CDNow 网站的用户购买明细——数据分析报告

【项目背景】

案例数据来自 CDNow 网站的用户购买明细，一共分为四部分：用户 ID，购买日期，购买数量，购买金额.我们要通过数据分析，归纳研究结论，并完成一份基础的数据分析报告。

【分析方法】

在 anaconda 软件上进行数据分析操作，观察并清洗数据后，对数据按月维度和用户维度进行切割，数据透视用户的消费数据；分别计算出用户复购率、回购率；然后完成用户回流占比、生命周期、留存率等计算工作。

【分析结论】

1、 首先，观察数据。

用户平均每笔订单购买 2.4 个商品，标准差在 2.3，稍稍具有波动性。中位数在 2 个商品，75 分位数在 3 个商品，说明绝大部分订单的购买量都不多。最大值在 99 个，数字比较高。购买金额的情况差不多，大部分订单都集中在小额。

```
In [4]: df.describe()
```

Out[4]:

	user_id	order_dt	order_products	order_amount
count	69659.000000	6.965900e+04	69659.000000	69659.000000
mean	11470.854592	1.997228e+07	2.410040	35.893648
std	6819.904848	3.837735e+03	2.333924	36.281942
min	1.000000	1.997010e+07	1.000000	0.000000
25%	5506.000000	1.997022e+07	1.000000	14.490000
50%	11410.000000	1.997042e+07	2.000000	25.980000
75%	17273.000000	1.997111e+07	3.000000	43.700000
max	23570.000000	1.998063e+07	99.000000	1286.010000

没有空值，很干净的数据。

```
In [5]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 69659 entries, 0 to 69658
Data columns (total 4 columns):
user_id           69659 non-null int64
order_dt          69659 non-null int64
order_products    69659 non-null int64
order_amount      69659 non-null float64
dtypes: float64(1), int64(3)
memory usage: 2.1 MB
```

2、 从用户维度和月维度进行分析。

从用户角度看，每位用户平均购买 7 张 CD，最多的用户购买了 1033 张，属于狂热用户了。用户的平均消费金额（客单价）100 元，标准差是 240，结合分位数和最大值看，平均值才和 75 分位接近，肯定存在小部分的高额消费用户。

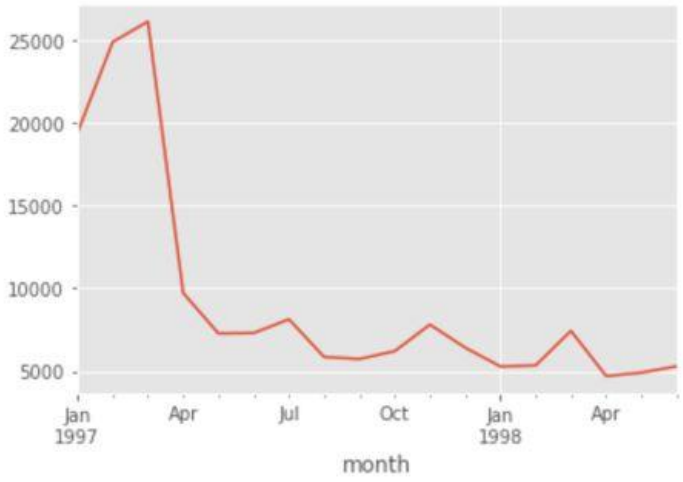
```
In [12]: user_grouped.describe()
```

Out[12]:

	order_dt	order_products	order_amount
count	2.357000e+04	23570.000000	23570.000000
mean	5.902627e+07	7.122656	106.080426
std	9.460684e+07	16.983531	240.925195
min	1.997010e+07	1.000000	0.000000
25%	1.997021e+07	1.000000	19.970000
50%	1.997032e+07	3.000000	43.395000
75%	5.992125e+07	7.000000	106.475000
max	4.334408e+09	1033.000000	13990.930000

按月的维度分析，按月统计每个月的 CD 销量。从图中可以看到，前几个月的销量非常高涨。数据比较异常。而后期的销量则很平稳。

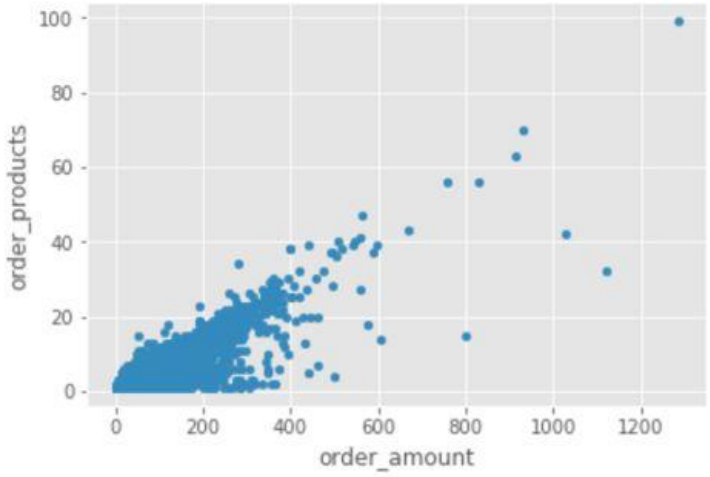
```
In [13]: df.groupby('month').order_products.sum().plot()
Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0x115417320>
```



3、 绘制每笔订单的散点图。

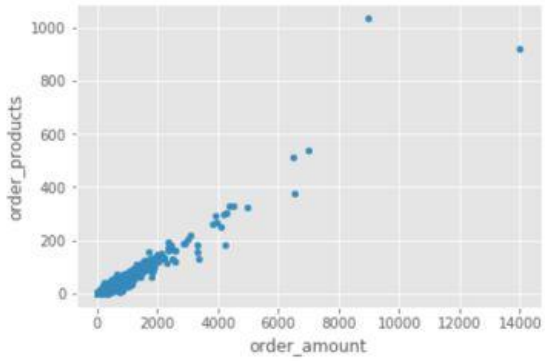
从图中观察，订单消费金额和订单商品量呈规律性，每个商品十元左右。订单的极值较少，超出 1000 的就几个。

```
In [15]: df.plot.scatter(x = 'order_amount',y = 'order_products')
Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x10c57fe80>
```



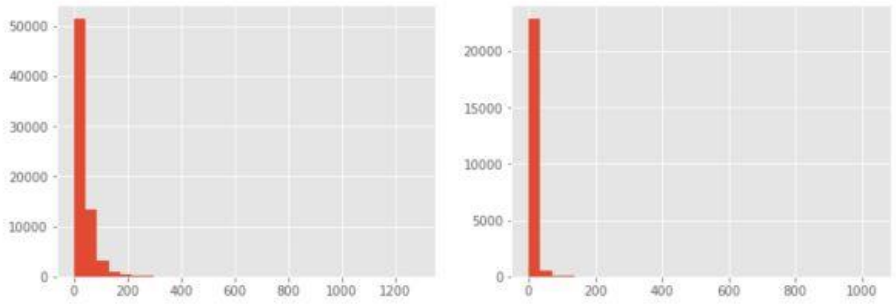
绘制用户的散点图，用户也比较健康，而且规律性比订单更强。因为这是 CD 网站的销售数据，商品比较单一，金额和商品量的关系也因此呈线性，没几个离群点。

```
In [16]: df.groupby('user_id').sum().plot.scatter(x = 'order_amount',y = 'order_products')
Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x1168fe4a8>
```



为了更好的观察，用直方图。

```
In [17]: plt.figure(figsize=(12,4))
         plt.subplot(121)
         df.order_amount.hist(bins = 30)

         plt.subplot(122)
         df.groupby('user_id').order_products.sum().hist(bins = 30)
Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x11624e9e8>
```



4、 接下来看消费的时间节点。

观察用户的最后一次消费时间。绝大部分数据依然集中在前三个月。后续的时间段内，依然有用户在消费，但是缓慢减少。

```
In [19]: df.groupby('user_id').month.max().value_counts()

Out[19]: 1997-02-01    4912
         1997-03-01    4478
         1997-01-01    4192
         1998-06-01    1506
         1998-05-01    1042
         1998-03-01     993
         1998-04-01     769
         1997-04-01     677
         1997-12-01     620
         1997-11-01     609
         1998-02-01     550
         1998-01-01     514
         1997-06-01     499
         1997-07-01     493
         1997-05-01     480
         1997-10-01     455
         1997-09-01     397
         1997-08-01     384
         Name: month, dtype: int64
```

5、 接下来分析消费中的复购率和回购率。
首先将用户消费数据进行数据透视。

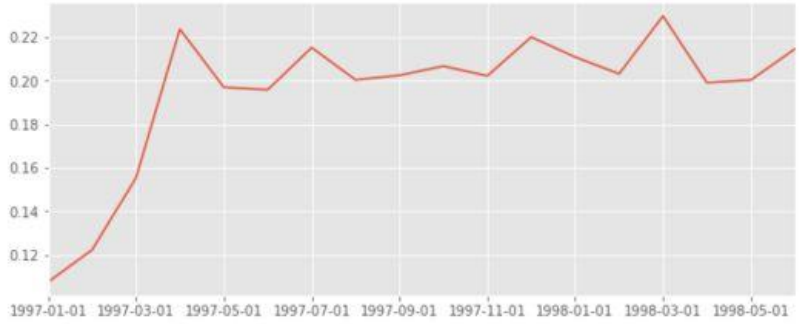
```
In [20]: pivoted_counts = df.pivot_table(index = 'user_id',columns = 'month',
                                         values='order_dt',aggfunc = 'count').fillna(0)
         columns_month = df.month.sort_values().astype('str').unique()
         pivoted_counts.columns = columns_month

         pivoted_counts.head()
```

Out[20]:

user_id	1997-01-01	1997-02-01	1997-03-01	1997-04-01	1997-05-01	1997-06-01	1997-07-01	1997-08-01	1997-09-01	1997-10-01	1997-11-01	1997-12-01	19
1	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	2.0	0.0	
4	2.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	1.0	
5	2.0	1.0	0.0	1.0	1.0	1.0	1.0	0.0	1.0	0.0	0.0	2.0	

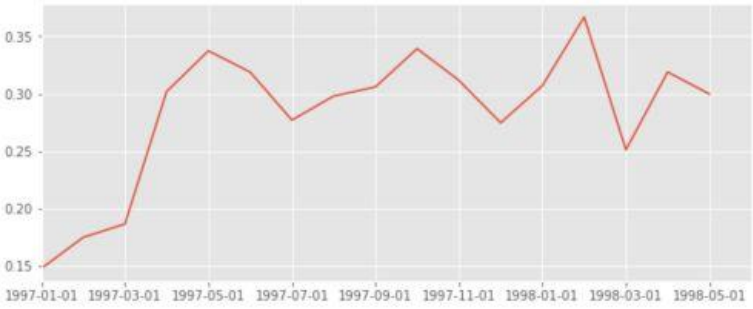
下图可以看出复购率在早期，因为大量新用户加入的关系，新客的复购率并不高，譬如1 月新客们的复购率只有 6%左右。而在后期，这时的用户都是大浪淘沙剩下的老客，复购率比较稳定，在 20%左右。

```
In [22]: (pivoted_counts_transf.sum() / pivoted_counts_transf.count()).plot(figsize = (10,4))
Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0x116be1f60>
```



从下图中可以看出，用户的回购率高于复购，约在 30%左右，波动性也较强。新用户的回购率在 15%左右，和老客差异不大。

```
In [30]: (pivoted_purchase_return.sum() / pivoted_purchase_return.count()).plot(figsize = (10,4))
Out[30]: <matplotlib.axes._subplots.AxesSubplot at 0x1163ada58>
```



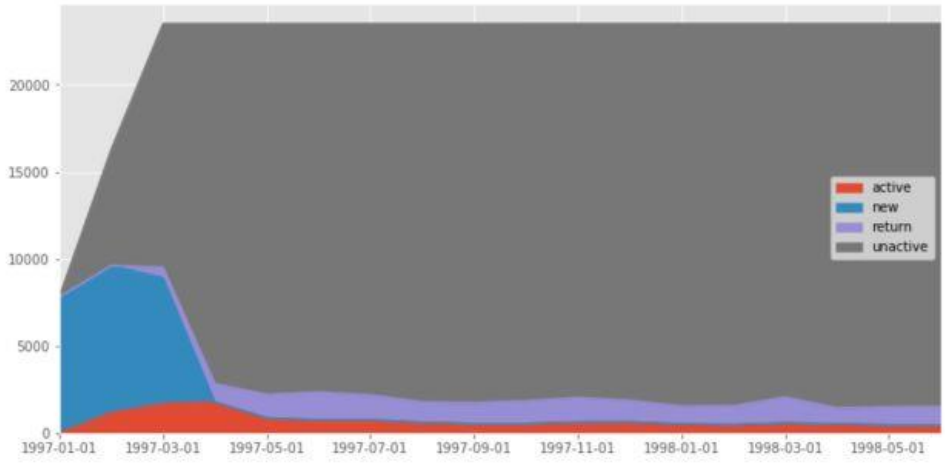
将回购率和复购率综合分析，可以得出，新客的整体质量低于老客，老客的忠诚度（回购率）表现较好，消费频次稍次，这是 CDNow 网站的用户消费特征。

6、 接下来进行用户分层。

我们按照用户的消费行为，简单划分成几个维度：新用户、活跃用户、不活跃用户、回流用户。

新用户的定义是第一次消费。活跃用户即老客，在某一个时间窗口内有过消费。不活跃用户则是时间窗口内没有消费过的老客。回流用户是在上一个窗口中没有消费，而在当前时间窗口内有过消费。以上的时间窗口都是按月统计。

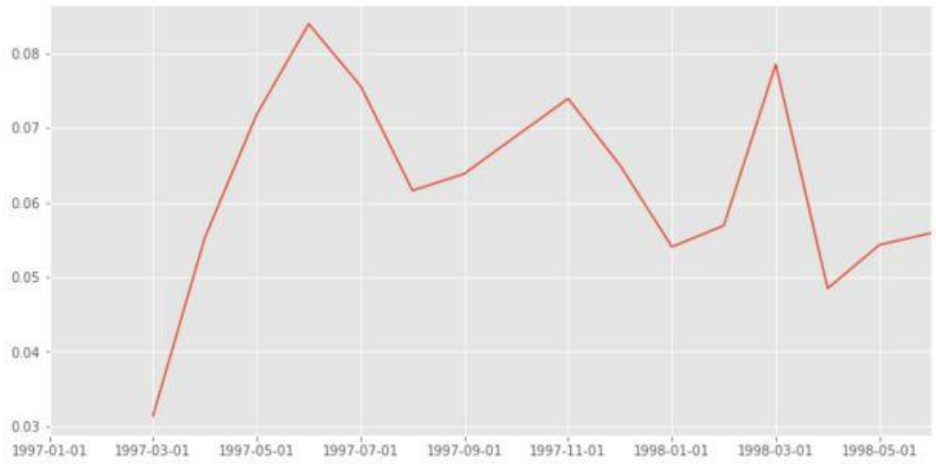
```
In [33]: purchase_status_counts.fillna(0).T.plot.area(figsize = (12,6))
Out[33]: <matplotlib.axes._subplots.AxesSubplot at 0x115c6bc50>
```



7、 用户回流占比。

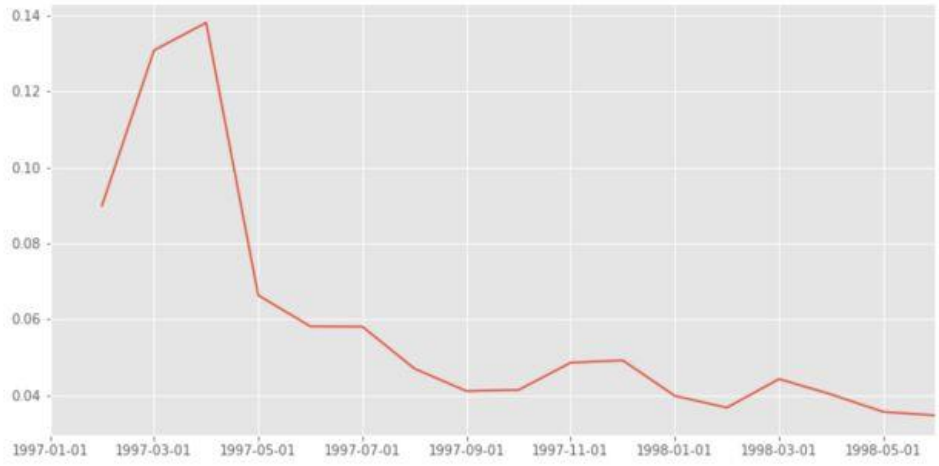
用户回流占比在 5%～8%，有下降趋势。

```
In [34]: return_rata = purchase_status_counts.apply(lambda x:x / x.sum(),axis =1)
         return_rata.loc['return'].plot(figsize = (12,6))
Out[34]: <matplotlib.axes._subplots.AxesSubplot at 0x11c3c64a8>
```



活跃用户的下降趋势更明显，占比在 3%～5%间。这里用户活跃可以看作连续消费用户，质量在一定程度上高于回流用户。

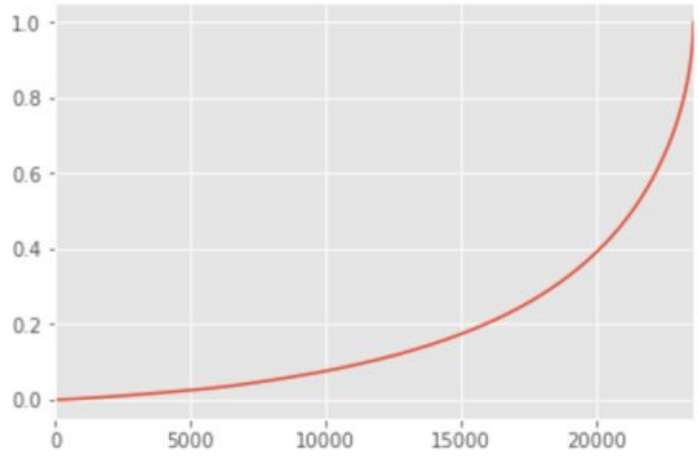
```
In [35]: return_rata.loc['active'].plot(figsize = (12,6))
Out[35]: <matplotlib.axes._subplots.AxesSubplot at 0x115d471d0>
```



8、 绘制用户购买趋势图。

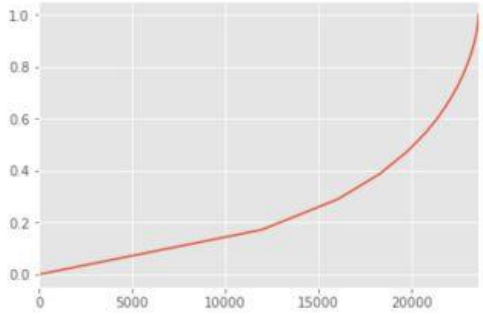
横坐标是按贡献金额大小排序而成，纵坐标则是用户累计贡献。可以很清楚的看到，前20000 个用户贡献了 40%的消费。后面 4000 位用户贡献了 60%，确实呈现消费领域经典的 28 倾向。

```
In [39]: user_amount.prop.plot()
Out[39]: <matplotlib.axes._subplots.AxesSubplot at 0x115d3cfd0>
```



统计一下销量，前两万个用户贡献了 45%的销量，高消费用户贡献了 55%的销量。

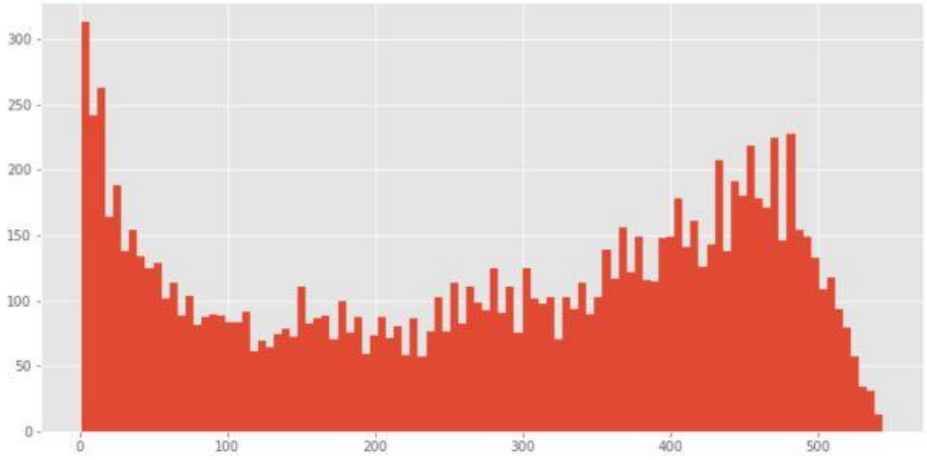
```
In [40]: user_counts = df.groupby('user_id').order_dt.count().sort_values().reset_index()
         user_counts['counts_cumsum'] = user_counts.order_dt.cumsum()
         counts_total = user_counts.counts_cumsum.max()
         user_counts['prop'] = user_counts.apply(lambda x:x.counts_cumsum / counts_total,axis =1)
         user_counts.prop.plot()
Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x115d6c630>
```



10、用户的生命周期

如下图是双峰趋势图。部分质量差的用户，虽然消费了两次，但是仍旧无法持续，在用户首次消费 30 天内应该尽量引导。少部分用户集中在 50 天～300 天，属于普通型的生命周期，高质量用户的生命周期，集中在 400 天以后，这已经属于忠诚用户了，大家有兴趣可以跑一下 400 天+的用户占老客比多少，占总量多少。

```
In [46]: life_time['life_time'] = life_time.order_date / np.timedelta64(1, 'D')
         life_time[life_time.life_time > 0].life_time.hist(bins = 100,figsize = (12,6))
Out[46]: <matplotlib.axes._subplots.AxesSubplot at 0x11c0331d0>
```

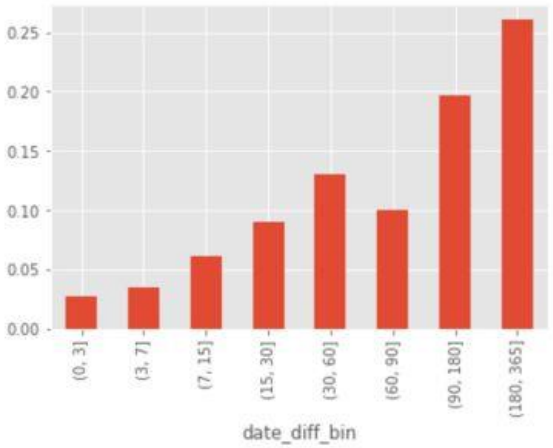


11、再来计算留存率

留存率也是消费分析领域的经典应用。它指用户在第一次消费后，有多少比率进行第二次消费。和回流率的区别是留存倾向于计算第一次消费，并且有多个时间窗口。

只有 2.5%的用户在第一次消费的次日至 3 天内有过消费，3%的用户在 3～7 天内有过消费。数字并不好看，CD 购买确实不是高频消费行为。时间范围放宽后数字好看了不少，有 20%的用户在第一次消费后的三个月到半年之间有过购买，27%的用户在半年后至 1 年内有过购买。从运营角度看，CD 机营销在教育新用户的同时，应该注重用户忠诚度的培养，放长线掉大鱼，在一定时间内召回用户购买。

```
In [57]: (pivoted_retention_trans.sum() / pivoted_retention_trans.count()).plot.bar()
Out[57]: <matplotlib.axes._subplots.AxesSubplot at 0x116923668>
```



12、计算出用户的平均购买周期

看一下直方图，典型的长尾分布，大部分用户的消费间隔确实比较短。不妨将时间召回点设为消费后立即赠送优惠券，消费后 10 天询问用户 CD 怎么样，消费后 30 天提醒优惠券到期，消费后 60 天短信推送。这样便可有效的利用本文的数据结论了。

```
In [65]: last_diff.hist(bins = 20)
Out[65]: <matplotlib.axes._subplots.AxesSubplot at 0x11d0a1550>
```