# 数据挖掘大作业二：关联规则挖掘

## 1. 数据预处理

对数据集进行处理，转换成适合关联规则挖掘的形式 数据集：dataset1：wine-reviews，来源https://www.kaggle.com/zynicide/wine-reviews 包含两个csv文件： winemag-data-130k-v2.csv：包含14个属性（3个数值属性，11个标称属性），129970条数据记录； winemag-data_first150k.csv：包含11个属性（3个数值属性，8个标称属性），150930条数据记录 该数据集是包括了许多葡萄酒的点评，本次作业打算分析红酒产地、品种的关系。 首先对数据集缺失值进行处理，采用的是用最高频率值来填补缺失值的方法 对数值属性使用'属性=百分比'的编码方式进行离散化，使用四分位数将原属性切分为四部分，使用0-0.25,0.25-0.5,0.5-0.75,0.75-1.0共4个离散化属性来替代原属性

```
import scipy.stats as stats
import pandas as pd
import matplotlib.pyplot as plt
import pylab
%matplotlib inline


df = pd.read_csv('assignment1/wine-reviews/winemag-data-130k-v2.csv', index_col=0)
df.head()
```

```
transactions = []
for index, row in df.iterrows():
    transactions += [(row['country'], row['variety'], row['winery'])]

transactions[:20]
```

```
[('Italy', 'White Blend', 'Nicosia'),
 ('Portugal', 'Portuguese Red', 'Quinta dos Avidagos'),
 ('US', 'Pinot Gris', 'Rainstorm'),
 ('US', 'Riesling', 'St. Julian'),
 ('US', 'Pinot Noir', 'Sweet Cheeks'),
 ('Spain', 'Tempranillo-Merlot', 'Tandem'),
 ('Italy', 'Frappato', 'Terre di Giurfo'),
 ('France', 'Gewürztraminer', 'Trimbach'),
 ('Germany', 'Gewürztraminer', 'Heinz Eifel'),
 ('France', 'Pinot Gris', 'Jean-Baptiste Adam'),
 ('US', 'Cabernet Sauvignon', 'Kirkland Signature'),
 ('France', 'Gewürztraminer', 'Leon Beyer'),
 ('US', 'Cabernet Sauvignon', 'Louis M. Martini'),
 ('Italy', 'Nerello Mascalese', 'Masseria Setteporte'),
 ('US', 'Chardonnay', 'Mirassou'),
 ('Germany', 'Riesling', 'Richard Böcking'),
 ('Argentina', 'Malbec', 'Felix Lavaque'),
 ('Argentina', 'Malbec', 'Gaucho Andino'),
 ('Spain', 'Tempranillo Blend', 'Pradorey'),
 ('US', 'Meritage', 'Quiévremont')]
```

```
# 把数值属性列作离散化
def ParticalDiscretization(df, nume_attr=[], bina_attr=[]):
    new_df = copy.deepcopy(df[bina_attr]);
    for i in nume_attr:
        new_attr = [i + '=0~0.25', i + '=0.25~0.5', i + '=0.5~0.75', i + '=0.75~1.0'];
        tmp = pandas.DataFrame(columns=new_attr, index=df.index);
        k = 0;
        for j in df[i]:
            if j >= df[i].quantile(0.75):
                tmp[i + '=0.75~1.0'][k] = 1;
            elif j >= df[i].quantile(0.5):
                tmp[i + '=0.5~0.75'][k] = 1;
            elif j >= df[i].quantile(0.25):
                tmp[i + '=0.25~0.5'][k] = 1;
            elif j >= df[i].quantile(0):
                tmp[i + '=0~0.25'][k] = 1;
            k = k + 1;
        new_df = pandas.concat([new_df, tmp], axis=1);
    new_df = new_df.fillna(value=0);
    return new_df;
```

## 2. 找出频繁模式

使用apriori算法

```
from collections import defaultdict
import itertools


def apriori(transactions, support=0.1, confidence=0.8, lift=1, minlen=2, maxlen=2):
    item_2_tranidxs = defaultdict(list)
    itemset_2_tranidxs = defaultdict(list)

    for tranidx, tran in enumerate(transactions):
        for item in tran:
            item_2_tranidxs[item].append(tranidx)
            itemset_2_tranidxs[frozenset([item])].append(tranidx)

    item_2_tranidxs = dict([(k, frozenset(v)) for k, v in item_2_tranidxs.items()])
    itemset_2_tranidxs = dict([
        (k, frozenset(v)) for k, v in itemset_2_tranidxs.items()])

    tran_count = float(len(transactions))
    # print('Extracting rules in {} transactions...'.format(int(tran_count)))

    valid_items = set(item
        for item, tranidxs in item_2_tranidxs.items()
            if (len(tranidxs) / tran_count >= support))

    pivot_itemsets = [frozenset([item]) for item in valid_items]
    freqsets = []

    if minlen == 1:
        freqsets.extend(pivot_itemsets)

    for i in range(maxlen - 1):
        new_itemset_size = i + 2
        new_itemsets = []

        for pivot_itemset in pivot_itemsets:
            pivot_tranidxs = itemset_2_tranidxs[pivot_itemset]
            for item, tranidxs in item_2_tranidxs.items():
                if item not in pivot_itemset:
                    common_tranidxs = pivot_tranidxs & tranidxs
                    if len(common_tranidxs) / tran_count >= support:
                        new_itemset = frozenset(pivot_itemset | set([item]))
                        if new_itemset not in itemset_2_tranidxs:
                            new_itemsets.append(new_itemset)
                            itemset_2_tranidxs[new_itemset] = common_tranidxs

        if new_itemset_size > minlen - 1:
            freqsets.extend(new_itemsets)

        pivot_itemsets = new_itemsets

    # print('{} frequent patterns found'.format(len(freqsets)))

    for freqset in freqsets:
        for item in freqset:
            rhs = frozenset([item])
            lhs = freqset - rhs
            support_rhs = len(itemset_2_tranidxs[rhs]) / tran_count
            if len(lhs) == 0:
                lift_rhs = float(1)
                if support_rhs >= support and support_rhs > confidence and lift_rhs > lift:
                    yield (lhs, rhs, support_rhs, support_rhs, lift_rhs)
            else:
                confidence_lhs_rhs = len(itemset_2_tranidxs[freqset]) \
                    / float(len(itemset_2_tranidxs[lhs]))
                lift_lhs_rhs = confidence_lhs_rhs / support_rhs

                if confidence_lhs_rhs >= confidence and lift_lhs_rhs > lift:
                    support_lhs_rhs = len(itemset_2_tranidxs[freqset]) / tran_count
                    yield (lhs, rhs, support_lhs_rhs, confidence_lhs_rhs, lift_lhs_rhs)
```

频繁项集（support>0.03, confidence>0.1, lift>1）如下：

```
rules = apriori(transactions, support=0.03, confidence=0.1, lift=1)
rules_sorted = sorted(rules, key=lambda x: (x[4], x[3], x[2]), reverse=True) # ORDER BY lift DESC, confidence
DESC, support DESC

for r in rules_sorted:
    print(r)
```

```
(frozenset({'Bordeaux-style Red Blend'}), frozenset({'France'}), 0.03635426364342815, 0.6832971800433839,
4.019771773295553)
(frozenset({'France'}), frozenset({'Bordeaux-style Red Blend'}), 0.03635426364342815, 0.2138686461775223,
4.019771773295553)
(frozenset({'Cabernet Sauvignon'}), frozenset({'US'}), 0.05628178593686284, 0.7722761824324325,
1.841580575864628)
(frozenset({'US'}), frozenset({'Cabernet Sauvignon'}), 0.05628178593686284, 0.13421033318655512,
1.841580575864628)
(frozenset({'Pinot Noir'}), frozenset({'US'}), 0.07605542774926714, 0.7448010849909584, 1.7760630745882846)
(frozenset({'US'}), frozenset({'Pinot Noir'}), 0.07605542774926714, 0.18136283575517392, 1.7760630745882844)
(frozenset({'Chardonnay'}), frozenset({'US'}), 0.052327057574381976, 0.5786607674636263, 1.3798825518863749)
(frozenset({'US'}), frozenset({'Chardonnay'}), 0.052327057574381976, 0.12477983267283135, 1.3798825518863749)
```

## 3. 导出关联规则及其支持度，置信度

```
import csv

with open('result.csv', 'wt') as f:
    f_csv = csv.writer(f, delimiter=',')
    f_csv.writerow(['rule', 'sup', 'conf', 'lift'])
    for r in rules_sorted:
        f_csv.writerow([f'{str(list(r[0])[0])} => {str(list(r[1])[0])}', r[2], r[3], r[4]])

pd.read_csv('result.csv')
```

```
.dataframe tbody tr th {
    vertical-align: top;
}

.dataframe thead th {
    text-align: right;
}
```

| | rule | sup | conf | lift |
|---|---|---|---|---|
| **0** | Bordeaux-style Red Blend => France | 0.036354 | 0.683297 | 4.019772 |
| **1** | France => Bordeaux-style Red Blend | 0.036354 | 0.213869 | 4.019772 |
| **2** | Cabernet Sauvignon => US | 0.056282 | 0.772276 | 1.841581 |
| **3** | US => Cabernet Sauvignon | 0.056282 | 0.134210 | 1.841581 |
| **4** | Pinot Noir => US | 0.076055 | 0.744801 | 1.776063 |
| **5** | US => Pinot Noir | 0.076055 | 0.181363 | 1.776063 |
| **6** | Chardonnay => US | 0.052327 | 0.578661 | 1.379883 |
| **7** | US => Chardonnay | 0.052327 | 0.124780 | 1.379883 |

## 4. 对规则进行评价，使用Lift， Kulc

上一步已经计算了Lift，这里再计算一下Kulc。

```
res = []
for r in rules_sorted:
    conf1 = r[3]
    for r2 in rules_sorted:
        if r2[0] == r[1] and r2[1] == r[0]:
            conf2 = r2[3]
    kulc = (conf1 + conf2) / 2
    res.append(kulc)

res
```

```
[0.4485829131104531,
 0.4485829131104531,
 0.4532432578094938,
 0.4532432578094938,
 0.46308196037306615,
 0.46308196037306615,
 0.3517203000682288,
 0.3517203000682288]
```

## 5. 对挖掘结果进行分析

这里以Bordeaux-style Red Blend => France为例。

由关联规则可知Bordeaux-style Red Blend这个品种的葡萄酒基本上产自法国，那么我们就来检验一下：

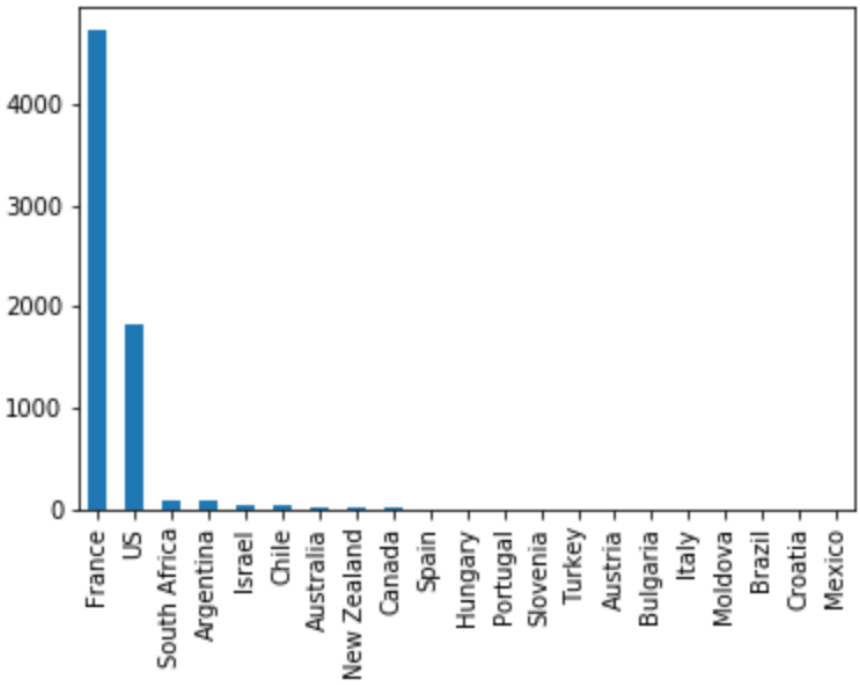
```
df[df['variety'] == 'Bordeaux-style Red Blend'].sample(20)
```

| | country | description | designation | points | price | province | region_1 | region_2 | taster_name | taster_twitter_handle | title |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **109896** | France | Named after the Swedish royal family, this est... | NaN | 89 | NaN | Bordeaux | Haut-Médoc | NaN | Roger Voss | @vossroger | Château Bernadotte 2013 Haut-Médoc |
| **31828** | France | Ripe fruit, juicy acidity and a fine balance o... | Boha | 90 | 17.0 | Bordeaux | Blaye Côtes de Bordeaux | NaN | Roger Voss | @vossroger | Château Magdeleine Bouhou 2015 Boha (Blaye Cô... |
| **64213** | France | Wood and smoke aromas precede ripe and dusty t... | Famille Lapalu | 86 | 10.0 | Bordeaux | Médoc | NaN | Roger Voss | @vossroger | Domaines Lapalu 2008 Famille Lapalu (Médoc) |
| **116444** | France | This is rounded, with its ripe fruit dominatin... | Cuvée Prestige | 87 | 14.0 | Bordeaux | Bordeaux Supérieur | NaN | Roger Voss | @vossroger | Château de Cornemps 2009 Cuvée Prestige (Bord... |
| **77402** | France | This wine is firm with plenty of structured ta... | NaN | 93 | NaN | Bordeaux | Margaux | NaN | Roger Voss | @vossroger | Château Giscours 2013 Margaux |
| **86644** | France | This ripe, bold and generous Gonfrier Frères w... | NaN | 89 | 14.0 | Bordeaux | Bordeaux | NaN | Roger Voss | @vossroger | Château Tassin 2015 Bordeaux |
| **36585** | US | This wine is made from a majority of Cabernet ... | Winston Hill | 92 | 150.0 | California | Rutherford | Napa | Virginie Boone | @vboone | Frank Family 2012 Winston Hill Red (Rutherford) |
| **51398** | US | Full bodied, with structured tannins and brigh... | New World Red | 85 | 34.0 | Virginia | Monticello | NaN | NaN | NaN | Kluge Estate 2009 New World Red Red (Monticello) |
| **103898** | US | A proprietary blend of 36% Cabernet Sauvignon,... | Contrarian | 91 | 135.0 | California | Napa Valley | Napa | Virginie Boone | @vboone | Blackbird Vineyards 2013 Contrarian Red (Napa ... |
| **78741** | France | This is a firmly structured wine, solid with f... | NaN | 92 | NaN | Bordeaux | Pessac-Léognan | NaN | Roger Voss | @vossroger | Château Carbonnieux 2014 Pessac-Léognan |
| **96518** | US | Based on Merlot, this Bordeaux-style blend is ... | Bastille | 86 | 38.0 | California | Sonoma County | Sonoma | NaN | NaN | De Novo 2009 Bastille Red (Sonoma County) |
| **122324** | US | After initial scents of smoke and dark toast s... | Corchaug Estate Ben's Blend | 89 | 48.0 | New York | North Fork of Long Island | Long Island | Anna Lee C. Iijima | NaN | McCall 2007 Corchaug Estate Ben's Blend Red (N... |

| | country | description | designation | points | price | province | region_1 | region_2 | taster_name | taster_twitter_handle | title |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **56833** | US | This wine is a blend of Merlot (44%), Cabernet... | Two Blondes Vineyard | 91 | 64.0 | Washington | Yakima Valley | Columbia Valley | Sean P. Sullivan | @wawinereport | Andrew Will 2013 Two Blondes Vineyard Red (Yak... |
| **63552** | France | An impressive blend of Merlot and Malbec, this... | Comtesse de Ségur | 90 | 19.0 | Southwest France | Montravel | NaN | Roger Voss | @vossroger | Château Laulerie 2012 Comtesse de Ségur (Mont... |
| **75613** | France | 94-96 Barrel sample. Full of blackcurrant frui... | Barrel sample | 95 | NaN | Bordeaux | Margaux | NaN | Roger Voss | @vossroger | Château Giscours 2009 Barrel sample (Margaux) |
| **33862** | US | High alcohol, softness and tremendously ripe f... | Maquette | 87 | 38.0 | California | Paso Robles | Central Coast | NaN | NaN | Sculpterra 2010 Maquette Red (Paso Robles) |
| **99251** | France | This structured wine has 25-year old vines as ... | NaN | 88 | 10.0 | Bordeaux | Bordeaux | NaN | Roger Voss | @vossroger | Château Arnaucosse 2012 Bordeaux |
| **65728** | Argentina | This is an elegant wine that shows that Argent... | Pasionado Cuatro Cepas | 91 | 50.0 | Mendoza Province | Mendoza | NaN | Michael Schachner | @wineschach | Andeluna 2005 Pasionado Cuatro Cepas Red (Mend... |
| **57233** | France | The vineyard surrounds a 14th century castle, ... | NaN | 90 | 36.0 | Bordeaux | Puisseguin Saint-Émilion | NaN | Roger Voss | @vossroger | Château Langlais 2010 Puisseguin Saint-Émilion |
| **129193** | France | Layered tannins and wood flavors have tended t... | NaN | 84 | 18.0 | Bordeaux | Bordeaux Supérieur | NaN | Roger Voss | @vossroger | Château Laville 2015 Bordeaux Supérieur |

再绘制一个直方图如下，可以看出，Bordeaux-style Red Blend这个品种的葡萄酒产自法国的确比较多。

```
df[df['variety'] == 'Bordeaux-style Red Blend']['country'].value_counts().plot(kind='bar')
```




对规则进行可视化如下

```
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('result.csv')
plt.scatter(df['sup'], df['conf'], c=df['lift'], s=20, cmap='Reds')
plt.xlabel('sup')
plt.ylabel('conf')
cb = plt.colorbar()
cb.set_label('lift')
plt.show()
```