卓越二班 刘瑞康 2016302580242

1.请思考数据挖掘可能会遇到哪些法律问题，可能会和哪些法律有关，请举出具体例子并讨论

Data mining may encounter legal issues in aspects of personal, business, and government. From all aspects, the right of personal privacy is the most seriously violated. As an example, insurers can use data mining to check the medical records of their clients to cut insurance spending by efusing to sell insurance to customers with predispositions. What’s worse, PRISM(棱镜计划), the name for a program under which the United States National Security Agencycollects internet communications from various U.S. internet companies, exactly takes advantage of data mining to sniff data on computer networks worldwide through industrial-scale systems. According to documents disclosed by Snowden, the National Security Agency has access to a large number of personal chat logs, stored data, voice communications, file transfers, and personal social network data. That’s rather more serious violations of citizens’ privacy rights from the national level. What is certain is that data mining has an indispensable role in the era of big data. It has an important impact on the development of many industries and even society. However, if the existing legal problems are restrained and rectified, the negative impact of data mining will seriously hinder its development.

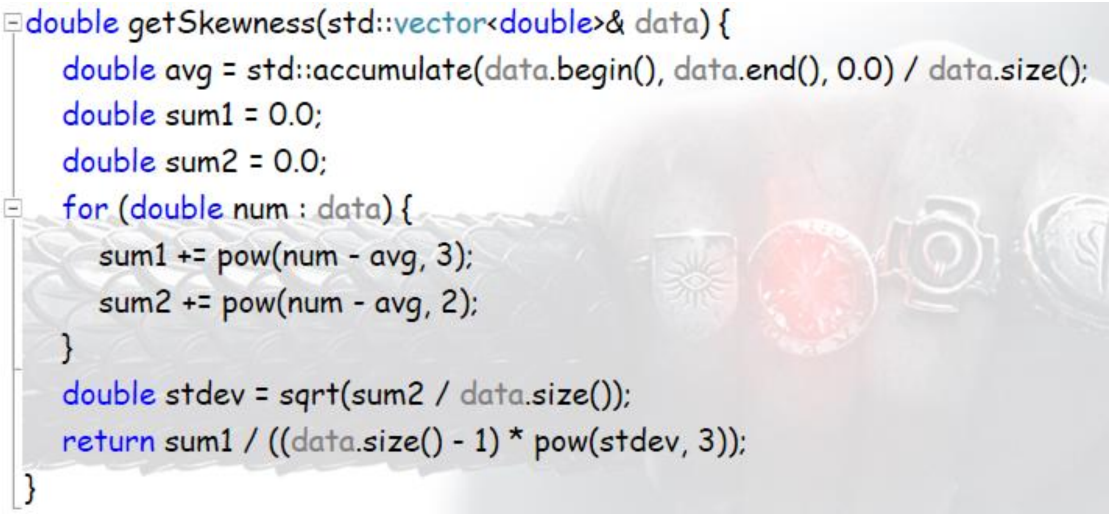
2. Skewness 是什么，请计算对称正态分布，正偏移和负偏移时候的 Skewness 的指（请自行拟定数据分布具体数值）

In a symmetric data distribution, the mean, median, and mode are all at the same center value. However, data in most real applications are not symmetric, in which cases we use Skewness to decribe the distribution. Skewness has two forms: positively skewed, the mode occurs at a value that is smaller than the median; negatively skewed, the mode occurs at a value greater than the median.

Skewness can be computed by the formula:

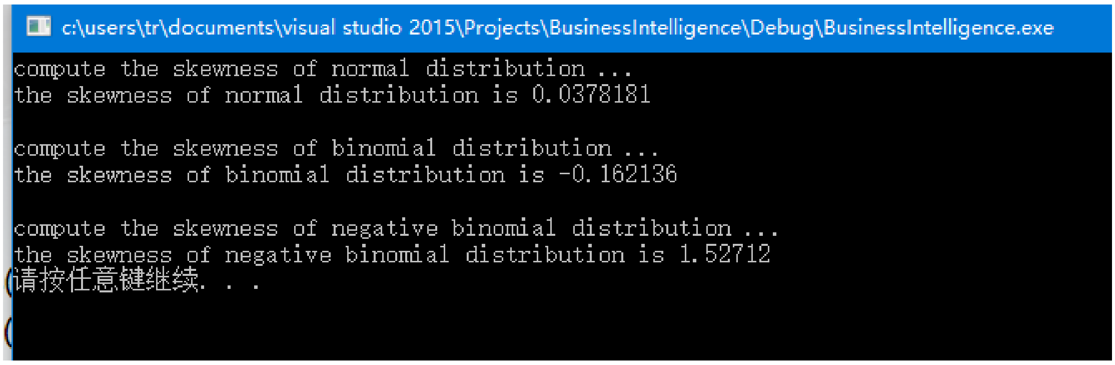
$$Skewness = S = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^3}{(n - 1)s^3}$$

Implemented in C++:



```cpp
double getSkewness(std::vector<double>& data) {
    double avg = std::accumulate(data.begin(), data.end(), 0.0) / data.size();
    double sum1 = 0.0;
    double sum2 = 0.0;
    for (double num : data) {
        sum1 += pow(num - avg, 3);
        sum2 += pow(num - avg, 2);
    }
    double stdev = sqrt(sum2 / data.size());
    return sum1 / ((data.size() - 1) * pow(stdev, 3));
}
```

Compute the skewness of normal distribution / binomial distribution / negative binomial distribution, the data set is generated randomly. See deails in the **skewness.cpp**. Here is the result:

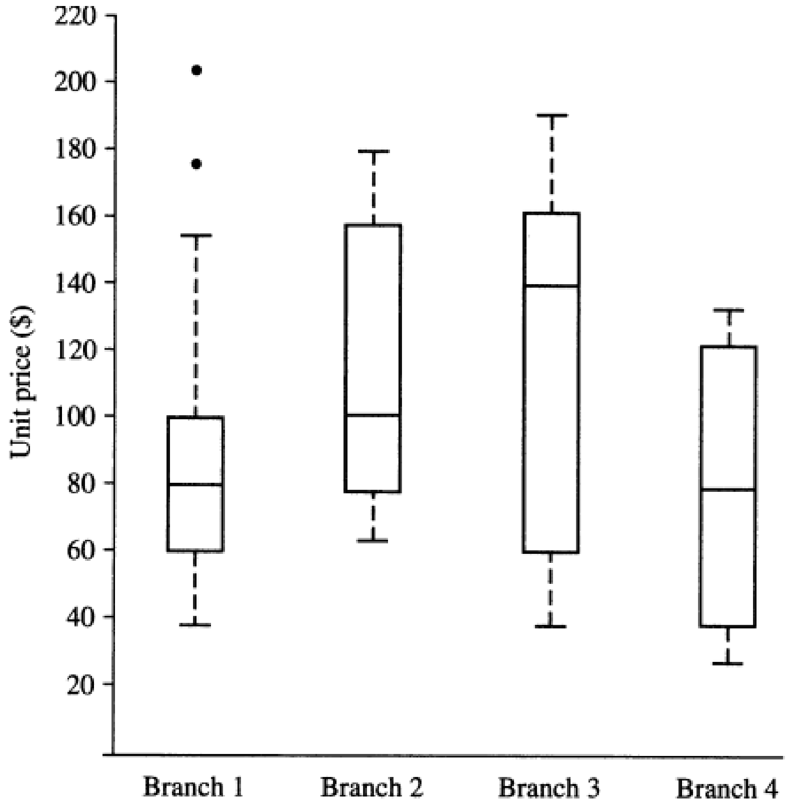

c:\users\tr\documents\visual studio 2015\Projects\BusinessIntelligence\Debug\BusinessIntelligence.exe

```
compute the skewness of normal distribution ...
the skewness of normal distribution is 0.0378181

compute the skewness of binomial distribution ...
the skewness of binomial distribution is -0.162136

compute the skewness of negative binomial distribution ...
the skewness of negative binomial distribution is 1.52712
请按任意键继续. . .
```

3. 请对下图进行分析



For branch1, the median price of items sold is $80, with prices ranging from $40 to $155. Two outlying observations for this branch were plotted individually and need checking, as their values are more than 1.5 times the IQR.

For branch2, the median price of items sold is $100, with prices ranging from about $60 to $180.

For branch2, the median price of items sold is $140, with prices ranging from about $40 to $190. The prices for this branch are dispersive.

For branch4, the median price of items sold is $80, with prices ranging from about $30 to $130. The prices for this branch are evenly centrally distributed.

4. Q-Q plot 中如果两个数据集的数目不相同该如何处理?

(1)The best way is to compute the percentile of each data set, with which draw the Q-Q plot instead of using the raw data.

(2)Another way is to proportionately delete or add data in the data sets to make them have the same size.

5.请自拟数据集计算混合类型相似度矩阵，数据应报告所有不同的类型，且属性总数目不低于10个

Define “Phone” with ten attributes such that

numeric attributes:

price; batteryCapacity; memoryCapacity; storageCapacity; sizeOfScreen;

ordinal attributes

cpuModel;

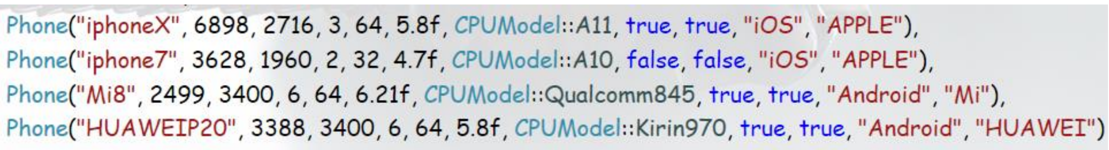
binary attributes

support2SIMs; supportFastCharge;

Nominal attributes

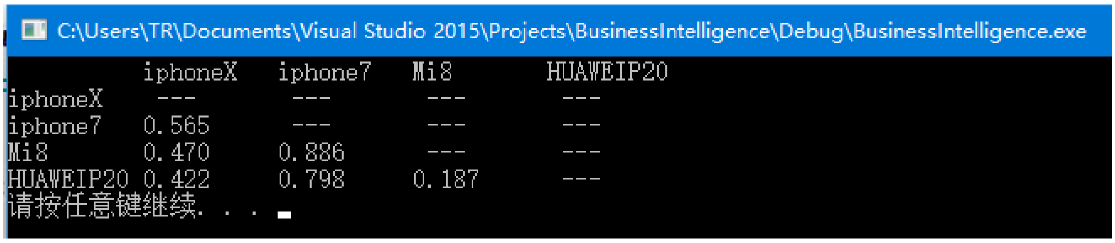
operatingSystem; brand;

compute the dissimilarity among iphoneX iphone7 Mi8 HUAWEIP20



```
Phone("iphoneX", 6898, 2716, 3, 64, 5.8f, CPUModel::A11, true, true, "iOS", "APPLE"),
Phone("iphone7", 3628, 1960, 2, 32, 4.7f, CPUModel::A10, false, false, "iOS", "APPLE"),
Phone("Mi8", 2499, 3400, 6, 64, 6.21f, CPUModel::Qualcomm845, true, true, "Android", "Mi"),
Phone("HUAWEIP20", 3388, 3400, 6, 64, 5.8f, CPUModel::Kirin970, true, true, "Android", "HUAWEI")
```

See deails in the **dissimilarity.cpp**. The result is


C:\Users\TR\Documents\Visual Studio 2015\Projects\BusinessIntelligence\Debug\BusinessIntelligence.exe

```
          iphoneX   iphone7   Mi8       HUAWEIP20
iphoneX    ---       ---       ---       ---
iphone7   0.565      ---       ---       ---
Mi8       0.470     0.886      ---       ---
HUAWEIP20 0.422     0.798     0.187      ---
请按任意键继续. . .
```

6.请编写程序实现 Apriori 算法和 FP-Growth 算法，算法可以根据给定的支持度和置信度获取所有的频繁项集和关联规则。请自拟数据集进行测试（数据集应按某种方式产生，请描述产生机制，数据集不少于 1 万条，商品数目不小于 100 种）并汇报以下结果并撰写报告汇报结果，

1）给定置信度为 80%，关联规则数目随支持度变化的曲线图

2）给定支持度为 30%，关联规则数目随置信度变化的曲线图

3）给定置信度为 80%，请确定某个支持度 s 使得获取的关联规则数目正好大于 20，请输出 s 和所获取的关联规则数目

the algorithm is implemented by python3.6 including four .py files – data.py, aprior.py, fp.py and main.py. See deails in the **folder 'BI'**.

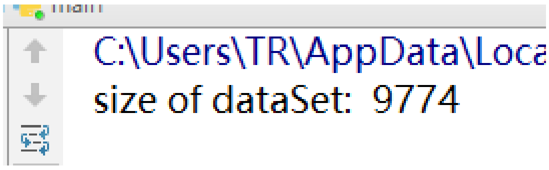
data.py implements a simply function used to load data set for computing and testing, the data set is obtained from the Internet, which is said to be used for machine learning. (however, it seems not to be good from the result)

aprior.py implements the aprior algorithm. firstly find the item set by scanning all trasactions of the whole data set. then use the formula to compute the support of them and find the 1-itemsets of the required frequent item sets. next join the 1-itemsets into 2-itemsets and filliter by delete the itemsets with lower support than needed, at the same time, maintain the support table of all the itemsets. and then so on till there are no more k-itemsets satisfy the minimal support. after that analysis, use the frequent item sets and the support table to generate rules with the formula confidence (A|B) = P(B | A) = P(AUB) / P(A), for each frequent item set, traverse all its subsets to compute the conditional probability, if the result is higher than the minimal confidence, it is a required association rule

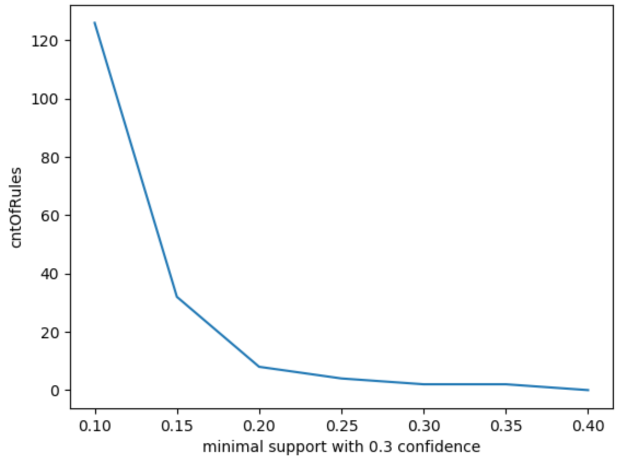
fp.py implements the FP-growth algorithm. firstly scan the data set and derive the set of frequent items and their support counts. then sort them in the order of descending support count. next start from each frequent length-1 pattern, construct the conditional pattern base. Then construct FP-tree, and mining recursively on the tree. after that concatenate the suffix pattern with the generated frequent patterns to construct a new FP-tree to perform the same mining with quite less nodes till there are no nodes to nodes to construct FP-tree

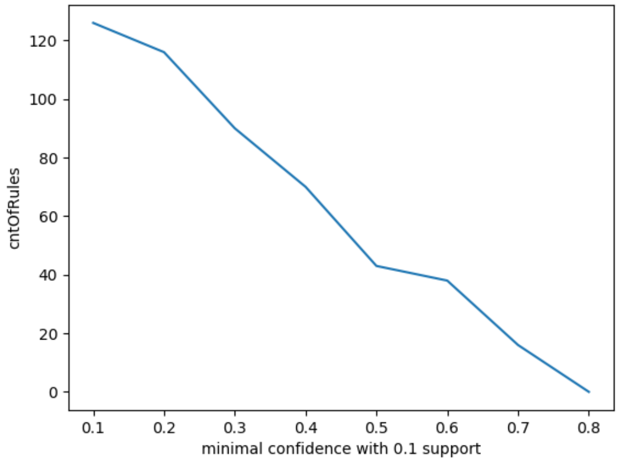
main.py use the other 3 .py files to perform the tasks with the help of numpy and matplotlib.

however, as I mentioned above, my data set is not so good, for which case when the minimal confidence = 0.8, it seems there are no frequent item sets at all so that I cannot finishi the 3) task, an when the minimal support is 0,3, there are at most 2 association rules, which is nonsenseble. anyway, here is the result in the range which makes sense



```
C:\Users\TR\AppData\Loca
size of dataSet:  9774
```





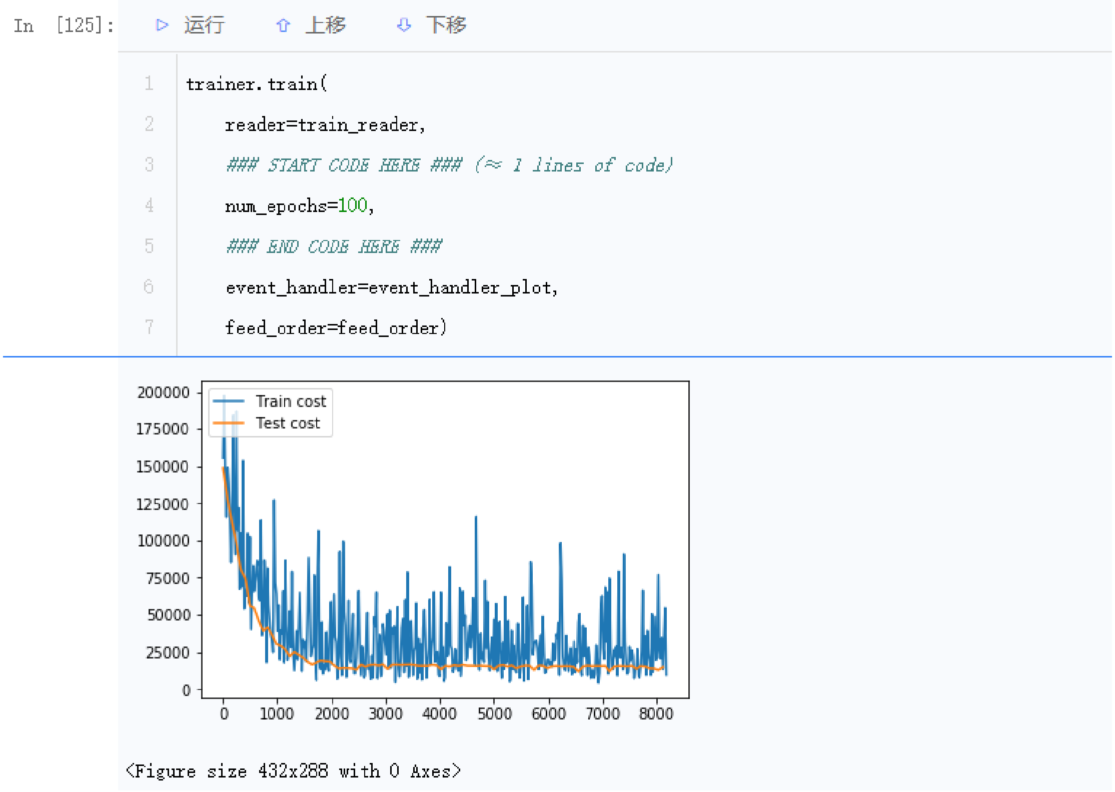
7. 请完成 AI Studio 中房价预测例子，并采集一定数目的真实数据，修改代码，汇报得到的结果

Task is finished using Web AI Studio.

data set is obtained from the CSDN and uploaded to the forked project.

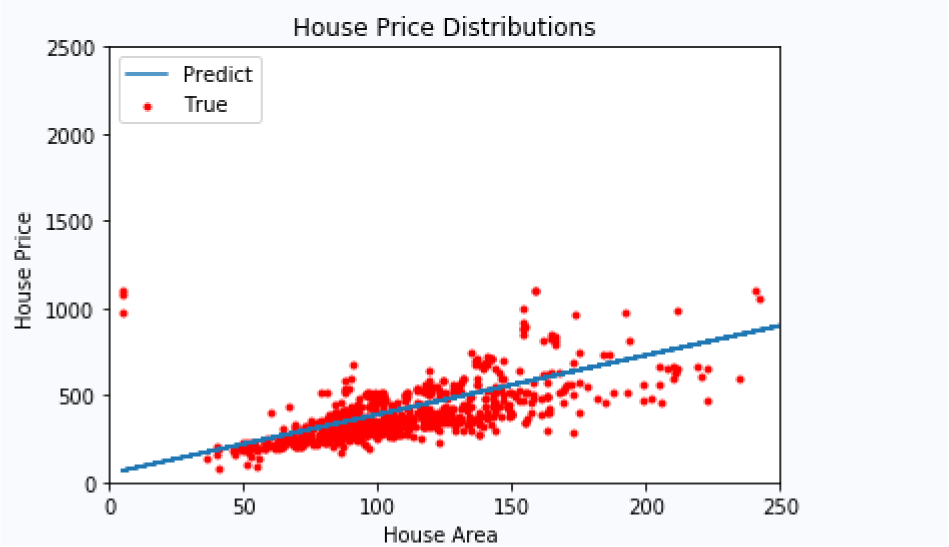
see more screenshots of the process in the **folder 'house'**

here is the result



```
trainer.train(
    reader=train_reader,
    ### START CODE HERE ### (≈ 1 lines of code)
    num_epochs=100,
    ### END CODE HERE ###
    event_handler=event_handler_plot,
    feed_order=feed_order)
```

```
<Figure size 432x288 with 0 Axes>
```



the points of true (area, price) pair are evenly distributed on both sides of the predicted line. but the data set seems to have some noise datas, which are far from the line, such as points around 0, 1000, and that makes the number of points below the line is more than those above