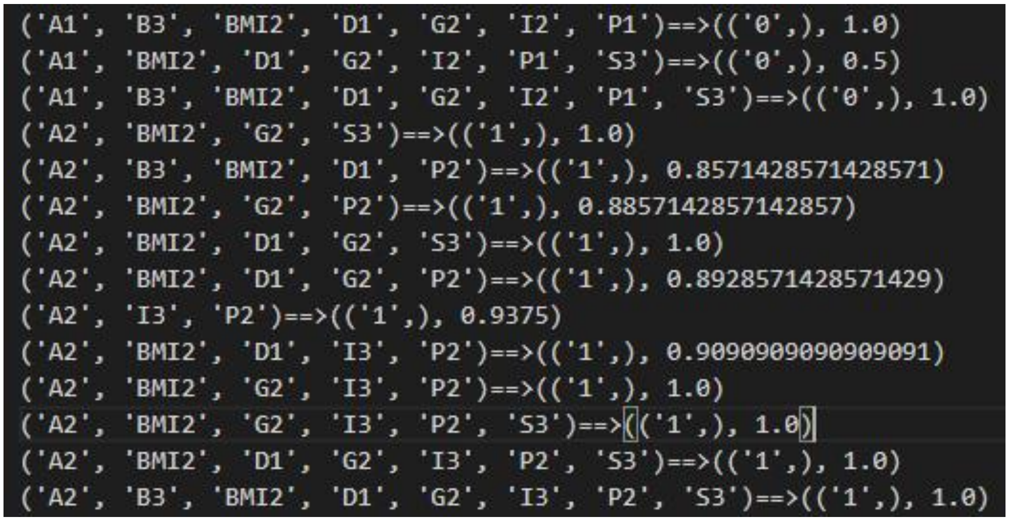
```
('P1',)==>(('0', 'B2', 'D1'), 0.5750487329434698)
('D1',)==>(('0', 'B2', 'P1'), 0.5555555555555556)
('B2',)==>(('0', 'D1', 'P1'), 0.5373406193078324)
('G1',)==>(('I1',), 0.5026315789473684)
('G1', 'P1')==>(('0', 'B2', 'D1', 'S2'), 0.5259938837920489)
('P2',)==>(('B2', 'I1'), 0.5346534653465347)
('G1', 'P2')==>(('A2', 'B2'), 0.8301886792452831)
('S2',)==>(('0', 'B2', 'D1', 'P1'), 0.5544303797468354)
('G2',)==>(('1',), 0.5769230769230769)
```

Fig. 10. Association rules obtained

The structure of our model is: (left) ➔(right, Confidence). The left is the causes and the right is the consequence. Confidence is a number in range of [0, 1] that can represent how much left can lead us to the right, who much the causes can lead as to a consequence, and we can use the equation (4) to calculate the confidence [16] .

$$Confidence(right \text{ ➔ left}) = \frac{support(right \cup left)}{support(right)} \quad (4)$$

Fig.11 shows the association rules between all the features, but in our case, we want to do a classification model, for that we filter the association rules to have in the consequences (right) just the items that represent the classes (‘0’ and ‘1’), the result is shown in Fig.11.



```
('A1', 'B3', 'BMI2', 'D1', 'G2', 'I2', 'P1')==>(('0',), 1.0)
('A1', 'BMI2', 'D1', 'G2', 'I2', 'P1', 'S3')==>(('0',), 0.5)
('A1', 'B3', 'BMI2', 'D1', 'G2', 'I2', 'P1', 'S3')==>(('0',), 1.0)
('A2', 'BMI2', 'G2', 'S3')==>(('1',), 1.0)
('A2', 'B3', 'BMI2', 'D1', 'P2')==>(('1',), 0.8571428571428571)
('A2', 'BMI2', 'G2', 'P2')==>(('1',), 0.8857142857142857)
('A2', 'BMI2', 'D1', 'G2', 'S3')==>(('1',), 1.0)
('A2', 'BMI2', 'D1', 'G2', 'P2')==>(('1',), 0.8928571428571429)
('A2', 'I3', 'P2')==>(('1',), 0.9375)
('A2', 'BMI2', 'D1', 'I3', 'P2')==>(('1',), 0.9090909090909091)
('A2', 'BMI2', 'G2', 'I3', 'P2')==>(('1',), 1.0)
('A2', 'BMI2', 'G2', 'I3', 'P2', 'S3')==>(('1',), 1.0)
('A2', 'BMI2', 'D1', 'G2', 'I3', 'P2', 'S3')==>(('1',), 1.0)
('A2', 'B3', 'BMI2', 'D1', 'G2', 'I3', 'P2', 'S3')==>(('1',), 1.0)
```

Fig.11: Results of association rules

In the figure we have the association for the classification model for example we have this association rule (‘A2’, ‘BMI2’, ‘G2’, ‘P2’) ➔ ((‘1’), 0.89).

That mean if the individual has A2 the age between [30, 80]. The BMI2 the Body mass index in range of [30, 60], the G2 Plasma glucose concentration 2 hours in an oral glucose tolerance test in range of [125, 200], and the P2 number of times pregnant between [7, 17], so we can see that the individual has a diabetes with the confidence of 0.89.

4. Performances evalution

The three algorithms FP-Growth, CFP-Growth and ICFP-Growth are evaluated using the same preprocessing that we were applied on the train dataset, to transform the numerical dataset into a transactional dataset. After that, we take a transaction from the dataset and calculate the distance between the test transaction and the left of the association’s rules of the model. In this case, we use an approach to calculate the distance. For example, we have T test transaction and G the left of an association rule that exist in the model.

T = [‘P1’, ‘G1’, ‘B2’, ‘S2’, ‘I3’, ‘BMI2’, ‘D2’, ‘A2’],

G = ['A1', 'B3', 'BMI2', 'D1', 'G2', 'I2', 'P1']

First, we have eight features in datasets. We initialized the distance by 8, and we check for each item of the T if it exist in the G and for each item exist we decrement the distance by one. In this example, we have P1 exist in the T and G, so the distance is 7. In addition, G1 not exist in G and so we still have distance is 7, and B2, S2, I3, D2, A2 also doesn’t exist in G, and we have BMI2 exist so we decrement the distance by 1 so we have the distance equal to 6, the distance between G and T is 6. In addition, after calculating the distances, we chose the three closet association rules, we count who much votes for the ‘0’ and for ‘1’, and we choose the class that have the highest number of votes. After this testing process, we calculate the accuracy for each algorithm. The accuracy for the three algorithms FP-Growth, CFP-Growth and ICFP-Growth respectively is 51.30%, 57% and 60.5%.

5. Conclusion

Frequent itemset mining is an important subject in data mining. In this paper, three association rules algorithms, which are FP-Growth, CFP-Growth, and ICFP-Growth, are implemented. These algorithms are used to extract item sets frequents on a diabetes dataset using python programming language. Experimental results show that the ICFP is more accurate than the others algorithms.

REFERENCES

1. Jiawei Han, Jian Pei, and Yiwen Yin, ‘Mining frequent patterns without candidate generation’, in Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD ’00, Dallas, Texas, United States, 2000, pp. 1–12.
2. Youssef FAKIR, et al., A Comparative Study between Relim and SaM Algorithms, International Journal of Computer Science and Information Security (IJCSIS),Vol. 18, No. 5, May 2020
3. Ya-Han Hu and Yen-Liang Chen, ‘Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism’, Decision Support Systems, vol. 42, no. 1, pp. 1–24, Oct. 2006.
4. JV Joshua et al., ‘Data Mining: A book recommender system using frequent pattern algorithm’, Journal of Software Engineering and Simulation, vol. 3, no. 3, pp. 01–13, 2016.

5. Shikha Pathania and Harpreet Singh, ‘A New Associative Classifier Based on CFP-Growth++ Algorithm’, in Proceedings of the Sixth International Conference on Computer and Communication Technology 2015, New York, NY, USA, 2015, pp. 20–25.
6. R. Uday Kiran and P. Krishna Reddy, ‘Novel techniques to reduce search space in multiple minimum supports-based frequent pattern mining algorithms’, in Proceedings of the 14th International Conference on Extending Database Technology - EDBT/ICDT ’11, Uppsala, Sweden, 2011, p. 11.
7. Youssef Fakir, et al., Closed frequent itemsets mining based on It-Tree, Global Journal of Computer Sciences: Theory and Research, Volume 11, Issue 1, 2021.
8. Youssef Fakir, Rachid El Ayachi, Mohamed Fakir, Mining Frequent Pattern by Titanic and FP-Tree algorithms, International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Vol.6, Issue 5, 2020.
9. R. Uday Kiran and P. Krishna Reddy, 'An Improved Frequent Pattern-Growth Approach to Discover Rare association rules’:, in Proceedings of the International Conference on Knowledge Discovery and Information Retrieval, Funchal - Madeira, Portugal, 2009, pp. 43–52.
10. Guimei liu, et al, CFP-tree: a compact disk-based structure for storing and querying frequent itemsets, Information Systems, Volume 32, Issue 2, April 2007, Pages 295-319.
11. Sadeq Darraba , Belgin Ergenc, Vertical Pattern Mining Algorithm for Multiple Support Thresholds, International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2017, 6-8 September 2017, Marseille, France.
12. Shafiul Alom Ahmed, Bhabesh Nath, ISSP-tree: An improved fast algorithm for constructing a complete prefix tree using single database scan, Expert Systems with Applications, Volume 185, 15 December 2021, 115603.
13. Wensheng Gan, Mining of frequent patterns with multiple minimum supports, Engineering Applications of Artificial Intelligence, Volume 60, April 2017, Pages 83-96
14. Chen Xingxing, Huang Hongbin and Du Wenya, Most frequent item sets mining algorithm based on MIS-Tree and multipl support array, International journal of research in engineering and sciences, Vol.4, issue 7, 2016.
15. M.Sinthuja, S.Sheeba Rachel and G.Janani, Mis-Tree algorithm for mining association ruls with multiple minimum support, Bonfring international journal of datamining, December 2021
16. Y. Fakir and R. El Ayachi, Frequent Patterns Mining, International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Vol.6, Issue 4, 2020.
17. Meera Narvekara , Shafaque Fatma Syed, An optimized algorithm for association rule mining using FP tree, nternational Conference on Advanced Computing Technologies and Applications (ICACTA-2015), Procedia Computer Science 45 (2015) 101 – 110.