



ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO LAB 03:

CLASSIFICATION & CLUSTERING

Sinh viên thực hiện:

Trần Thanh Tùng - 18120258.

Trần Hữu Chí Bảo - 18120288.

Môn: Khai thác dữ liệu và ứng dụng

Thành phố Hồ Chí Minh - 2020

MỤC LỤC

MỤC LỤC	2
PHẦN I. TIỀN XỬ LÝ.....	3
1.1 Cột CaptureTime:	3
1.2 Cột ReleaseTime:	3
1.3 Cột BandNumber:.....	4
1.4 Cột Sex:.....	5
1.5 Các cột còn lại:.....	6
PHẦN II. PHÂN LỚP DỮ LIỆU BẰNG WEKA EXPLORER.....	7
1.1 Thực nghiệm A:	7
1.2 Thực nghiệm B:.....	7
1.3 Thực nghiệm C:.....	8
PHẦN III. PHÂN LỚP DỮ LIỆU BẰNG WEKA EXPERIMENT.....	10
PHẦN IV. ĐÁNH GIÁ	12
PHẦN V. TÀI LIỆU THAM KHẢO	14

PHẦN I. TIỀN XỬ LÝ

Nguyên nhân: Do dữ liệu thô ban đầu có nhiều thuộc tính có các giá trị missing.

1.1 Cột CaptureTime:

- Trong cột này, có 1 dòng dữ liệu bị bỏ trống.
- Ta đơn giản là tìm giá trị mean và điền vào giá trị bỏ trống đó.
- Đồng thời, ta đổi thời gian dạng Giờ: Phút sang Phút (phút thứ bao nhiêu trong ngày) do Weka không hiểu thuộc tính này là liên tục. Bước này để ta có thể rời rạc hóa dữ liệu của các yêu cầu B, C.

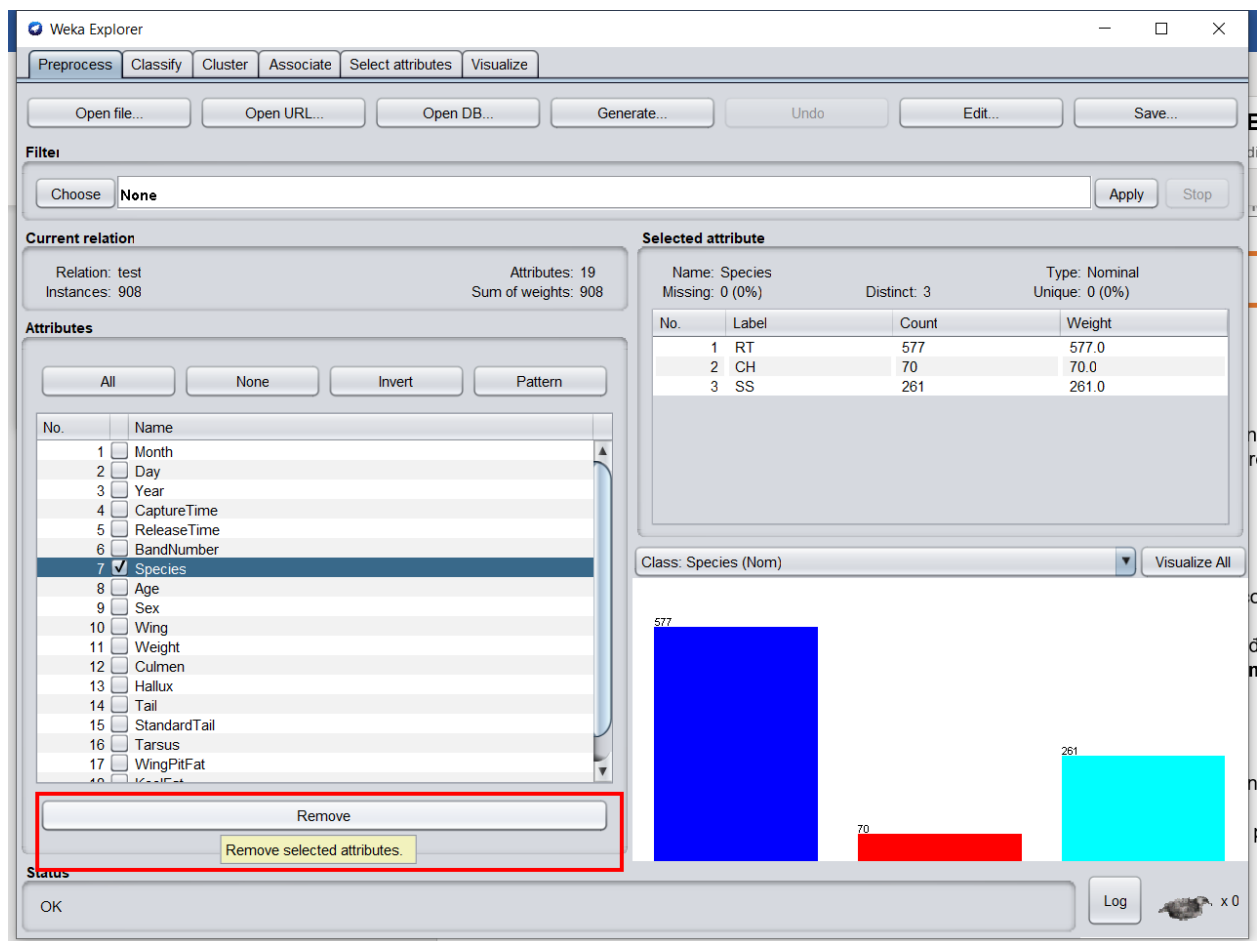
Ta dùng thư viện pandas và numpy để xử lý việc này

```
import numpy as np
import pandas as pd
raw_df = pd.read_csv('hawks.csv') #doc file hawks.csv
#rút trích số phút cột CaptureTime
minute = pd.to_numeric(raw_df['CaptureTime'].str.extract(r'[:](\d+)', expand = False), errors = 'coerce')
#rút trích số giờ cột CaptureTime
hour = pd.to_numeric(raw_df['CaptureTime'].str.extract(r'(\d+):', expand = False), errors = 'coerce')
#chuyển cột CaptureTime về phút
raw_df['CaptureTime'] = (minute + 60*hour).fillna((minute + 60*hour).mean()).astype(int)
```

Hình 1. Xử lý Cột CaptureTime bằng pandas

1.2 Cột ReleaseTime:

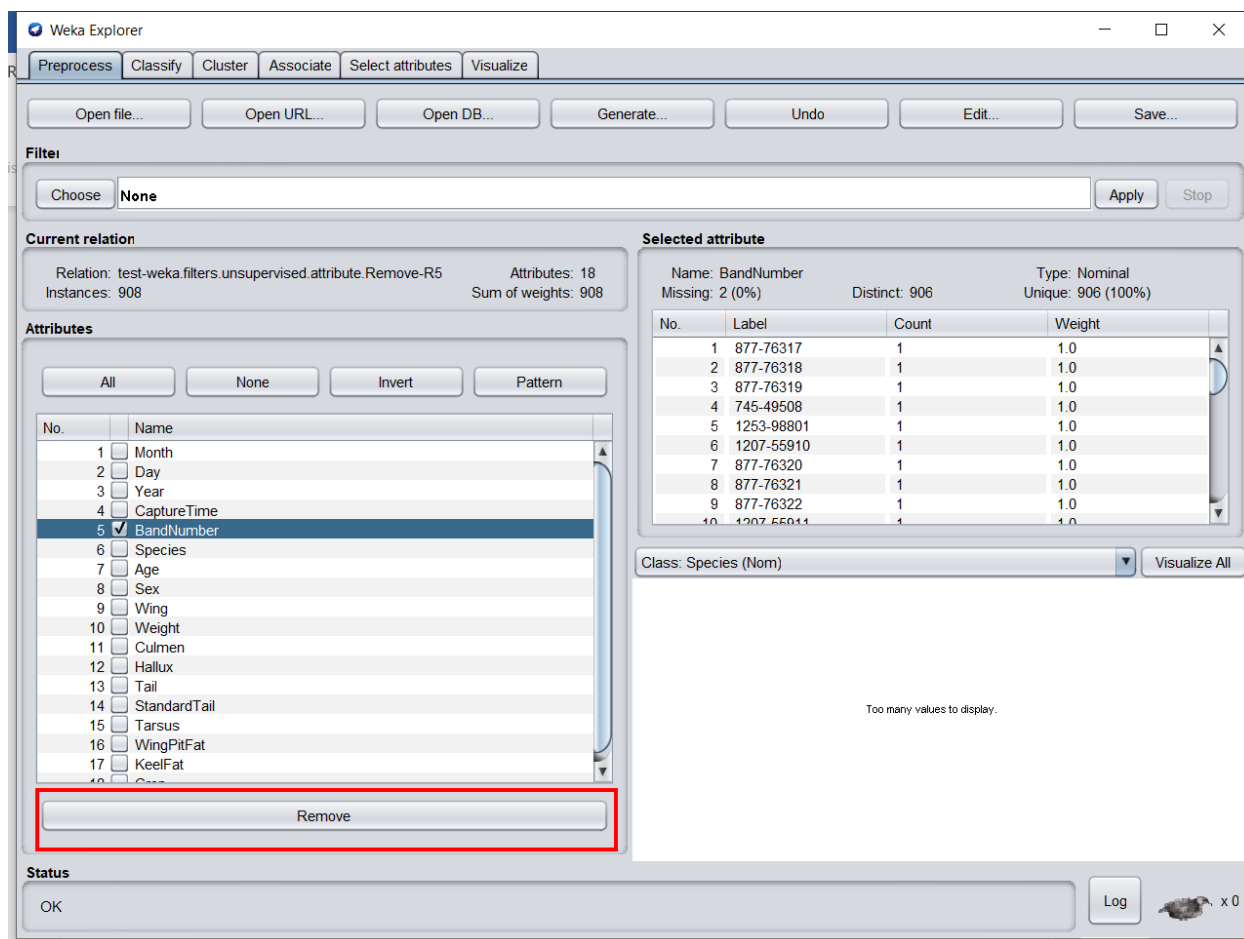
- Thông qua mô tả của tập dữ liệu, đây là cột ghi thời gian thả ra của những con chim ưng sau khi bắt được.
- Thuộc tính này có đến 93% missing, hơn nữa ta nhận thấy rằng sau khi bắt được người ta thường thả nó sau 10-30' → Có thể suy ra từ thuộc tính **CaptureTime** nên ta có thể không xét đến thuộc tính này khi phân tích.
- Ta tiến hành Remove thuộc tính ReleaseTime bằng Weka:



Hình 2. Remove thuộc tính ReleaseTime bằng Weka

1.3 Cột BandNumber:

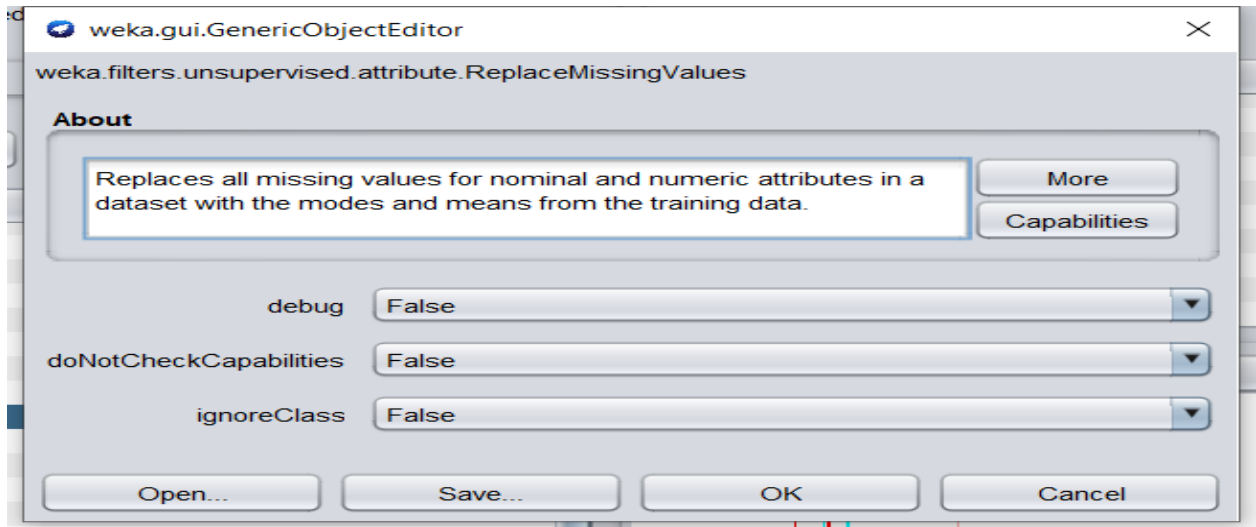
- Thông qua mô tả của tập dữ liệu, thì cột này chứa mã định danh của các con chim ưng được quan sát. Mỗi con mang một mã duy nhất.
- Do đó ta nhận thấy rằng có thể không xét đến thuộc tính này trong quá trình phân lớp.
- Ta tiến hành Remove thuộc tính BandNumber bằng Weka:



Hình 3. Remove thuộc tính BandNumber bằng Weka

1.4 Cột Sex:

- Trong cột này, có đến 63% dữ liệu bị bỏ trống.
- Ta sử dụng filter ReplaceMissingData của Weka để điền các giá trị trống cho thuộc tính:



Hình 4. Bảng thông số của thuộc tính *Replaces all missing values*

1.5 Các cột còn lại:

- Các cột **Wing, Weight, Culmen, Hallux, Tail, StandardTail, Tarsus, WingPitFat, KeelFat, Crop**
- Ta dùng chiến lược điền mean vào các dòng bị bỏ trống của các cột này nguyên nhân do các cột này là các thuộc tính liên tục.

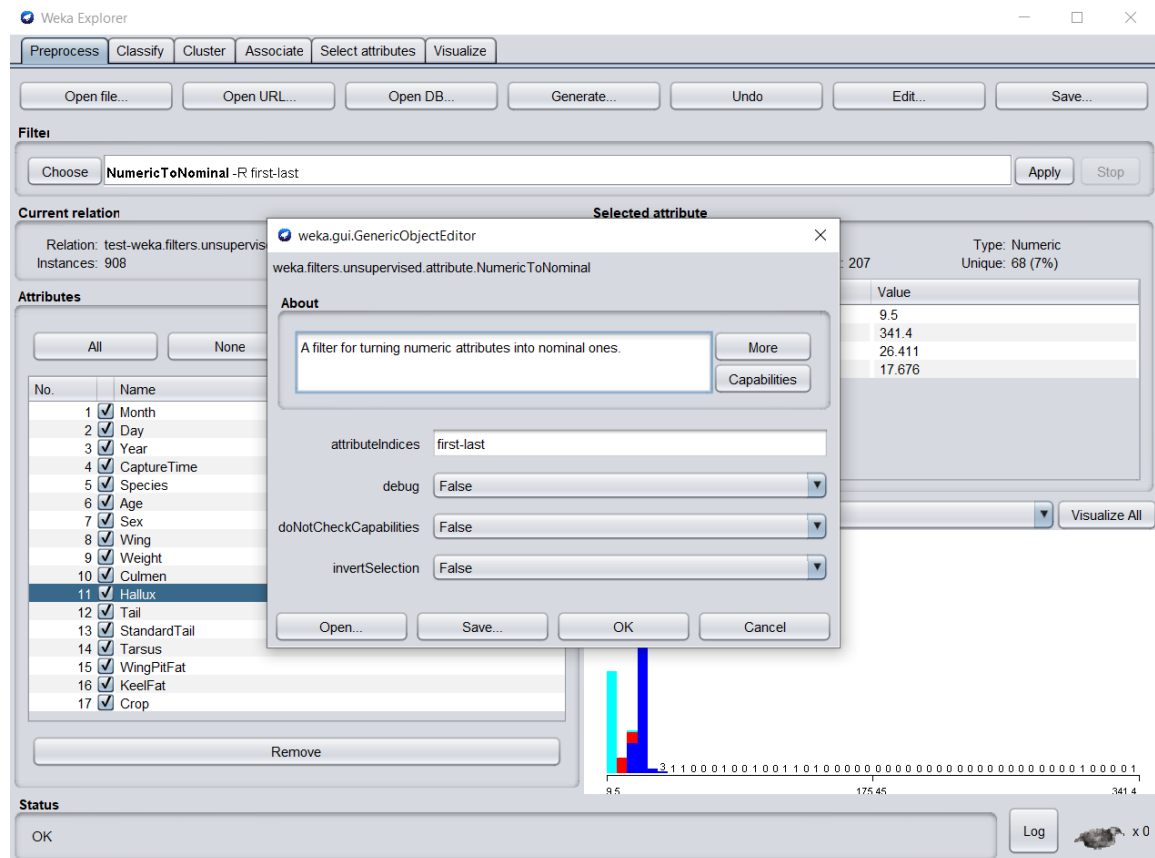
```
def fill_na_col(col):
    if col.name in ['Wing', 'Weight', 'Culmen', 'Hallux', 'Tail', 'StandardTail', 'Tarsus', 'WingPitFat', 'KeelFat', 'Crop']:
        return col.fillna(round(col.mean(),2))
    return col
temp_df = raw_df.apply(fill_na_col)
raw_df = temp_df
raw_df.to_csv('test123.csv', index = False) #lưu kết quả xuống file
```

Hình 5. Điền các dòng bị bỏ trống bằng giá trị mean

PHẦN II. PHÂN LỚP DỮ LIỆU BẰNG WEKA EXPLORER

1.1 Thực nghiệm A:

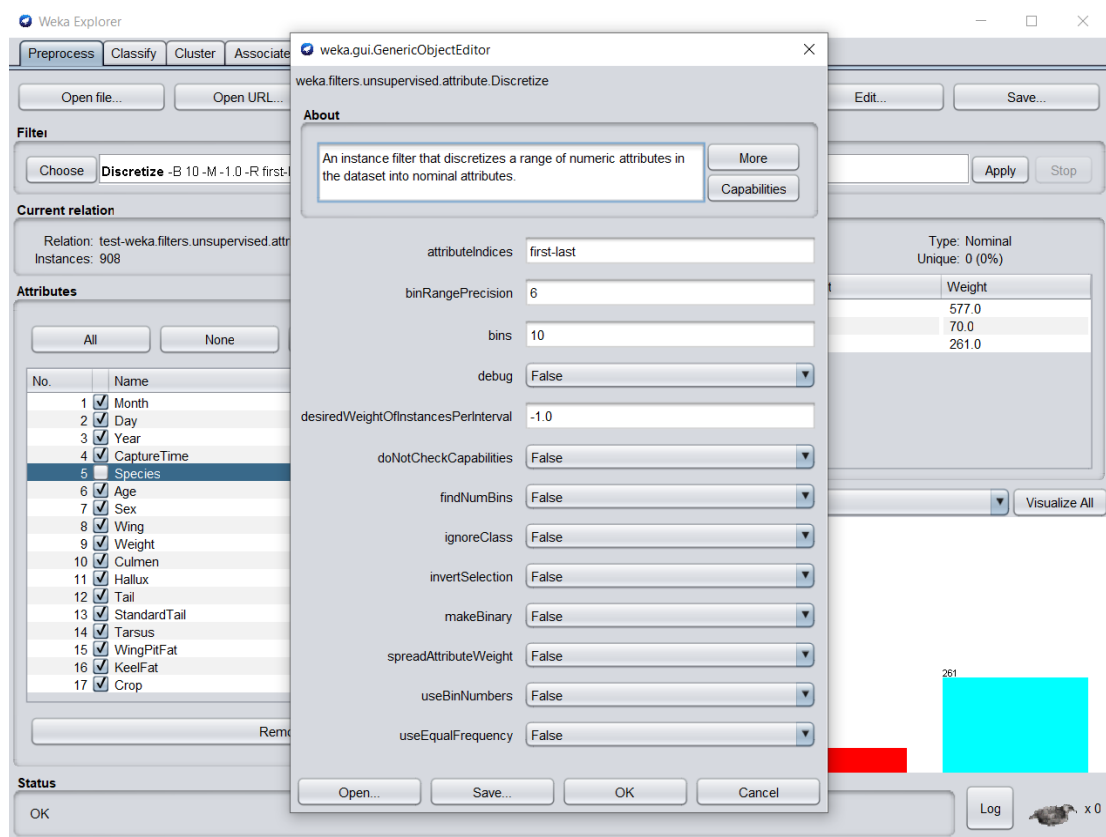
- Chuyển tất cả thuộc tính của dữ liệu đầu vào về nominal do yêu cầu của thuật toán.



Hình 6. Chuyển các thuộc tính thành nominal

1.2 Thực nghiệm B:

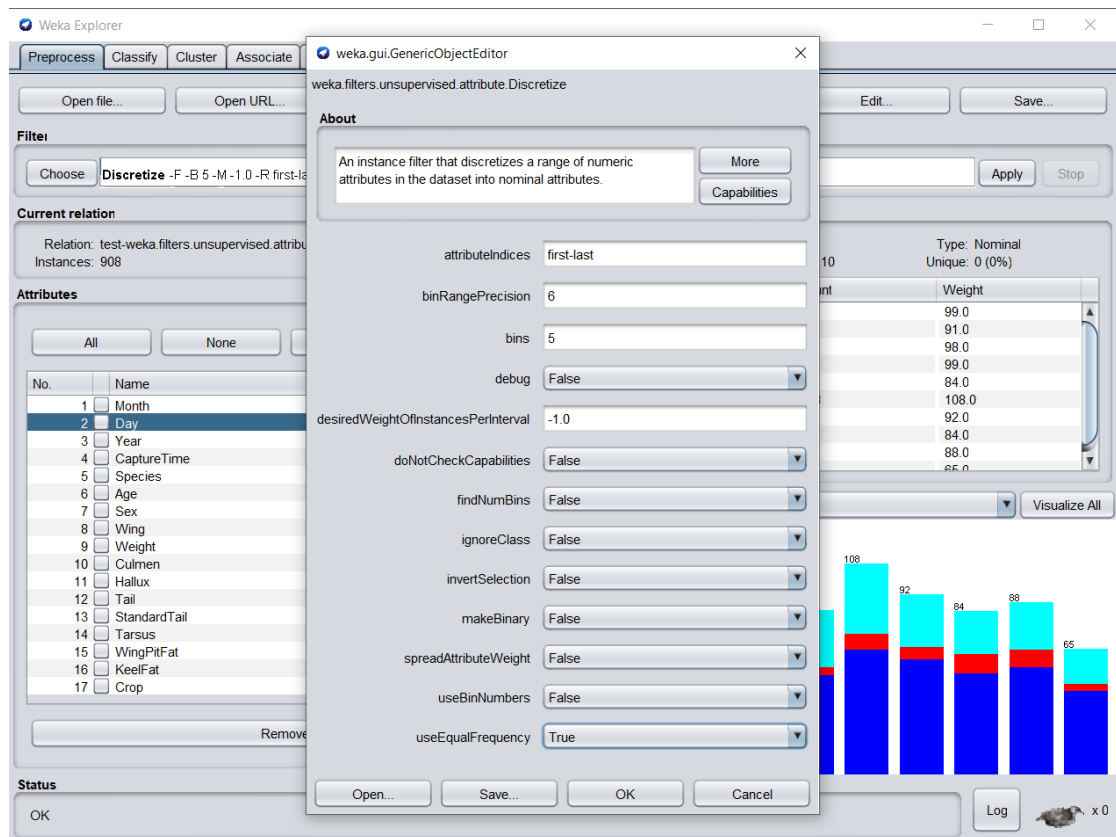
- Rời rạc hóa mọi thuộc tính không phải là lớp trong tập dữ liệu thành 10 giỏ có độ rộng bằng nhau.
- Sử dụng chức năng “Filter” trong cửa sổ “Preprocess” của Explorer, chọn ‘filters’ → ‘unsupervised’ → ‘attribute’ → ‘Discretize’



Hình 7. Bảng thông số.

1.3 Thực nghiệm C:

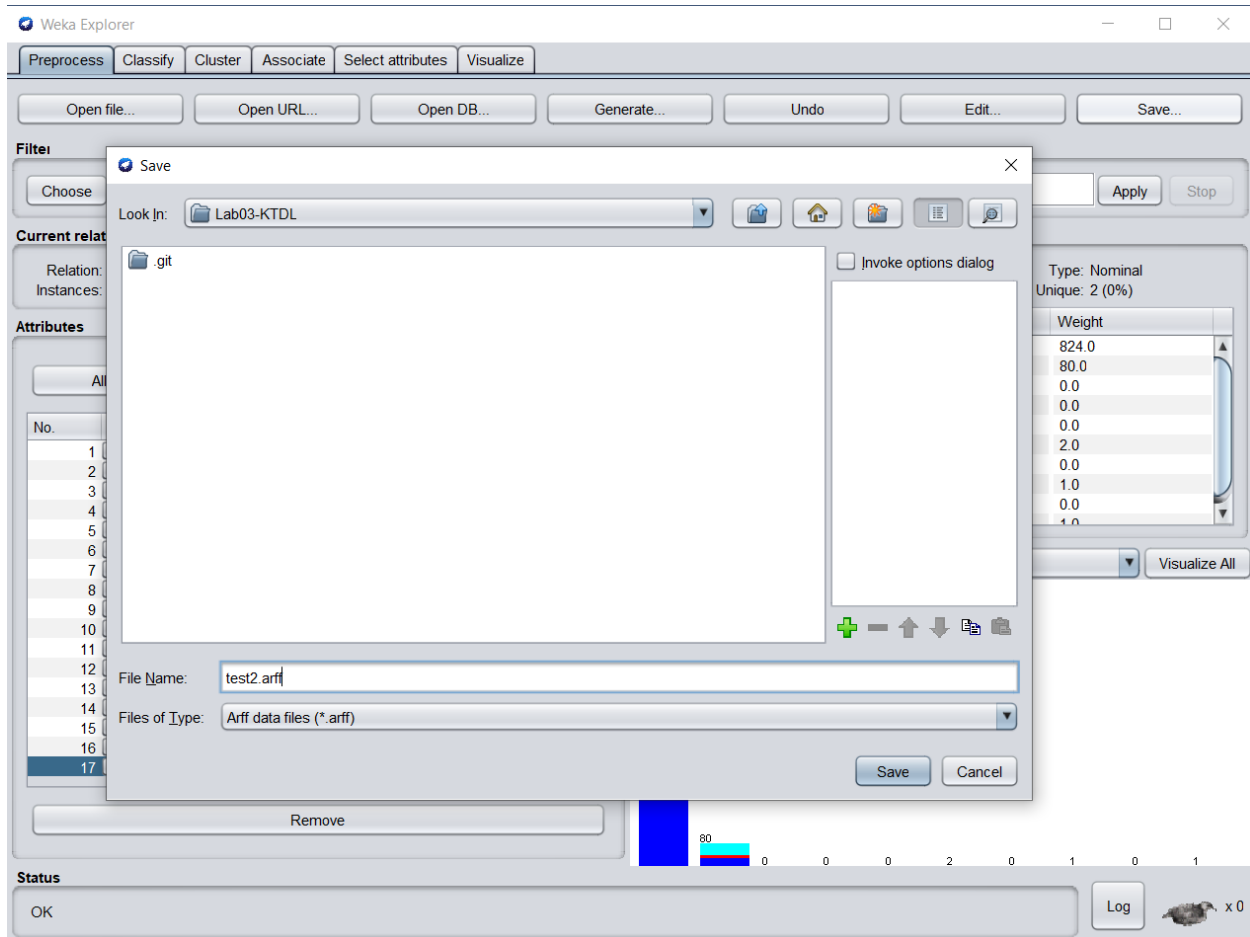
- Rời rạc hóa mọi thuộc tính không phải là lớp trong tập dữ liệu thành 5 giỏ có độ sâu bằng nhau:
- Sử dụng chức năng “Filter” trong cửa sổ “Preprocess” của Explorer, chọn ‘filters’ → ‘unsupervised’ → ‘attribute’ → ‘Discretize’
- **Lưu ý:** Số bins chính = 5 và useEqualFrequency = true.



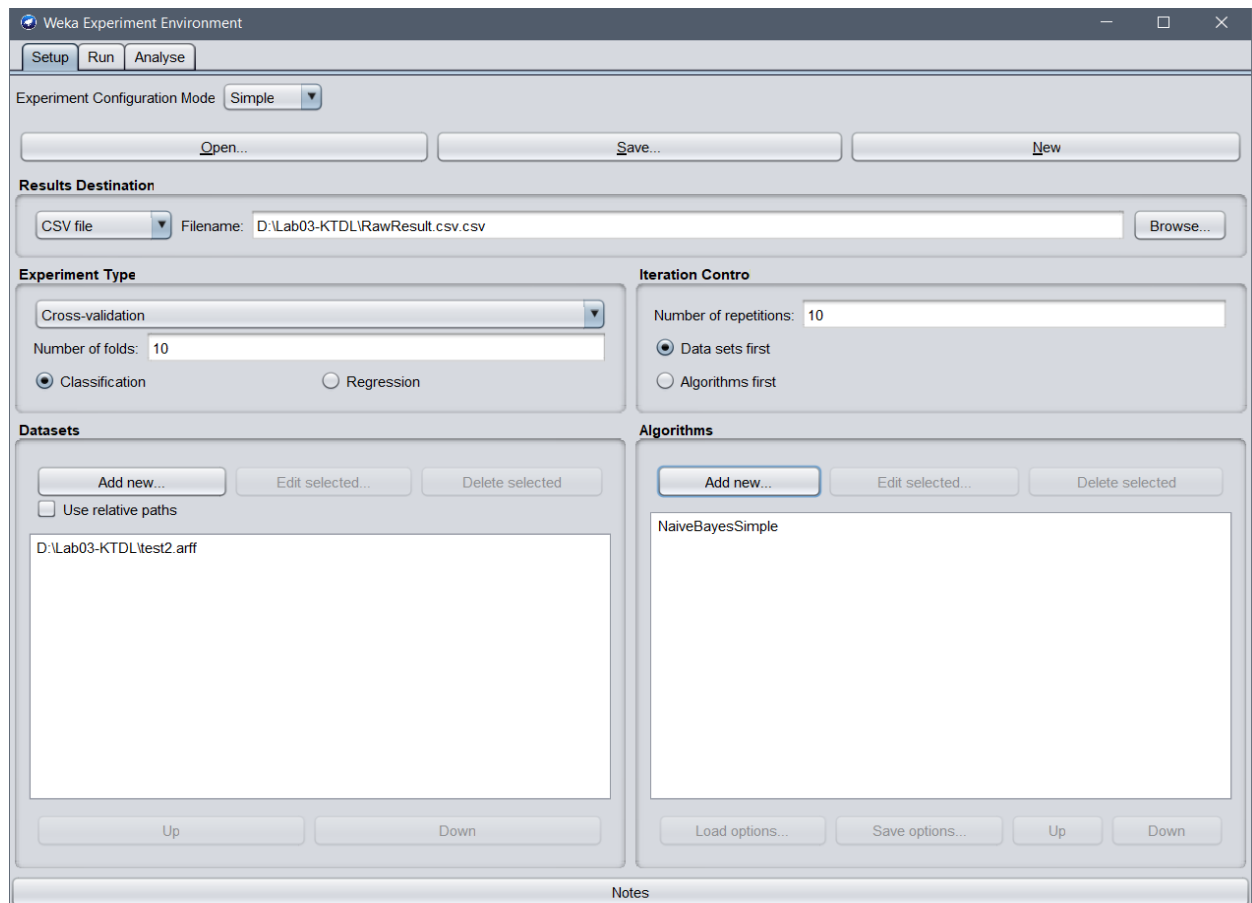
Hình 8. Bảng thông số.

PHẦN III. PHÂN LỚP DỮ LIỆU BẰNG WEKA EXPERIMENT

- Ở thực nghiệm này ta cần sử dụng file arff với các thuộc tính đều là nominal.
- Ta tạo ra file này bằng cách lưu lại file dữ liệu sau khi đã tiền xử lý ở thực nghiệm A dưới dạng file arff.



Hình 9. Lưu lại file đã tiền xử lý ở thực nghiệm A dưới dạng file arff



Hình 10. Giao diện trên weka Experiment

Sau khi chạy thực nghiệm xong, ta tính tỉ lệ trung bình của các mẫu được phân lớp đúng sau 10×10 lượt chạy bằng cách tính mean của cột Percent_correct trong tập tin ResultRaw.csv

M1																									
	Key	Datas	Key_Run	Key_Fold	Key_Schen	Key_Schen	Key_Schen	Date_time	Number_c	Number_c	Number_c	Number_c	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	Key	Datas	Key_Run	Key_Fold	Key_Schen	Key_Schen	Key_Schen	Date_time <td>Number_c<td>Number_c<td>Number_c<td>Number_c<td>K<td>L<td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td></td></td></td></td></td></td>	Number_c <td>Number_c<td>Number_c<td>Number_c<td>K<td>L<td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td></td></td></td></td></td>	Number_c <td>Number_c<td>Number_c<td>K<td>L<td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td></td></td></td></td>	Number_c <td>Number_c<td>K<td>L<td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td></td></td></td>	Number_c <td>K<td>L<td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td></td></td>	K <td>L<td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td></td>	L <td>M<td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td></td>	M <td>N<td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td></td>	N <td>O<td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td></td>	O <td>P<td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td></td>	P <td>Q<td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td></td>	Q <td>R<td>S<td>T<td>U<td>V<td>W</td></td></td></td></td></td>	R <td>S<td>T<td>U<td>V<td>W</td></td></td></td></td>	S <td>T<td>U<td>V<td>W</td></td></td></td>	T <td>U<td>V<td>W</td></td></td>	U <td>V<td>W</td></td>	V <td>W</td>	W
2	test2-weki	1	1	weka.class	1	weka.class	2.02E+07	817	91	66	25	0	0	72.52747253	72.47253	0	0.580954	0.052522	0.19526	46.18614	82.02239	177.8518	155.2056	22.64	
3	test2-weki	1	2	weka.class	1	weka.class	2.02E+07	817	91	65	26	0	0	71.42857143	71.42857143	0	0.576593	0.05544	0.192418	48.75189	80.82856	177.8518	146.5624	31.28	
4	test2-weki	1	3	weka.class	1	weka.class	2.02E+07	817	91	63	28	0	0	69.23076923	70.76923	0	0.542138	0.055662	0.199378	49.15423	84.15286	175.607	147.9329	27.67	
5	test2-weki	1	4	weka.class	1	weka.class	2.02E+07	817	91	63	28	0	0	69.23076923	70.76923	0	0.541891	0.05412	0.19605	47.79282	82.74846	175.607	136.651	38	
6	test2-weki	1	5	weka.class	1	weka.class	2.02E+07	817	91	67	24	0	0	73.62637363	67.37363	0	0.603486	0.052173	0.185809	46.03472	78.35181	182.8635	164.3294	18.57	
7	test2-weki	1	6	weka.class	1	weka.class	2.02E+07	817	91	70	21	0	0	76.92307692	70.07692	0	0.645125	0.050336	0.184032	44.17971	77.14088	189.3915	155.7927	33.55	
8	test2-weki	1	7	weka.class	1	weka.class	2.02E+07	817	91	61	30	0	0	67.03296703	92.96703	0	0.509081	0.058615	0.203921	51.45173	85.48856	187.3915	205.7378	-18.7	
9	test2-weki	1	8	weka.class	1	weka.class	2.02E+07	817	91	71	20	0	0	78.02197802	91.97802	0	0.66538	0.045115	0.172384	39.60204	72.26758	187.3915	161.2387	26.15	
10	test2-weki	1	9	weka.class	1	weka.class	2.02E+07	818	90	68	22	0	0	75.55555556	44.44444	0	0.617613	0.045635	0.178248	40.26009	75.16972	174.1813	133.6925	40.48	
11	test2-weki	1	10	weka.class	1	weka.class	2.02E+07	818	90	68	22	0	0	75.55555556	44.44444	0	0.636697	0.045931	0.179416	40.53035	75.66252	174.1813	140.5798	33.66	
12	test2-weki	2	1	weka.class	1	weka.class	2.02E+07	817	91	68	23	0	0	74.72527473	52.7473	0	0.628043	0.047687	0.178433	42.11139	75.31244	175.607	118.4216	57.18	
13	test2-weki	2	2	weka.class	1	weka.class	2.02E+07	817	91	69	22	0	0	75.82417582	41.7582	0	0.635801	0.045513	0.178392	40.20367	75.32039	176.4104	125.4302	50.98	
14	test2-weki	2	3	weka.class	1	weka.class	2.02E+07	817	91	63	28	0	0	69.23076923	70.76923	0	0.534527	0.052966	0.191759	46.7872	80.96411	176.4104	133.3582	43.05	
15	test2-weki	2	4	weka.class	1	weka.class	2.02E+07	817	91	66	25	0	0	72.52747253	27.47253	0	0.586664	0.051942	0.189587	45.5955	79.48105	186.3915	173.0931	13.25	
16	test2-weki	2	5	weka.class	1	weka.class	2.02E+07	817	91	62	29	0	0	68.13186813	3.86813	0	0.533828	0.056236	0.196877	49.36226	82.53391	187.9765	193.1161	-5.13	
17	test2-weki	2	6	weka.class	1	weka.class	2.02E+07	817	91	66	25	0	0	72.52747253	27.47253	0	0.57269	0.051849	0.18993	45.54371	79.68382	183.72	172.8948	10.82	
18	test2-weki	2	7	weka.class	1	weka.class	2.02E+07	817	91	61	30	0	0	67.03296703	39.6703	0	0.515786	0.058088	0.201295	51.02389	84.45198	183.72	175.3173	8.402	
19	test2-weki	2	8	weka.class	1	weka.class	2.02E+07	817	91	71	20	0	0	78.02197802	2.97802	0	0.667337	0.047146	0.176993	41.41227	74.25659	183.72	134.7362	48.98	
20	test2-weki	2	9	weka.class	1	weka.class	2.02E+07	818	90	62	28	0	0	68.88888889	3.11111	0	0.526938	0.051467	0.190914	45.41558	80.51105	174.1813	157.1121	17.06	
21	test2-weki	2	10	weka.class	1	weka.class	2.02E+07	818	90	64	26	0	0	71.11111111	28.88889	0	0.560563	0.052534	0.192031	46.35729	80.98246	174.1813	159.233	14.5	
22	test2-weki	3	1	weka.class	1	weka.class	2.02E+07	817	91	67	24	0	0	73.62637363	26.37363	0	0.599193	0.047991	0.180547	42.39243	76.23024	176.4104	122.6944	53.71	
23	test2-weki	3	2	weka.class	1	weka.class	2.02E+07	817	91	68	23	0	0	74.72527473	25.27473	0	0.615469	0.047895	0.183471	42.2668	77.37926	179.2786	147.3801	31.85	
24	test2-weki	3	3	weka.class	1	weka.class	2.02E+07	817	91	63	28	0	0	69.23076923	70.76923	0	0.537904	0.056542	0.198476	49.89554	83.70388	180.8635	184.2908	-3.42	
25	test2-weki	3	4	weka.class	1	weka.class	2.02E+07	817	91	66	25	0	0	72.52747253	72.47253	0	0.593387	0.050529	0.183049	44.38508	76.79887	182.72	136.7424	45.97	
26	test2-weki	3	5	weka.class	1	weka.class	2.02E+07	817	91	65	26	0	0	71.42857143	71.42857143	0	0.565233	0.050806	0.192036	44.62827	80.56927	182.72	175.3711	7.34	
27	test2-weki	3	6	weka.class	1	weka.class	2.02E+07	817	91	66	25	0	0	72.52747253	27.47253	0	0.59404	0.053769	0.190546	47.2302	79.94233	183.72	152.4002	31.31	
28	test2-weki	3	7	weka.class	1	weka.class	2.02E+07	817	91	71	20	0	0	78.02197802	91.97802	0	0.659877	0.046419	0.178342	40.77372	74.8224	183.72	157.6685	26.05	
29	test2-weki	3	8	weka.class	1	weka.class	2.02E+07	817	91	64	27	0	0	70.32967033	99.67033	0	0.5632	0.058313	0.200677	51.2214	84.19302	183.72	173.2595	10.46	

Hình 11. Cột Percent_correct trong file ResultRaw.csv

PHẦN IV. ĐÁNH GIÁ

Từ kết quả thu được qua các thực nghiệm đánh giá, ta rút ra được một số kết luận:

- **Phương pháp phân lớp nào thường cho kết quả cao nhất?**
 - Phương pháp phân lớp NaiveBayesSimple thường cho kết quả cao nhất.
- **Phương pháp nào không thực hiện tốt và tại sao?**
 - 2 phương pháp ID3 và J48 khi sử dụng 2 chiến lược đánh giá “cross validation với 10 fold” và “Percentage split với tỷ lệ 66%” khi thực nghiệm với tập dữ liệu không được chia thành các bin thì cho ra kết quả không tốt, nguyên nhân do 2 thuật toán này là thuật toán phân lớp bằng cách tạo ra cây quyết định. Mà trong tập dữ liệu của ta, có khá nhiều thuộc tính missing (được điền bằng giá trị mean) vì thế cây quyết định tạo ra không mang tính hiệu quả cao.
 - Trong thực nghiệm D bằng weka Experimenter ta nhận thấy cả 2 phương pháp NaiveBayesSimple và J48 đều không mang kết quả khả quan khi mức độ phân lớp đúng của cả 2 chỉ hơn 70%. NaiveBayesSimple còn cho kết quả tệ hơn J48 khi kết quả phân lớp đúng trung bình chỉ là 72%, nguyên nhân vì NaiveBayesSimple dựa trên giả định rằng các thuộc tính độc lập với nhau, điều này là phi thực tế trong tập dữ liệu của ta.
- **Tại sao ta sử dụng phiên bản đã rời rạc hóa của tập dữ liệu nếu tập dữ liệu đã được rời rạc hóa?**
 - Ta sử dụng phiên bản đã rời rạc hóa của tập dữ liệu nếu tập dữ liệu đã được rời rạc hóa vì một số thuộc tính rời rạc mang quá nhiều giá, điều này đôi khi khiến cho các thuật toán phân lớp hoạt động kém hiệu quả. Để giải quyết vấn đề này, ta sử dụng phiên bản đã được rời rạc hóa của chúng.
- **Việc rời rạc hóa và cách rời rạc hóa có ảnh hưởng đến kết quả phân lớp hay không, nếu có thì ảnh hưởng thế nào?**
 - Việc rời rạc hóa và cách rời rạc hóa có ảnh hưởng tới việc phân lớp dữ liệu. Đối với 2 thuật toán ID3 và J48, thì việc phân loại được thực hiện tốt hơn trong trường hợp các thuộc tính được rời rạc hóa. So sánh 2 cách rời rạc hóa dữ liệu là rời rạc theo chiều rộng giỏ và rời rạc theo chiều sâu của giỏ ta cũng thấy có sự khác biệt trong kết quả phân loại.
- **Chiến lược nào trong ba chiến lược đánh giá đã đánh giá quá cao (overestimate) độ chính xác và tại sao?**
 - Trong 3 chiến lược đánh giá, thì “Use training set” đã đánh giá quá cao (overestimate) độ chính xác. Nguyên nhân vì cách hoạt động của Use training set:

1. Weka lấy tất cả dữ liệu đã được gán nhãn
 2. Dùng thuật toán được chọn để xây dựng mô hình phân lớp từ các dữ liệu này.
 3. Sau đó, dùng mô hình này để phân loại lại các dữ liệu ban đầu.
 4. Cho ra kết quả phân loại.
- Chính vì cách hoạt động như vậy mà chiến lược đánh giá này đánh giá quá cao độ chính xác.
- **Chiến lược nào đánh giá thấp (underestimate) độ chính xác và tại sao?**
 - Chiến lược “Percentage Split với tỷ lệ 66% đánh giá thấp độ chính xác (underestimate) vì chiến lược này sẽ chia tập dữ liệu của chúng ta thành 66% training data và 34% testing data, và cách chia là ngẫu nhiên. Nên trong nhiều trường hợp khiến cho độ chính xác thấp hơn so với thực tế.

PHẦN V. TÀI LIỆU THAM KHẢO

- [1] <https://stackoverflow.com/questions/10437677/cross-validation-in-weka>
- [2] https://www.periyaruniversity.ac.in/ijcii/issue/marnew/2_mar_18.pdf
- [3] <https://towardsdatascience.com/6-different-ways-to-compensate-for-missing-values-data-imputation-with-examples-6022d9ca0779>
- [4] https://www.w3schools.com/python/python_regex.asp
- [5] <https://stats.stackexchange.com/questions/419803/cross-validation-or-percentage-split>
- [6] <https://towardsdatascience.com/5-reasons-why-you-should-use-cross-validation-in-your-data-science-project-8163311a1e79#:~:text=The%20classic%20approach%20is%20to,can%20create%20these%20folds%20with.>
- [7] <https://www.cs.waikato.ac.nz/ml/weka/mooc/dataminingwithweka/transcripts/Transcript2-2.txt#:~:text=If%20we%20had%20just%20one,random%20split%20of%20the%20dataset>