Homework 3

Kartik Joshi

December 15, 2023

Contents

1 Introduction

In this paper, I introduce the implementation of a Lazy FCA classification algorithm based on pattern structures. The proposed algorithm was compared

with baseline Lazy FCA and other popular models(Decision tree, ,Random forest ,xGboost ,k-NN ,Naive Bayes ,logistic regression) using three datasets:

- `Telecom_Curn`
`https://www.kaggle.com/datasets/blastchar/telco-customer-churn`;
- `Dry Bean Dataset`
`https://archive.ics.uci.edu/dataset/602/dry+bean+dataset`;
- `In-vehicle coupon recommendation dataset`
`https://archive.ics.uci.edu/dataset/603/in+vehicle+coupon+recommendation`.

You can find all code in my GitHub repository https://github.com/kartikj360/LazyFCA. Code base structure

1. `LazyFCA/In_vehicle_coupon_recommendation.ipynb` – notebook with comparison of base algorithm with proposed on vehicle coupon selection;
2. `LazyFCA/Dry_Bean_Dataset.ipynb` – notebook with comparison of base algorithm with proposed one on dry bean classification prediction data-set and comparison with popular models;
3. `LazyFCA/Telecom_Curn.ipynb` – notebook with comparison of proposed algorithm with popular models on telecom company data dataset;
4. `LazyFCA/custom_cross_val.py` – custom made cross-validation approach for LazyFCA model for both binarized classification and for pattern structure approach.
5. $\texttt{data}_set --folderwithdatasets$
6. `images` – folder of plots images.

All other files were taken from base repository or just utils for developing.

2 Proposed algorithm

We are given the data $X =$
$x_1, x_2, x_3, ...$ and the corresponding labels Y where each label $y \in Y$ is either

True of False. The task is to make a "prediction" $\hat{y}$ of a label $y \in Y$ for each datum $x \in X$ as if y is unknown.

To estimate the quality of predictions we split the data X into two non-overlapping sets X_{train} and X_{test}. Then we make make a prediction $\hat{y}$ for each test datum $x \in X_{test}$ based on the information obtained from the training data X_{train}. Finally, we measure how well predictions $\hat{y}$ represent the true test labels y.

The original data X often comes in various forms: numbers, categories, texts, graphs, etc. Throughout the course of this assignment, you are going to work with two types of data: scaled (binarized) data, which you will obtain from preprocessing your datasets, and non-binarized data.

2.1 Baseline algorithm

Assume that we want to make a prediction for description $x \subseteq M$ given the set of training examples $X_{train} \subseteq 2^M$ and the labels $y_x \in$
$False, True$ corresponding to each $x \in X_{train}$.

First, we split all examples X_{train} to positive X_{pos} and negative X_{neg} examples:

$$X_{pos} = x \in X_{train} \mid y_x \text{ is True}, \quad X_{neg} = X \setminus X_{pos}.$$

To classify the descriptions x we follow the procedure: 1) For each positive example $x_{pos} \in X_{pos}$ we compute the intersection $x \cap x_{pos}$. Then, we count the support in positive and negative classes for this intersection, that is the number of respectively positive and negative examples $x_{train} \in X_{train}$ containing intersection $x \cap x_{pos}$;

2) Dually, for each negative example $x_{neg} \in X_{neg}$ we compute the intersection $x \cap x_{neg}$. Then, we count the support in negative and positive classes for this intersection, that is the number of respectively negative and positive examples $x_{train} \in X_{train}$ containing intersection $x \cap x_{neg}$;.

Finally, we plug the obtained values of support into decision function and classify x based on them. There are three parameterized methods for your choice: 1) "standard" we consider number of all alpha-weak hypothes and classify based on it:

$$y = \frac{\sum\limits_{x_{pos} \in X_{pos}} [b_{x_{pos}} \leq \alpha * |X_{neg}|]}{|X_{pos}|} > \frac{\sum\limits_{x_{neg} \in X_{neg}} [b_{x_{neg}} \leq \alpha * |X_{pos}|]}{|X_{neg}|}$$

2) "standard-support" we consider number of all alpha-weak hypothes with their support and classify based on it:

$$y = \frac{\sum\limits_{x_{pos} \in X_{pos}} a_{x_{pos}}[b_{x_{pos}} \leq \alpha * |X_{neg}|]}{|X_{pos}|^2} > \frac{\sum\limits_{x_{neg} \in X_{neg}} a_{x_{neg}}[b_{x_{neg}} \leq \alpha * |X_{pos}|]}{|X_{neg}|^2}$$

3) "ratio-support" we consider the ratio of support in target class and in opposite one for hypotheses which support in target class is α times higher than in other:

$$y = \underset{k}{argmax} \left(\frac{|X_{train} \backslash X_{c_k}| \cdot \sum_{x_i \in X_{c_k}} a_i \cdot [\frac{a_i}{|X_{c_k}|} \geq \alpha \frac{b_i}{|X_{train} \backslash X_{c_k}|}]}{|X_{c_k}| \cdot \sum_{x_i \in X_{c_k}} b_i \cdot [\frac{a_i}{|X_{c_k}|} \geq \alpha \frac{b_i}{|X_{train} \backslash X_{c_k}|}]} \right)$$

where a_{x_k} is support in class k, and b_{x_k} is support in the opposite class, of the intersection $x \cap x_k$.

2.2 Comparison of models base algorithm

Our model is compared with existing state of the art models and the results compared over with out model provide some major drawbacks and advantages to use of LAZY FCA, as being a lazy learning approach , it always have a benefit of having to work continuously on newly streamed data and provide an ace in the Assembly. The exist version are comparatively slow and non optimized in a manner. The amount of time consumed to what the existing state of the art models have achieved is comparable. While working on the 3 data set, I had to reduce to just make out some minor results and at times is tedious for online virtual machines with limited resources. Keep time aside we would be comparing out the performances of the models on different metrics to have optimum comparison. The models are compared out on there Accuracy which is one of the most basic method to check the viable of the model, but we have other metrics like recall , precision and F1 score. Not only sticking to the metrics for evaluation of the models, we used cross validation techniques like K fold(10 folds for all model) and scarified shuffling of data to avoid bias in data. Due to cutting down the data to process faster, there could be a possibility of the minor bias ,but can we removed on the provision of higher computing power.

3 Telecom Churn data

3.1 About Data-set

"Predict behavior to retain customers. You can analyze all relevant customer data and develop focused customer retention programs." [IBM Sample Data Sets]

Content Each row represents a customer, each column contains customer's attributes described on the column Metadata.

The data set includes information about:

Customers who left within the last month – the column is called Churn Services that each customer has signed up for – phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies Customer account information – how long they've been a customer, contract, payment method, paperless billing, monthly charges, and total charges Demographic info about customers – gender, age range, and if they have partners and dependents

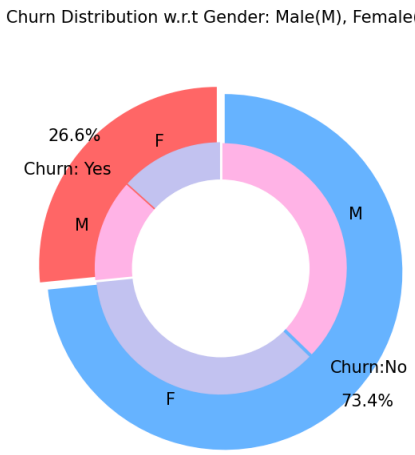

Figure 1: Caption for the first image

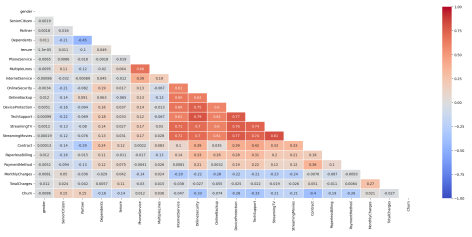

Figure 2: Caption for the second image

Figure 3: Overall caption for both images

3.2 Results and pre-processing

This given data-set exist in comparatively cleaner form and require minimum to no pre-cleaning and pre-processing except categorical attributes conversion and binarization , For categorical fields it is quite easy, we will use Sklearn LabelEncoder() to transform the values into numeric for. Some of the major issue was faced on will processing the data are defining ranges of the bins to divide the objects for particular attributes.We used strategy and set thresholds with 0.25, 0.50, 0.75 quantiles we used the pd.qcut() for the following task and I had a option to use the pd.cut() function, but for this I had to either part my data with bias or have to define specific bins, this task is at time laborious so I implemented the pd.qcut() function, the categories were then further divided into binary values by pd.get_dummies() to have the binarazied values and then converting them to bool datatype for to fed into Binarize classifier.

For test we will use 30% of data and 70% for training, the other categorical attributes. The results of Cross validation of respective models are given in the following table.

Table 1: Model Evaluation Metrics

Model	**Precision**	**Recall**	**F1 Score**	**Mean Accuracy**
K-NN	0.83	0.85	0.84	0.75
Random Forest	0.84	0.90	**0.87**	0.80
Logistic Regression	0.85	0.86	0.86	0.79
Decision Tree	0.82	0.80	0.81	0.72
XGBClassifier	0.83	0.91	**0.87**	0.75
GaussianNB	**0.89**	0.73	0.80	0.78
BinaryClassifier(Standard)	0.79	0.88	0.83	0.81
BinaryClassifier(Standard-Support)	0.79	0.89	0.83	0.81
BinaryClassifier(Ratio Support)	0.72	**1**	0.83	0.79
PatternClassifier(with binarized data)	0.79	0.93	0.85	0.81
PatternClassifier	0.80	0.90	0.85	**0.96**

You can see results on table 1. From this we can see that proposed algorithm peaks on the accuracy with 81% for all the proposed classifier. We should note that choosing numerical processing strategy hardly effect model scores and there are a huge field for testing different methods(Very minor for

that of Ratio support for the given data set). Also we can note unpredictable time consuming on every prediction because it is hardly depend on how fast we can fiend appropriate extent.

More interesting results we can get from comparing best models from previous step with popular classification models: Random forest, Logistic Regression and DT , XGB classifier and Gaussina NB. In this step we won't compare time because sklearn perfect written algorithms with C++ boosting of course work faster, but that is quite interesting to note that proposed models have comparable accuracy and F1 scores. Also for this data-set we mainly want better recall score, because we want to have as much positive labels as possible (so that the telecom company should know they should improvise the services for a certain customer) and for this purpose our algorithms performed good but they have quite nice recall results. Moving towards the results of the pattern classifier , we got some of the best recall and accuracy score, highest in all the proposed and classical classifiers. We can clearly see that only the pattern classifier provides outstanding performs for the given telecom set.

4 In Vehicle Coupon data

4.1 About Data-set

Source: Tong Wang, tong-wang '@' uiowa.edu, University of Iowa
Cynthia Rudin, cynthia '@' cs.duke.edu, Duke University

Data Set Information:

This data was collected via a survey on Amazon Mechanical Turk. The survey describes different driving scenarios including the destination, current time, weather, passenger, etc., and then ask the person whether he will accept the coupon if he is the driver. For more information about the data-set, please refer to the paper: Wang, Tong, Cynthia Rudin, Finale Doshi-Velez, Yimin Liu, Erica Klampfl, and Perry MacNeille. 'A Bayesian framework for learning rule sets for interpretable classification.' The Journal of Machine Learning Research 18, no. 1 (2017): 2357-2393.

Attribute Information:

1. destination: No Urgent Place, Home, Work

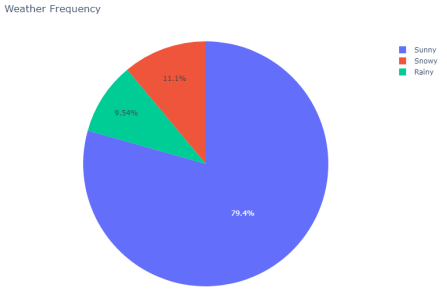

Figure 4: Caption for the first image

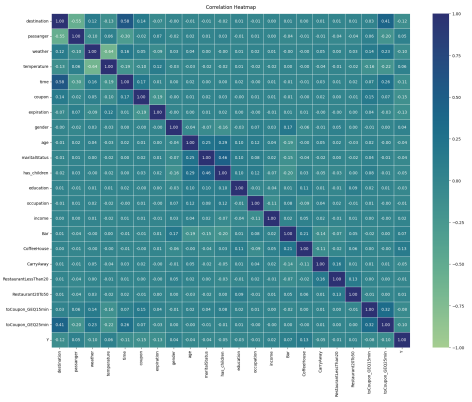

Figure 5: Caption for the second image

Figure 6: Overall caption for both images

2. passanger: Alone, Friend(s), Kid(s), Partner (who are the passengers in the car)
3. weather: Sunny, Rainy, Snowy
4. temperature: 55, 80, 30
5. time: 2PM, 10AM, 6PM, 7AM, 10PM
6. coupon: Restaurant (lesser 20), Coffee House, Carry out & Take away, Bar, Restaurant (20-50)
7. expiration: 1d, 2h (the coupon expires in 1 day or in 2 hours)
8. gender: Female, Male
9. age: 21, 46, 26, 31, 41, 50plus, 36, below21
10. maritalStatus: Unmarried partner, Single, Married partner, Divorced, Widowed
11. has_Children: 1, 0

12. education: Some college - no degree, Bachelors degree, Associates degree, High School Graduate, Graduate degree (Masters or Doctorate), Some High School

13. occupation: Unemployed, Architecture & Engineering, Student, Education & Training & Library, Healthcare Support, Healthcare Practitioners & Technical, Sales & Related, Management, Arts Design Entertainment Sports & Media, Computer & Mathematical, Life Physical Social Science, Personal Care & Service, Community & Social Services, Office & Administrative Support, Construction & Extraction, Legal, Retired, Installation Maintenance & Repair, Transportation & Material Moving, Business & Financial, Protective Service, Food Preparation & Serving Related, Production Occupations, Building & Grounds Cleaning & Maintenance, Farming Fishing & Forestry

14. income: $37500 - $49999, $62500 - $74999, $12500 - $24999, $75000 - $87499, $50000 - $62499, $25000 - $37499, $100000 or More, $87500 - $99999, Less than $12500

15. Bar: never, less1, 1-3, gt8, nan4-8 (feature meaning: how many times do you go to a bar every month?)

16. CoffeeHouse: never, less1, 4-8, 1-3, gt8, nan (feature meaning: how many times do you go to a coffeehouse every month?)

17. CarryAway: n4-8, 1-3, gt8, less1, never (feature meaning: how many times do you get take-away food every month?)

18. RestaurantLessThan20: 4-8, 1-3, less1, gt8, never (feature meaning: how many times do you go to a restaurant with an average expense per person of less than $20 every month?)

19. Restaurant20To50: 1-3, less1, never, gt8, 4-8, nan (feature meaning: how many times do you go to a restaurant with average expense per person of $20 - $50 every month?)

20. toCoupon_GEQ15min: 0,1 (feature meaning: driving distance to the restaurant/bar for using the coupon is greater than 15 minutes)

21. toCoupon_GEQ25min: 0, 1 (feature meaning: driving distance to the restaurant/bar for using the coupon is greater than 25 minutes)

22. direction_same: 0, 1 (feature meaning: whether the restaurant/bar is in the same direction as your current destination)

23. direction_opp: 1, 0 (feature meaning: whether the restaurant/bar is in the opposite direction as your current destination)

Class Label: Y: 1, 0 (whether the coupon is accepted)

4.2 Results and pre-processing

In the data-set we had a very large number of categorical attributes to deal out with and further converting them to binary encoding the number of attributes grew exponentially and the processing of these attributes during the testing phase was very time consuming and yet the results weren't much promising .

Table 2: Model Evaluation Metrics

Model	**Precision**	**Recall**	**F1 Score**	**Mean Accuracy**
K-NN	0.58	0.50	0.54	0.54
Random Forest	**0.71**	0.50	0.59	0.67
Logistic Regression	0.58	0.47	0.52	0.62
Decision Tree	0.63	0.66	**0.64**	0.59
XGBClassifier	**0.71**	0.50	0.59	0.64
GaussianNB	0.56	0.43	0.49	0.61
BinaryClassifier(Standard)	0.65	0.46	0.54	0.33
BinaryClassifier(Standard-Support)	0.66	0.47	0.55	0.33
BinaryClassifier(Ratio Support)	0.46	**1**	0.63	0.33
PatternClassifier(with binarized data)	0.65	0.61	0.63	**0.84**
PatternClassifier	0.65	0.61	0.63	**0.84**

From first test I got 33% with 54% F1 . From table we can see that PatternClassifier give super good results and all other proposed algorithms are not so good for this data set. Other methods have comparable results on f1 score and precision score, but in the case of accuracy, a high difference was achieved only by one proposed algorithm. As that of recall we can see the results repeating itself as compared to previous results of telecom churn data-set. The recall of Ratio support method for Binary Classifier gives to

be the highest but yet perform very poor for mean accuracy. There could be a possibility of lesser training data as to save the processing time,Only 1000 rows were used to work for this mode. As of the case of Pattern classifier we try to use it with the binarized and non binzaried data-set (which is just a small experiment). We saw not much diffrence in the results in using the binzaried data , and for on non binzaried dataset on our pattern classifier.

5 Dry bean data

5.1 About Data-set

Abstract

Images of 13,611 grains of 7 different registered dry beans were taken with a high-resolution camera. A total of 16 features; 12 dimensions and 4 shape forms, were obtained from the grains.

Relevant Information

Seven different types of dry beans were used in this research, taking into account the features such as form, shape, type, and structure by the market situation. A computer vision system was developed to distinguish seven different registered varieties of dry beans with similar features in order to obtain uniform seed classification. For the classification model, images of 13,611 grains of 7 different registered dry beans were taken with a high-resolution camera. Bean images obtained by computer vision system were subjected to segmentation and feature extraction stages, and a total of 16 features; 12 dimensions and 4 shape forms, were obtained from the grains.

Attribute Information

1. Area (A): The area of a bean zone and the number of pixels within its boundaries.

2. Perimeter (P): Bean circumference is defined as the length of its border.

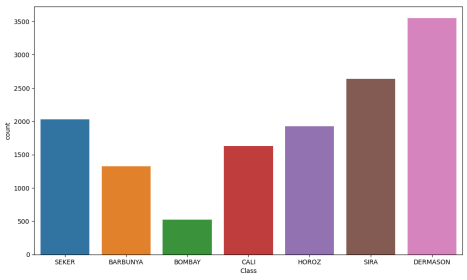

Figure 7: Caption for the first image

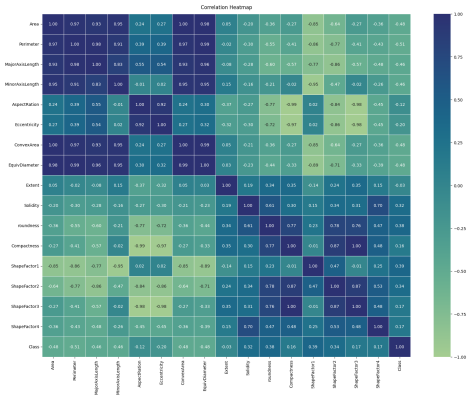

Figure 8: Caption for the second image

Figure 9: Overall caption for both images

3. Major axis length (L): The distance between the ends of the longest line that can be drawn from a bean.

4. Minor axis length (l): The longest line that can be drawn from the bean while standing perpendicular to the main axis.

5. Aspect ratio (K): Defines the relationship between L and l.

6. Eccentricity (Ec): Eccentricity of the ellipse having the same moments as the region.

7. Convex area (C): Number of pixels in the smallest convex polygon that can contain the area of a bean seed.

8. Equivalent diameter (Ed): The diameter of a circle having the same area as a bean seed area.

9. Extent (Ex): The ratio of the pixels in the bounding box to the bean area.

10. Solidity (S): Also known as convexity. The ratio of the pixels in the convex shell to those found in beans.

11. Roundness (R): Calculated with the following formula: $\frac{4\pi A}{P^2}$

12. Compactness (CO): Measures the roundness of an object: $\frac{Ed}{L}$
13. ShapeFactor1 (SF1)
14. ShapeFactor2 (SF2)
15. ShapeFactor3 (SF3)
16. ShapeFactor4 (SF4)
17. Class (Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sira)

5.2 Results and pre-processing

You can see results on table 3. From this we can see that proposed algorithm peaks on the accuracy with 100% for all the proposed classifier(except for Pattern Classifier , which is still 99%). We should note that choosing numerical processing strategy hardly effect model scores and there are a huge field for testing different methods.Highest amount of time was consumed for this data set around 21 minutes for cross validation for nearly all models.

Table 3: Model Evaluation Metrics

Model	**Precision**	**Recall**	**F1 Score**	**Mean Accuracy**
K-NN	0.79	0.89	0.84	0.18
Random Forest	0.89	0.94	0.92	0.77
Logistic Regression	0.78	0.90	0.84	0.67
Decision Tree	0.88	0.89	0.90	0.69
XGBClassifier	0.89	0.94	0.92	0.75
GaussianNB	0.86	0.85	0.85	0.72
BinaryClassifier(Standard)	**1**	**1**	**1**	**1**
BinaryClassifier(Standard-Support)	**1**	**1**	**1**	**1**
BinaryClassifier(Ratio Support)	0.53	**1**	0.69	**1**
PatternClassifier(with binarized data)	**1**	0.99	0.99	0.99
PatternClassifier	**1**	0.82	0.90	0.97

More interesting results we can get from comparing best models from previous step with popular models. We have found some of the wonder full results during this comparison which doubts to be unreal, but are true. Also

for this data-set we mainly got very profound results for all F1 , recall and precision. The given results may change as only 1000 out of 13000 tuples were used to work on this model. Over reworking through the algorithm with increased data-set gave the same results. The new iteration was done using doubled amount around 2000 tuples for the 2nd case, there was a minor change in the precision of the Binary Classifier (changed from 0.54 to 0.53 which is negligible) , this also effected the F1 score but the accuracy remain same even on changing the amount of data. As of the case of Pattern classifier we try to use it with the binarized and non binzaried data-set (which is just a small experiment). We saw very improved results in using the binzaried data , and for saying on non binzaried dataset our model still was able to outperform the classical models with marginal differences.

6 Conclusion

In conclusion I can mention that proposed algorithm has greater results that base one and has comparable results with popular classification models.Our models worked very nicely for the telecom and dry bean data-set , but lag in with the in vehicle coupon data, but as that of pattern classifier it gave very nice results with major differences.Doing a little expriment over using the binarized data over the pattern classifier and feeding it as a categorical attrubute do gave imporved results. However there is some problems with performance and over fitting, so there are a huge field for research and testing different approaches. The optimization of the models are very much required so that they can be easily implemented over larger data-sets with lesser time. With the increasing scope of big data and in in-assembly models we expect to have a model that tend to process faster. Comparison on the accuracy and other metrics the model have given very profound results and have peaked in testing. With the better and optimized version of the Lazy FCA model we would be able to have higher quality of results with lesser time consumption.