

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Машинное обучение»
Тема: Частотный анализ

Студент гр. 6304

Ковынев М.В.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

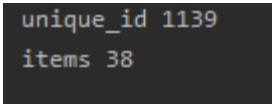
2020

Цель

Ознакомиться с методами частотного анализа из библиотеки MLxtend

Ход работы

1. Загружен датасет по ссылке: <https://www.kaggle.com/acostasg/random-shopping-cart>. Данные представлены в виде csv таблицы. Данные представляют собой информацию о том, какой покупатель что и когда покупал.
2. Создан Python скрипт. Загружены данные в датафрейм
3. Получен список всех id покупателей, которые есть в файле
4. Получен список всех товаров, которые есть в файле



```
unique_id 1139
items 38
```

Рисунок 1 — Количество покупателей и товаров

5. Далее сформирован датасет подходящий для частотного анализа. Для этого были слиты все товары одного покупателя в один список. Для дальнейшего частотного анализа id покупателя будет не нужен.
6. Так как полученный датасет не пригоден для анализа напрямую, так как каждый список пользователя может содержать разное количество товаров. Поэтому данные были закодированы так, чтобы их можно было представить в виде матрицы. Для кодирования данных использовался TransactionEncoder
7. Выведен полученный dataframe. Строки - id покупателей, столбцы – товары, пересечение – наличие у покупателя товара.
8. Применим алгоритм apriori с минимальным уровнем поддержки 0.3. Результат представлен в Приложении А. В таблице товары (или несколько), которые встречаются не реже, чем в 0.3 наборах товаров.
9. Применим алгоритм apriori с тем же уровнем поддержки, но ограничим максимальный размер набора единицей. Результат представлен в Приложении Б. В таблице товары, которые встречаются не реже, чем в 0.3 наборах товаров.

- 10.Применим алгоритм *apriori* и выведем только те наборы, которые имеют размер 2, а также количество таких наборов. Результат представлен в Приложении В. В таблице товары (кол-во 2 штуки), которые встречаются не реже, чем в 0.3 наборах товаров.
- 11.Посчитано количество наборов при различных уровнях поддержки. Начальное значение поддержки 0.05, шаг 0.01. Построен график зависимости количества наборов от уровня поддержки.
- 12.Определены значения уровня поддержки при котором перестают генерироваться наборы размера 1,2,3, и. т.д. Отмечены полученные уровни поддержки на графике, построенном в пункте 11

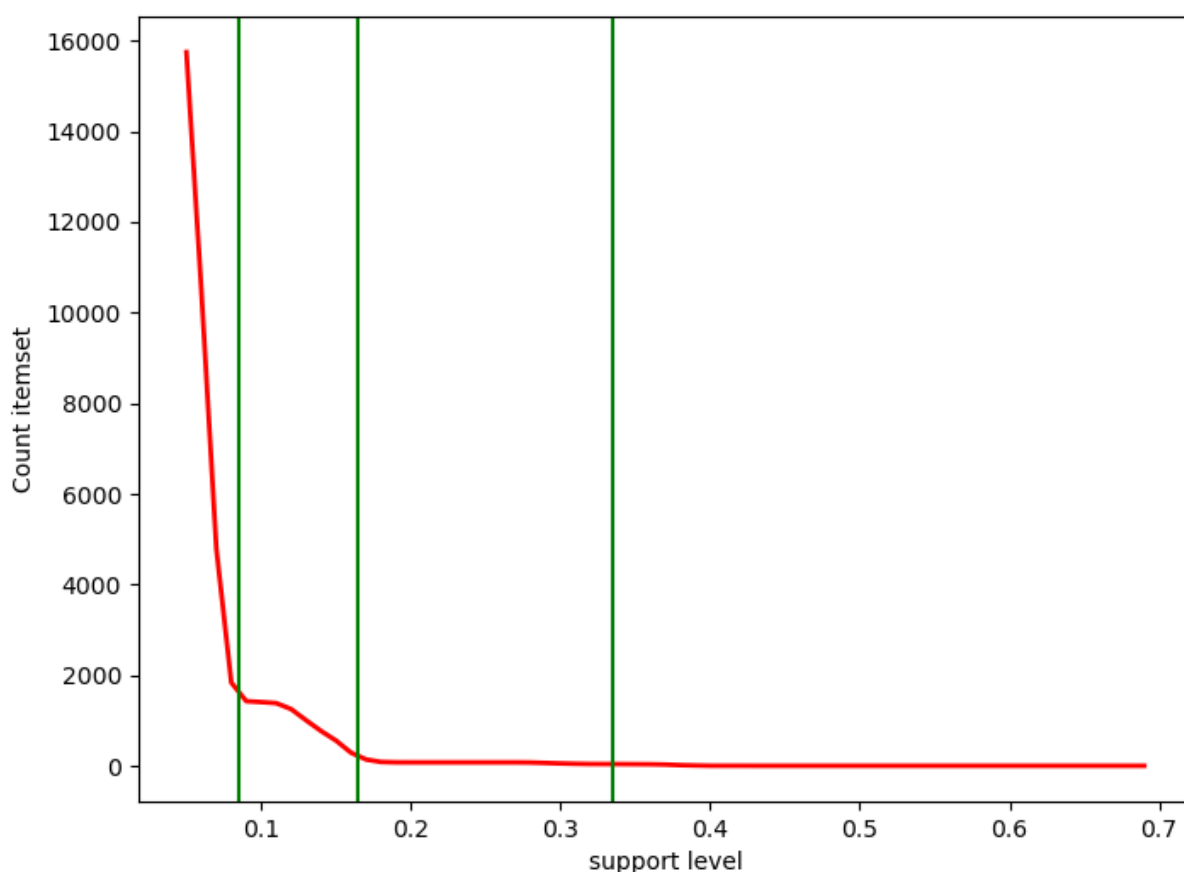


Рисунок 2 — Зависимости количества наборов от уровня поддержки.

Уровни поддержки при котором перестают генерироваться наборы

- 4 - 0.085
- 3 - 0.165
- 2 - 0.335

13. Построим датасет только из тех элементов, которые попадают в наборы размером 1 при уровне поддержки 0.38.
14. Приведен полученный датасет к формату, который можно обработать
15. Проведен ассоциативный анализ при уровне поддержки 0.3 для нового датасета. Результат представлен в Приложении Г. В таблице товары, которые аналогичные в Приложение А, только товары длиной 1 имеет минимальный уровень поддержки 0.38
16. Проведен ассоциативный анализ при уровне поддержки 0.15 для нового датасета. Выведен все наборы размер которых больше 1 и в котором есть 'yogurt' или 'waffles'. Результат представлен в Приложении Д.
17. Построен датасет, из тех элементов, которые не попали в датасет в п. 6 и приведите его к удобному для анализа виду. Результат в приложении Е.
18. Написано правило, для вывода всех наборов, в которых хотя бы два элемента начинаются на 's'. Код представлен в Приложении Ж, данные в Приложении И.
19. Написано правило, для вывода всех наборов, для которых уровень поддержки изменяется от 0.1 до 0.25. представлен в Приложении К, данные в Приложении Л.

Вывод

В ходе выполнения данной лабораторной работы было выполнено ознакомление с методами частотного анализа из библиотеки MLxtend. Установлено, что увеличение уровня поддержки ведет к уменьшению количества наборов большей длины.

Приложение А

Apriori, поддержка 0.3

	support	itemsets	length
0	0.374890	(all- purpose)	1
1	0.384548	(aluminum foil)	1
2	0.385426	(bagels)	1
3	0.374890	(beef)	1
4	0.367867	(butter)	1
5	0.395961	(cereals)	1
6	0.390694	(cheeses)	1
7	0.379280	(coffee/tea)	1
8	0.388938	(dinner rolls)	1
9	0.388060	(dishwashing liquid/detergent)	1
10	0.389816	(eggs)	1
11	0.352941	(flour)	1
12	0.370500	(fruits)	1
13	0.345917	(hand soap)	1
14	0.398595	(ice cream)	1
15	0.375768	(individual meals)	1
16	0.376646	(juice)	1
17	0.371378	(ketchup)	1
18	0.378402	(laundry detergent)	1
19	0.395083	(lunch meat)	1
20	0.380158	(milk)	1
21	0.375768	(mixes)	1
22	0.362599	(paper towels)	1
23	0.371378	(pasta)	1
24	0.355575	(pork)	1
25	0.421422	(poultry)	1
26	0.367867	(sandwich bags)	1
27	0.349429	(sandwich loaves)	1
28	0.368745	(shampoo)	1
29	0.379280	(soap)	1
30	0.390694	(soda)	1
31	0.373134	(spaghetti sauce)	1
32	0.360843	(sugar)	1
33	0.378402	(toilet paper)	1
34	0.369622	(tortillas)	1
35	0.739245	(vegetables)	1
36	0.394205	(waffles)	1
37	0.384548	(yogurt)	1
38	0.310799	(aluminum foil, vegetables)	2
39	0.300263	(vegetables, bagels)	2
40	0.310799	(vegetables, cereals)	2
41	0.309043	(cheeses, vegetables)	2
42	0.308165	(vegetables, dinner rolls)	2
43	0.306409	(dishwashing liquid/detergent, vegetables)	2
44	0.326602	(vegetables, eggs)	2
45	0.302897	(ice cream, vegetables)	2
46	0.309043	(vegetables, laundry detergent)	2
47	0.311677	(vegetables, lunch meat)	2
48	0.331870	(vegetables, poultry)	2
49	0.305531	(soda, vegetables)	2
50	0.315189	(waffles, vegetables)	2
51	0.319579	(yogurt, vegetables)	2

Приложение Б
Apriori, поддержка 0.3, максимум 1

0	0.374890	(all- purpose)
1	0.384548	(aluminum foil)
2	0.385426	(bagels)
3	0.374890	(beef)
4	0.367867	(butter)
5	0.395961	(cereals)
6	0.390694	(cheeses)
7	0.379280	(coffee/tea)
8	0.388938	(dinner rolls)
9	0.388060	(dishwashing liquid/detergent)
10	0.389816	(eggs)
11	0.352941	(flour)
12	0.370500	(fruits)
13	0.345917	(hand soap)
14	0.398595	(ice cream)
15	0.375768	(individual meals)
16	0.376646	(juice)
17	0.371378	(ketchup)
18	0.378402	(laundry detergent)
19	0.395083	(lunch meat)
20	0.380158	(milk)
21	0.375768	(mixes)
22	0.362599	(paper towels)
23	0.371378	(pasta)
24	0.355575	(pork)
25	0.421422	(poultry)
26	0.367867	(sandwich bags)
27	0.349429	(sandwich loaves)
28	0.368745	(shampoo)
29	0.379280	(soap)
30	0.390694	(soda)
31	0.373134	(spaghetti sauce)
32	0.360843	(sugar)
33	0.378402	(toilet paper)
34	0.369622	(tortillas)
35	0.739245	(vegetables)
36	0.394205	(waffles)
37	0.384548	(yogurt)

Приложение В

Apriori, поддержка 0.3, длина 2

	support	itemsets	length
38	0.310799	(vegetables, aluminum foil)	2
39	0.300263	(bagels, vegetables)	2
40	0.310799	(cereals, vegetables)	2
41	0.309043	(cheeses, vegetables)	2
42	0.308165	(dinner rolls, vegetables)	2
43	0.306409	(vegetables, dishwashing liquid/detergent)	2
44	0.326602	(eggs, vegetables)	2
45	0.302897	(ice cream, vegetables)	2
46	0.309043	(laundry detergent, vegetables)	2
47	0.311677	(lunch meat, vegetables)	2
48	0.331870	(poultry, vegetables)	2
49	0.305531	(soda, vegetables)	2
50	0.315189	(waffles, vegetables)	2
51	0.319579	(yogurt, vegetables)	2

Count of result itemstes = 14

Приложение Г

Apriori, поддержка 0.38, new_dataset

	support	itemsets	length
0	0.384548	(aluminum foil)	1
1	0.385426	(bagels)	1
2	0.395961	(cereals)	1
3	0.390694	(cheeses)	1
4	0.388938	(dinner rolls)	1
5	0.388060	(dishwashing liquid/detergent)	1
6	0.389816	(eggs)	1
7	0.398595	(ice cream)	1
8	0.395083	(lunch meat)	1
9	0.380158	(milk)	1
10	0.421422	(poultry)	1
11	0.390694	(soda)	1
12	0.739245	(vegetables)	1
13	0.394205	(waffles)	1
14	0.384548	(yogurt)	1
15	0.310799	(vegetables, aluminum foil)	2
16	0.300263	(bagels, vegetables)	2
17	0.310799	(vegetables, cereals)	2
18	0.309043	(vegetables, cheeses)	2
19	0.308165	(vegetables, dinner rolls)	2
20	0.306409	(vegetables, dishwashing liquid/detergent)	2
21	0.326602	(vegetables, eggs)	2
22	0.302897	(vegetables, ice cream)	2
23	0.311677	(lunch meat, vegetables)	2
24	0.331870	(vegetables, poultry)	2
25	0.305531	(soda, vegetables)	2
26	0.315189	(vegetables, waffles)	2
27	0.319579	(vegetables, yogurt)	2

Приложение Д

Apriori, поддержка 0.15, yougurt или waffles

	support	itemsets	length
0	0.169447	(waffles, aluminum foil)	2
1	0.159789	(bagels, waffles)	2
2	0.160667	(waffles, cereals)	2
3	0.172959	(cheeses, waffles)	2
4	0.169447	(waffles, dinner rolls)	2
5	0.175593	(dishwashing liquid/detergent, waffles)	2
6	0.169447	(eggs, waffles)	2
7	0.172959	(ice cream, waffles)	2
8	0.184372	(waffles, lunch meat)	2
9	0.166813	(waffles, poultry)	2
10	0.177349	(waffles, soda)	2
11	0.315189	(vegetables, waffles)	2
12	0.173837	(yogurt, waffles)	2
13	0.157155	(waffles, vegetables, lunch meat)	3

Приложение Е

Apriori, поддержка 0.3, новый датасет

	support	itemsets
0	0.374890	(all- purpose)
1	0.384548	(aluminum foil)
2	0.385426	(bagels)
3	0.374890	(beef)
4	0.367867	(butter)
5	0.395961	(cereals)
6	0.390694	(cheeses)
7	0.379280	(coffee/tea)
8	0.388938	(dinner rolls)
9	0.388060	(dishwashing liquid/detergent)
10	0.389816	(eggs)
11	0.352941	(flour)
12	0.370500	(fruits)
13	0.345917	(hand soap)
14	0.398595	(ice cream)
15	0.375768	(individual meals)
16	0.376646	(juice)
17	0.371378	(ketchup)
18	0.378402	(laundry detergent)
19	0.395083	(lunch meat)
20	0.380158	(milk)
21	0.375768	(mixes)
22	0.362599	(paper towels)
23	0.371378	(pasta)
24	0.355575	(pork)
25	0.421422	(poultry)
26	0.367867	(sandwich bags)
27	0.349429	(sandwich loaves)
28	0.368745	(shampoo)
29	0.379280	(soap)
30	0.390694	(soda)
31	0.373134	(spaghetti sauce)
32	0.360843	(sugar)
33	0.378402	(toilet paper)
34	0.369622	(tortillas)
35	0.739245	(vegetables)
36	0.394205	(waffles)
37	0.384548	(yogurt)
38	0.310799	(aluminum foil, vegetables)
39	0.300263	(vegetables, bagels)
40	0.310799	(cereals, vegetables)
41	0.309043	(cheeses, vegetables)
42	0.308165	(dinner rolls, vegetables)
43	0.306409	(dishwashing liquid/detergent, vegetables)
44	0.326602	(eggs, vegetables)
45	0.302897	(ice cream, vegetables)
46	0.309043	(laundry detergent, vegetables)
47	0.311677	(lunch meat, vegetables)
48	0.331870	(poultry, vegetables)
49	0.305531	(soda, vegetables)
50	0.315189	(waffles, vegetables)
51	0.319579	(yogurt, vegetables)

Приложение Ж

Правило для двух товаров на ‘s’

```
two_elems_starts_with_s = lambda df: df[df['itemsets'].apply(
    lambda x:
        np.fromiter(map(lambda y: y.startswith('s'), x),
dtype=bool).sum())>=2
]
print(two_elems_starts_with_s(apriori_results[0]))
```

Приложение И

Данные - два товаров на 's'

	support	itemsets
675	0.137840	(sandwich loaves, sandwich bags)
676	0.146620	(shampoo, sandwich bags)
677	0.158911	(soap, sandwich bags)
678	0.162423	(sandwich bags, soda)
679	0.147498	(spaghetti sauce, sandwich bags)
...
15722	0.064091	(yogurt, sugar, vegetables, soda)
15729	0.058824	(spaghetti sauce, sugar, vegetables, toilet pa...
15730	0.050044	(tortillas, spaghetti sauce, sugar, vegetables)
15731	0.057946	(waffles, sugar, vegetables, spaghetti sauce)
15732	0.061457	(yogurt, spaghetti sauce, sugar, vegetables)

Приложение К

Правило для 0.1-0.25

```
subset_10_25 = lambda df: df[np.logical_and(df.support >= 0.1, df.support <= 0.25)]
```

Приложение Л

Данные - 0.1-0.25

	support	itemsets
38	0.157155	(all- purpose, aluminum foil)
39	0.150132	(all- purpose, bagels)
40	0.144864	(all- purpose, beef)
41	0.147498	(all- purpose, butter)
42	0.151010	(cereals, all- purpose)
...
9136	0.135206	(toilet paper, waffles, vegetables)
9137	0.130817	(toilet paper, yogurt, vegetables)
9139	0.121159	(tortillas, waffles, vegetables)
9140	0.130817	(tortillas, yogurt, vegetables)
9142	0.146620	(yogurt, waffles, vegetables)