# 数据仓库与数据挖掘 第一次作业

姓名：王宸昊

学号：2019214541

时间：2019.10.19

## 数据预处理

首先，先通过观察，分析几张表当中数据存在的噪音类型，然后通过脚本针对数据中的噪音进行分类，再通过脚本进行处理。在这里通过Python的脚本对数据进行分析和处理，其中利用pandas库对csv文件进行数据分析。

首先8张表中都存在的问题：

1. 数据重复：在每张表中都存在不同程度的数据重复的情况，即在每张表中都存在一些行的数据冗余，在这里处理的方法是pandas中的drop\_duplicates()方法，在导入csv文件后先进行一步去重的工作。
2. 外键依赖：在例如account等表中存在district\_id等字段，该字段对应district表当中的id，所以对于account表中的district\_id都应该对应district真实存在的数据。所以在对每个存在外键依赖的表中，进行检查，如果外键在对应的表中不存在，则直接将这行删除。

其他表中数据存在的问题和预处理方法总结如下：

## 1.1 account:

1. frequency字段不规范：将POPLATEKMESICNE改写为POPLATEK MESICNE。

2. account\_id 字段有错误：account\_id应该是int类型，但是有几条数据有错误，例如1675\_[typo]等形式，解决方案为去掉后面的部分，只保留前面的数字。

3. 将date字段转为date格式。

## 1.2 card:

1. type字段有错误：将拼写为golden修改为gold

2. 将issued字段转为date格式。

## 1.3 client:

1. Birth\_number字段中数据有错误，超过了数据含义的范围：将错误的数据直接删除。

2. 将Birth\_number字段根据含义拆分为birth\_day和gender字段：根据月份+50判定男女。

## 1.4 disp:

1. 外键不存在

## 1.5 district:

## 1.6 loan:

1. duration字段不符合规范：duration的值应该是12的整倍数，但是部分不满足，对于不满足的数据直接删除。

2. payments字段不符合含义：表中的数据应该满足payments\*duration = amount，检查每行数据，不符合的直接删除。

3. 将date字段转为date格式。

## 1.7 order:

1. 外键不存在

## 1.8 trans:

1. 将date字段转为date格式

2. balance字段不规范：balance字段的值应该是数字类型，部分数据非数值，直接删除。

3. Bank字段值为空：部分数据Bank的字段为null,直接删除。

4. 外键不存在

## 数据导入

1. 新建数据库homework，需要注意将不允许修改表结构选项关闭
2. 将CSV当中的数据导入该数据库中，共新建八张表，表的结构如下：

account表：



Card表：



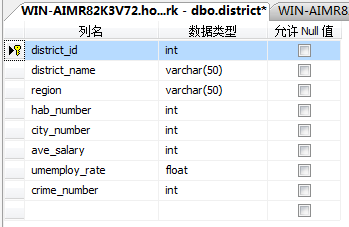
Client表：



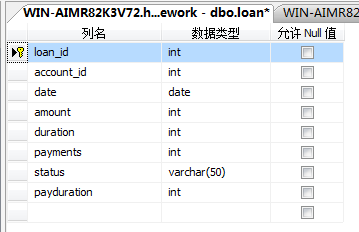
Disp表：



District表：



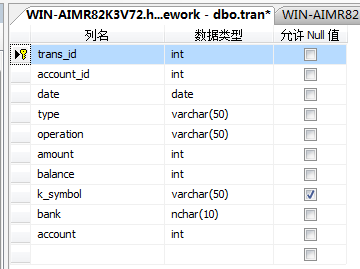
Loan表：



Order表：



Tran表：



## 数据库设计

第一问中需要从不同性别、不同地区、不同信用卡类型进行分析。

第二问中需要从年龄阶段、不同性别、不同地区、交易的类型进行分析。

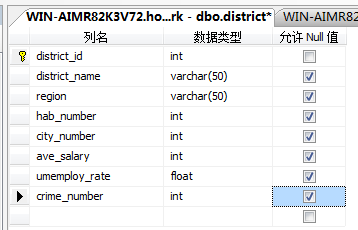
所以首先建立以下维表：

## 3.1 维表:

**性别维度：**从性别维度分析，则性别需要作为一个维度，该有2个属性值，属性值有2个可能：male和female。



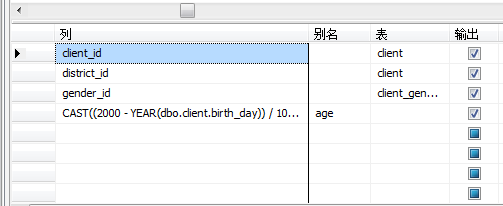
**地区维度：**地区维度可以直接使用原来的district表。



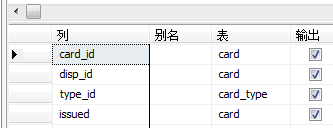
**卡类型维度：**卡类型共有三种值：classic, junior, gold。因此建立card\_type维表



**Client维表：**包含3个属性，其中两个外键连接到district表，一个连接到gender表。这里需要注意的是，需要将连续的年龄的值离散化：其中age字段做一个函数运算，将2000减去出生日期，按照10年为一个阶梯，将不同年龄段的人分开。



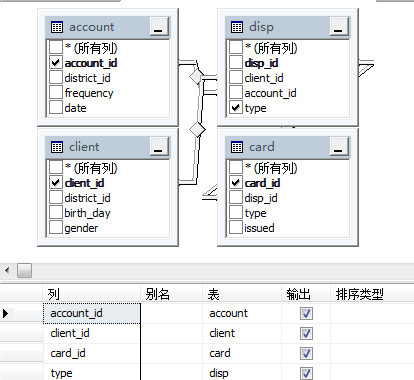
**Card维表：**包含3个属性，两个外键连接到crad表格card\_type表，还有一个issued字段。



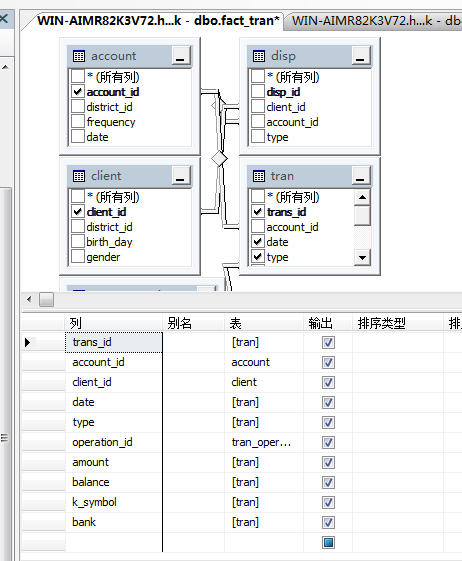
## 3.2 事实表:

**Disp\_fact事实表：**根据disp表当中的数据进行聚合，将用于分析的数据聚合到本表中，其中的数据来源于account,client,card,disp表。

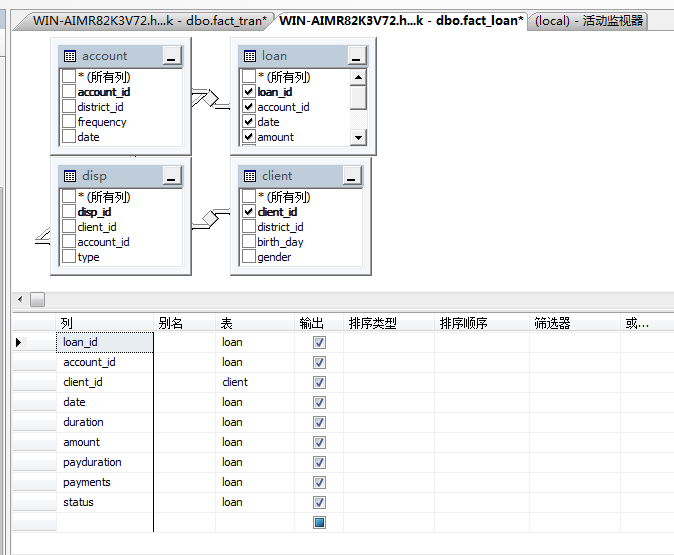
事实表的结构设计如下：



**Tran\_fact事实表：**该表中主要聚合记录交易信息，数据来源于account、disp、client等，进行第二题的分析



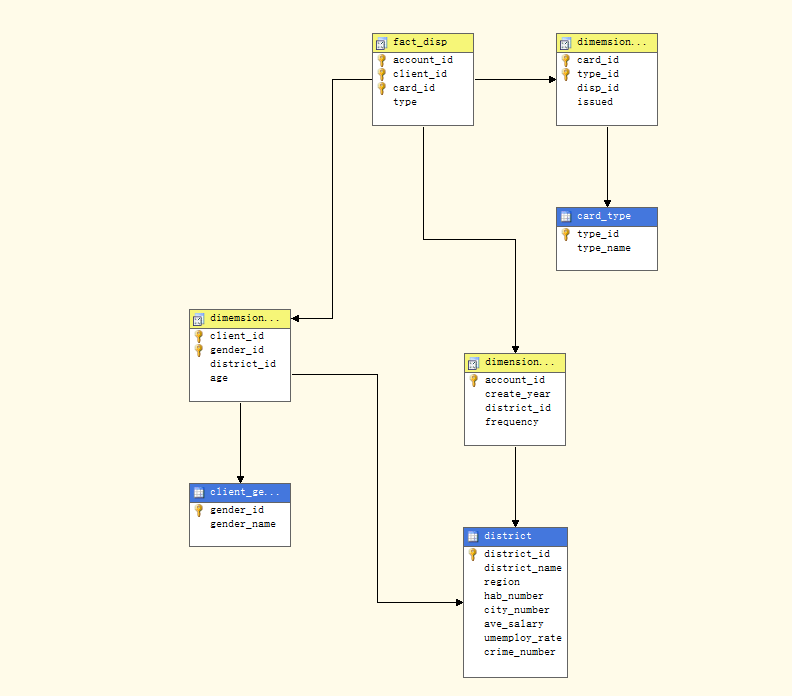
Loan\_Fact事实表：主要记录贷款相关的信息，用于第三问的数据分析，具体的设计如下“

” 

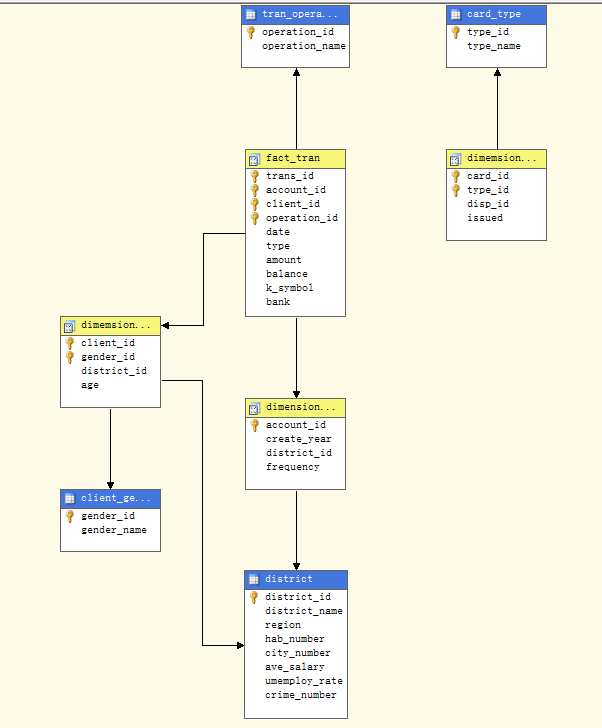
## 数据立方体设计

第一问：第一问中要对用户进行分析，所以使用fact\_disp事实表进行分析，从地区、性别、卡类别、年龄四个维度进行分析。

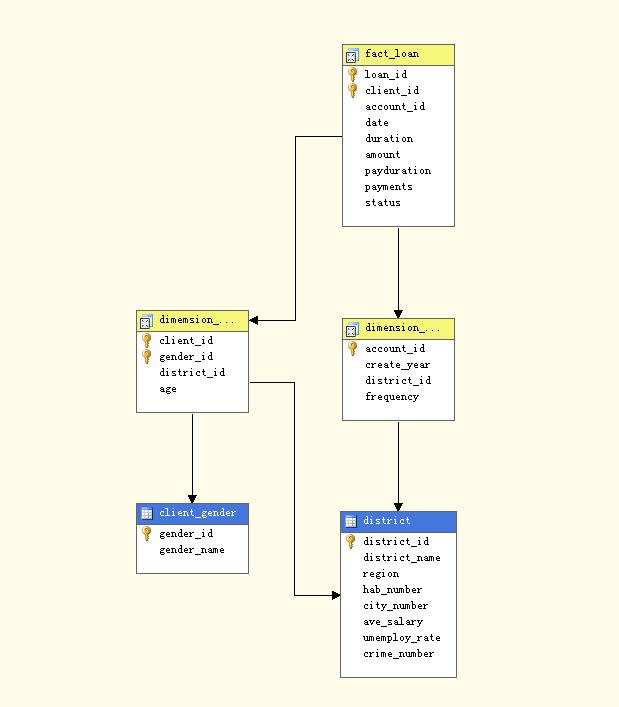
数据仓库的设计如下：



第二问：第二问主要针对trans表当中的数据进行分析，主要对数据当中不同的操作类别进行分析，量度为收入和支出的数目，同时还需要额外进行一次运算，计算出来净支出，所以涉及到的事实表为fact\_tran，维度由4个维表当中的属性所规定，设计如下。



第三问：由于第三问主要是针对贷款的情况进行分析，用于分析的事实表为fact\_loan表，量度为贷款总数，观察的维度包括年龄段、性别、地区、信用卡的类型等多个方面。其他维表的设计如下：



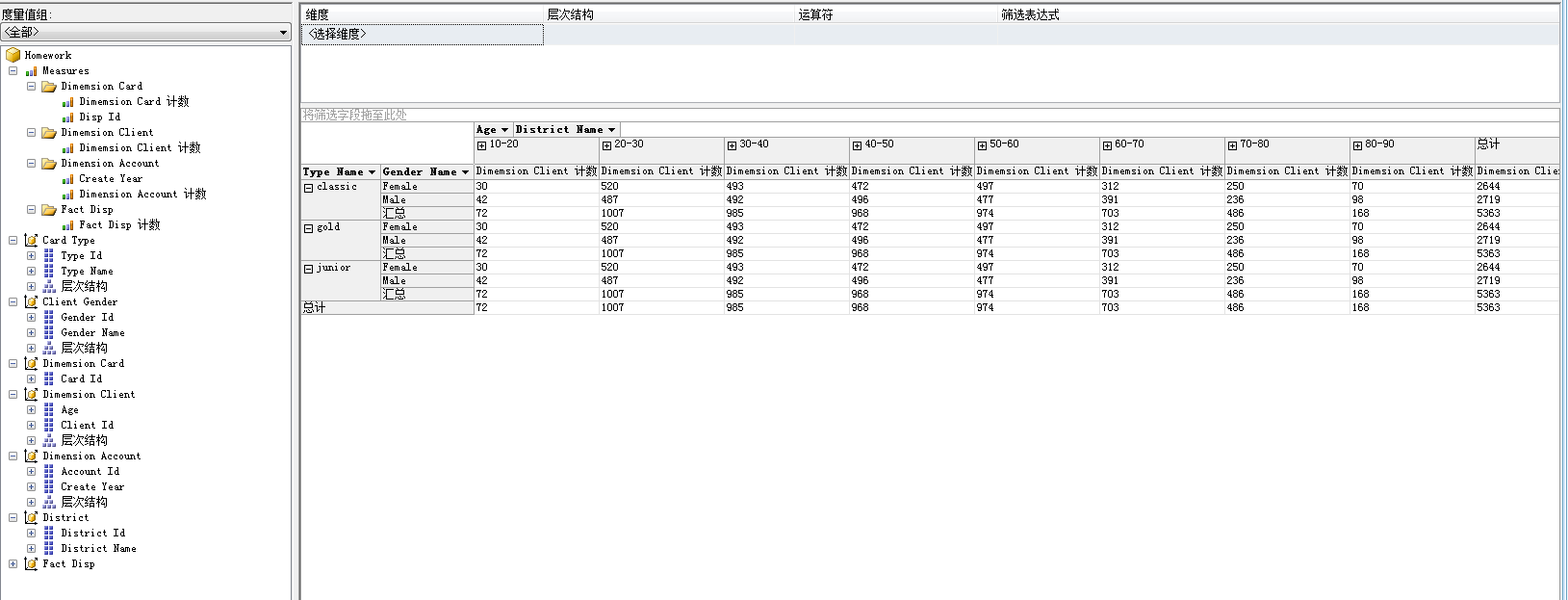
## 实验结果分析

## 5.1 题目一:

问题：针对不同性别，不同地区，不同信用卡类型（card.type）的用户，分析其

中老年（年龄大于 50）用户的数量。

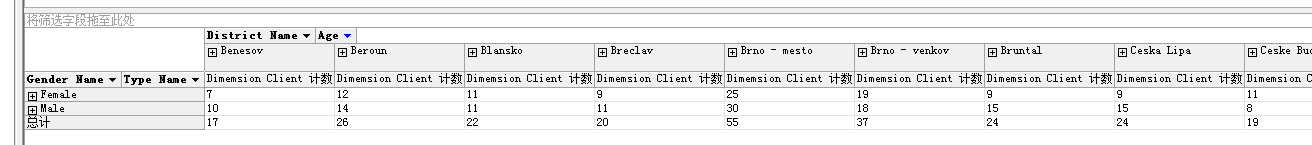
设定维度为Type Name, Gender Name, District Name,,Age。量度为Client的数量。



从表中的数据可以分析得出，从性别上看，50岁以上的男性用户要略多于女性用户。在所有的年龄段中，80-90岁的人用户的人数是最少的，

对于卡的类型同时随着年龄的增长，用户数量越来越少，但是卡的级别越来越高。

针对不同地区，由于地区的名称过多，数据比较分散，不利于分析，但是从整体上看，男性用于要高于女性用户。

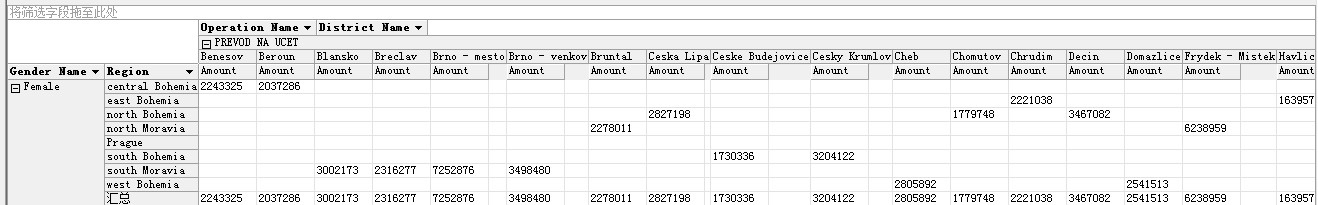


## 5.2 题目二:

问题：针对不同的年龄阶段，不同性别，不同地区的用户，分析交易的收入

（trans.type=PRIJEM）情况，支出情况（trans.type=VYDAJ），净支出（所

有支出减去所有收入）状况。



从上图可以分析，就地区而言south\_Moravia的收入和支出的金额是最多的。

就性别而言，女生的交易额是高于男生的。

但是这里并没有完全达到题目的要求，并没有按照交易的类型，对净支出进行分析。

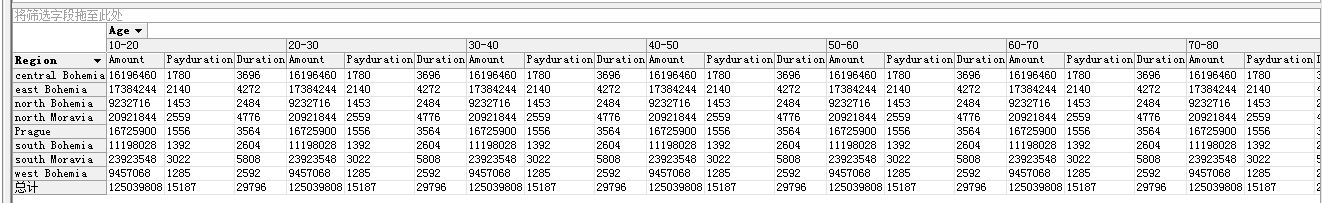
## 5.3 题目三:

问题：假设你是一个银行的数据分析人员，银行希望能够放出更多的贷款。请

以这个作为目标，选择合适的维度进行分析，告知管理人员应该把优惠政策和宣传力度集中到哪类人群。

在数据立方体Homework-3当中，选取了3个量度进行评判放贷的情况，分别是amount贷款总额、duration分期还款数，payduration记录已还款期数。之所以从这三个角度分析，是因为从贷款总额中可以判断出一个人的贷款总的预期，而通过duration和payduration可以估计出某个人的偿还能力，综合以上者三个量度可以更好地判断放贷的情况。

为了展示的更清楚，首先从客户的个人属性：年龄和性别的角度观察，如下图：

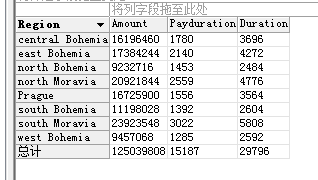


从年龄上看如下图：

大量的贷款的额度的用户主要集中在20-60岁之间，其中数量最大的30-40岁，而且一般这个年龄段的人贷款的周期比较长，偿还能力也比较高，所以将贷款的宣传作用在年龄段在30-50左右的人群是比较合适的。

就性别而言，在30-50这个年龄段的女性的贷款能力明显优于

从地区的角度观察如下图：



可以看到就贷款的总数上来说，north Moravia的总贷款数是最多的，而且已经偿还的比例也相当高，同时贷款的周期还比较短。因此综合以上三个因素，在地域的选择上，在north Moravia大力宣传的重要性是非常高的。

## 实验总结

终于在最后一刻“算是”完成了作业。回顾做作业的过程，感觉时间分配的并不是非常合理，在这里总结和反思一下这个大作业的过程。

首先本次作业我认为可以主要分为三个部分，数据预处理、数据仓库设计、数据分析。在数据预处理的部分，我花费了比较多的时间，通过写python的脚本，对每个异常数据进行检查，在这个过程中也是比较投入的。

但是在后来SQLServer的使用上遇到了比较大的问题，首先是对任务目标的不了解，并不知道建立一个数据仓库需要的基础知识有哪些，PPT上的介绍也比较模糊，并不知道如何下手，而且对维表事实表等概念的理解并不深刻，并不知道设计一个好的数据仓库的结构，所以在最后的设计与实验中并没有完全完成任务，也是希望能在具体的实践操作中能得到多一些的指导，这样做的时候就不会感到迷茫了。

## 文件目录介绍

- SQL脚本

-- create\_dim.sql: 创建维表的SQL脚本

-- create\_fact,sql: 创建事实表的SQL脚本

- 数据分析工程文件：AnalysisService工程文件

- 数据预处理

-- clean\_data: 清洗后的数据

-- dataset: 网络学堂上的数据

-- dirty\_data: 清洗前的数据

-- clean\_data.py: 清洗8张表的Python脚本

-- tools.py: 清洗过程的公用函数

- 实验报告.pdf