Reglas de asociación - MLxtend

Inteligencia Artificial

Prof. Dr. Ariel Monteserin

2021

MLxtend

- MLxtend: http://rasbt.github.io/mlxtend/
  - Clasificación
  - Clustering
  - Reglas de asociación
    - Extracción de itemsets frecuentes
    - Generación de reglas de asociación

MLxtend

- Descubrimiento de itemset frecuentes

```
from mlxtend.frequent_patterns import apriori

frequent_itemsets = apriori(df, min_support=0.6)
```

- `df` es un DataFrame de Pandas con un formato "*one-hot encoded*"

Formato one-hot encoded

	Apple	Corn	Eggs	…	Onion	Milk	Yogurt
0	False	False	False	…	True	False	True
1	False	False	True	…	True	False	True
2	True	False	False	…	False	False	False
3	False	True	False	…	False	True	True
4	False	True	False	…	True	False	False

Preprocesamiento

- Transacciones con items

```
dataset = [['Milk', 'Onion', 'Nutmeg', 'Kidney Beans',
            Eggs', 'Yogurt'],
           ['Dill', 'Onion', 'Nutmeg', 'Kidney Beans',
            Eggs', 'Yogurt'],
           ['Milk', 'Apple', 'Kidney Beans', 'Eggs'],
           ['Milk', 'Unicorn', 'Corn', 'Kidney Beans',
            'Yogurt'],
           ['Corn', 'Onion', 'Onion', 'Kidney Beans',
            'Ice cream', 'Eggs']]
```

- TransactionEncoder() provisto por MLxtend

Preprocesamiento

- Atributos nominales

	Brand	Price
0	Ferrari	high
1	Honda	medium
2	Fiat	low

- get_dummies() provisto por Pandas

	brand_ Ferrari	brand_ Fiat	brand_ Honda	price_ high	price_ low	price_ medium
0	1	0	0	1	0	0
1	0	0	1	0	0	1
2	0	1	0	0	1	0

Itemsets frecuentes

- Descubrimiento de itemset frecuentes

```
from mlxtend.frequent_patterns import apriori

frequent_itemsets = apriori(df, min_support=0.8,
                            use_colnames=True)
```

	support	itemsets
0	0.8	(Eggs)
1	1.0	(Kidney Beans)
2	0.8	(Eggs, Kidney Beans)

Generación de reglas de asociación

- Generación de reglas de asociación

```
from mlxtend.frequent_patterns import
                                    association_rules

rules = association_rules(frequent_itemsets,
            metric="confidence", min_threshold=0.7)
```

Generación de reglas de asociación

	Ant.	Cons.	Ant. support	Cons. support	support.	confidense	lift	leverage	conviction
0	(Eggs)	(Kidney Beans)	0.8	1.0	0.8	1.00	1.00	0.00	inf
1	(Kidney Beans)	(Eggs)	1.0	0.8	0.8	0.80	1.00	0.00	1.0
2	(Onion)	(Eggs)	0.6	0.8	0.6	1.00	1.25	0.12	inf
3	(Eggs)	(Onion)	0.8	0.6	0.6	0.75	1.25	0.12	1.6
4	(Milk)	(Kidney Beans)	0.6	1.0	0.6	1.00	1.00	0.00	inf
5	(Onion)	(Kidney Beans)	0.6	1.0	0.6	1.00	1.00	0.00	inf
6	(Yogurt)	(Kidney Beans)	0.6.	1.0	0.6	1.00	1.00	0.00	inf

Postprocesamiento

- Podemos utilizar las herramientas que provee Pandas
  - `rules[(rules['conviction'] > 1.5) & (rules['lift'] > 1.2)]`
  - `rules[rules['antecedents'] == {'Eggs', 'Kidney Beans'}]`
  - `rules[rules['consequents'].apply(lambda x: 'Onion' in x)]`