天津大学

基于大众点评的天津美食大数据

科技报告

组　　名　智能与计算学部

专　　业　计算机科学与技术

组　　员　迪丽菲娅　3019244005

组　　员　穆耶赛尔　3019244006

组　　员　格桑曲珍　3019244018

组　　员　吴柯睿　3019244365

年　　级　2019 级

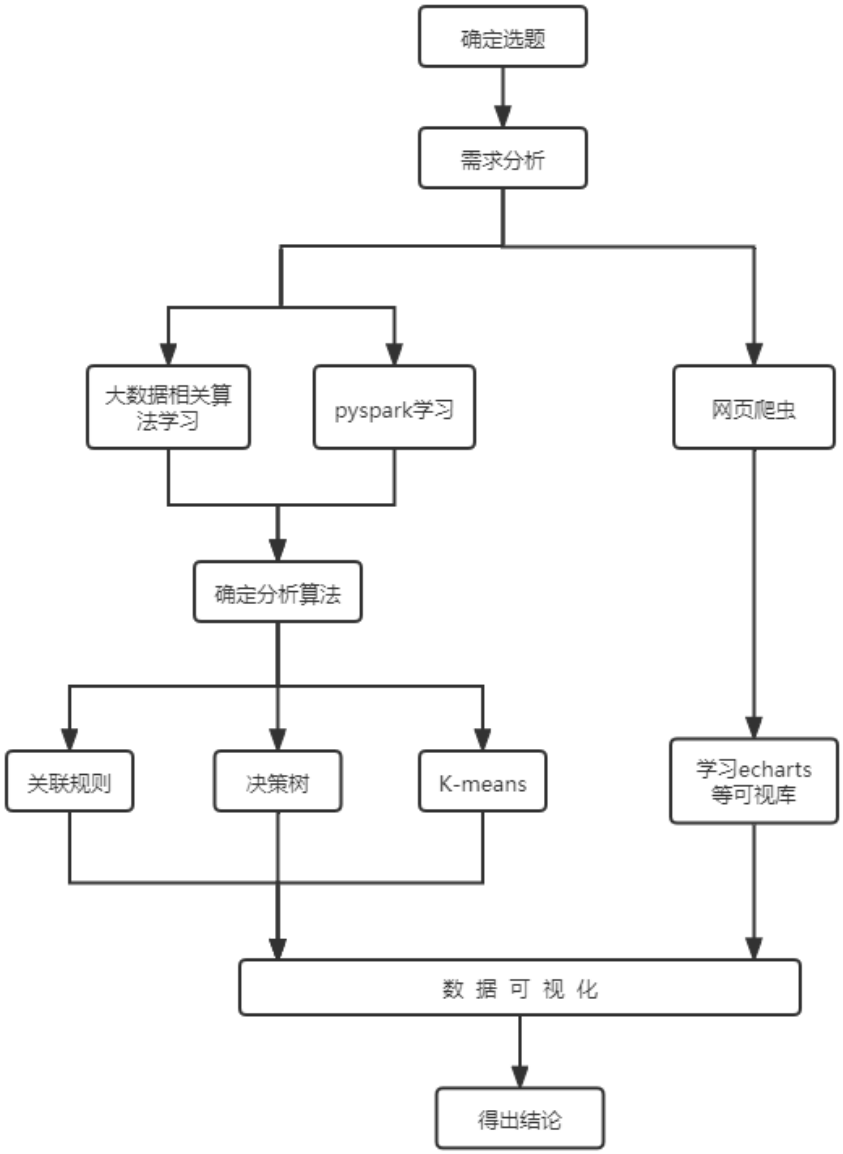
目录

1.课题背景及意义

“大数据”作为当下最流行的词汇，已经成为了科技化，信息化，现代化时代的代名词。“大数据”已经向各行各业渗透，成为了行业发展必不可少缺少的一部分，我们可以几乎肯定的说，没有哪一行能够脱离大数据单独存在。“大数据”改变了每个人看待问题、分析问题、解决问题的方式，为每个领域的发展带来了新的机遇，掀起了各个行业数字化转型的浪潮。餐饮行业作为最能感知经济律动、最快反映消费需求的民生行业，在刚性需求、高消费频次等特征下，成为了发展迅猛的行业之一。“大数据”已经广泛地运用在了餐饮行业并初具成效，已经开启了以“智慧餐饮”为主的智能化餐饮新时代，为餐饮企业提供了营销、运营、管理等工具和服务的系统，提升了餐饮企业的经营效率和消费者消费体验。

餐饮大数据不仅可以帮助餐饮行业精确市场定位通过大数据科学系统的收集、管理和分析数据，并基于大数据数学模型对未来市场进行预测，提出更好的解决问题的方案和建议。还可以帮助餐饮行业需求开发对网上餐饮行业的评论数据进行收集、聚类、情感分析等方法了解消费者的消费行为、价值趋向、新消费需求和企业产品质量问题，以此来改进和创新产品，量化产品价值。此外积累、收集和整理消费者的消费行为方面的信息数据，再对这些收集到的大数据进行处理和分析，以此来了解消费者的消费取向，投消费者所好，利用各种各样的 APP 对用户定向宣传，实现营销效果最大化，获得更多的利润。本次实验中我们小组将通过分析‘大众点评’上的数据集来获得一系列的分析结果从而更好地定位各类餐饮行业。

迪丽菲娅 3019244005

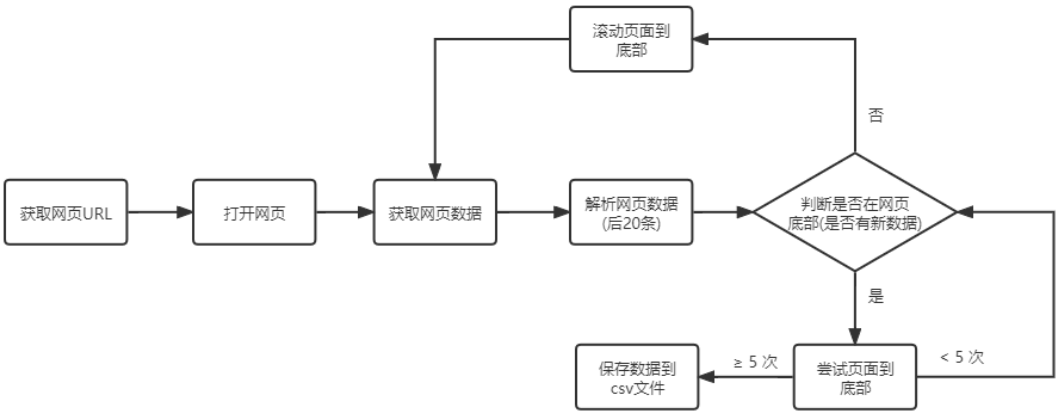


2.技术路线

3.数据采集

迪丽菲娅 3019244005

在本次实验中我们选则了大众点评手机网页版(m.dianping.com/tianjin/ch10) 作为数据源，因为相比于翻页式的电脑网页，瀑布流式的手机网页能够获取更多的数据。我们利用 selenium 库编写了爬虫代码，爬取了天津 16 个区的店铺信息数据 ,保存为单独的 CSV 文件后将这些文件整合为了 All.csv (共计 48390 条数据)。爬虫流程如下：



All.csv 中，每条信息都包含 7 个字段：店铺名称（shop_name）、店铺评分（shop_star）、店铺评论数(shop_comment)、店铺均价（shop_price）、店铺所在商区(shop_region)、店铺菜品类型(shop_category)、店铺所在区(shop_district)。

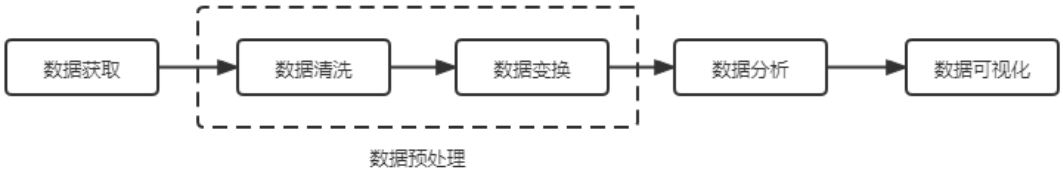
4.数据可视化分析

迪丽菲娅 3019244005

数据可视化是指将数据以视觉形式来呈现、是一种直观形象的数据表达方式。可视化的数据可以帮助人们快速、轻松地提取数据中的含义，了解数据间的关系。用可视化方式，可以充分展示数据的模式，占比，趋势和相关性等许多特征，是大数据分析中必不可少缺少的一环。

4.1 可视化流程

数据可视化的主要任务是实现数据到图表的转换。在本次实验中，数据可视化按照以下流程展开：



1）数据清洗

利用 pyspark 库，主要从以下三个方面对数据进行了清洗：

① 删除了包含 null 字段的数据项。

② 删除了店铺评分(shop_star)和店铺评论数(shop_comment)为 0 的数据项。

③ 删除了店铺菜品种类(shop_category)中包含 ' 更多食品保健品', '保健品', '栗子/干果', '水果店', '水果生鲜', '生鲜' 等字段的数据项，仅保留了与此次分析有关的店铺信息。

2）数据变换

利用 pyspark 库，主要从以下两个方面对数据进行了变换：

① 经过观察后发现，店铺只有为连锁店的时候店铺名称才会重复出现，且店铺

名称的格式为：店铺名称（XXX 分店）。为了更好利用该字段，我们删除了所有店铺名称中包含括号的内容，以便在后续进行对于连锁店铺的数据可视化。

② 为了在分析和可视化过程中得到更为直观的结果，我们对店铺菜品种类（shop_category）字段的数据进行了概化。如，包含“韩式”字符串的字段会被替换为“韩国料理”；“重庆火锅”，“四川火锅”等内容会被替换为“火锅”。

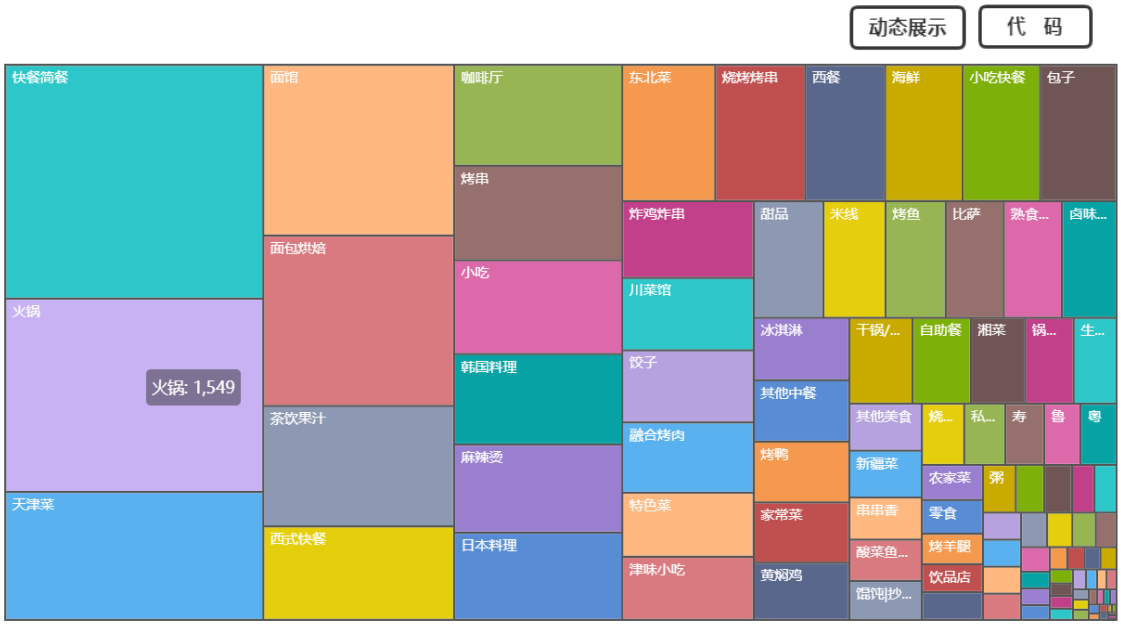
3）数据分析及可视化

利用 pandas 库，对各个字段进行聚类、计数、求和、求平均、排序等操作得到了从不同角度描述天津美食的数据，并用 Python，JavaScript 编程语言，利用 ECharts 和 pyecharts 等数据可视化图表库，对分析后的数据进行了可视化。

4.2 可视化展示

温馨提示：单击“动态展示”可以获得更好的观看体验哟（๑ˉ◡ˉ)/"""

4.2.1 菜品种类

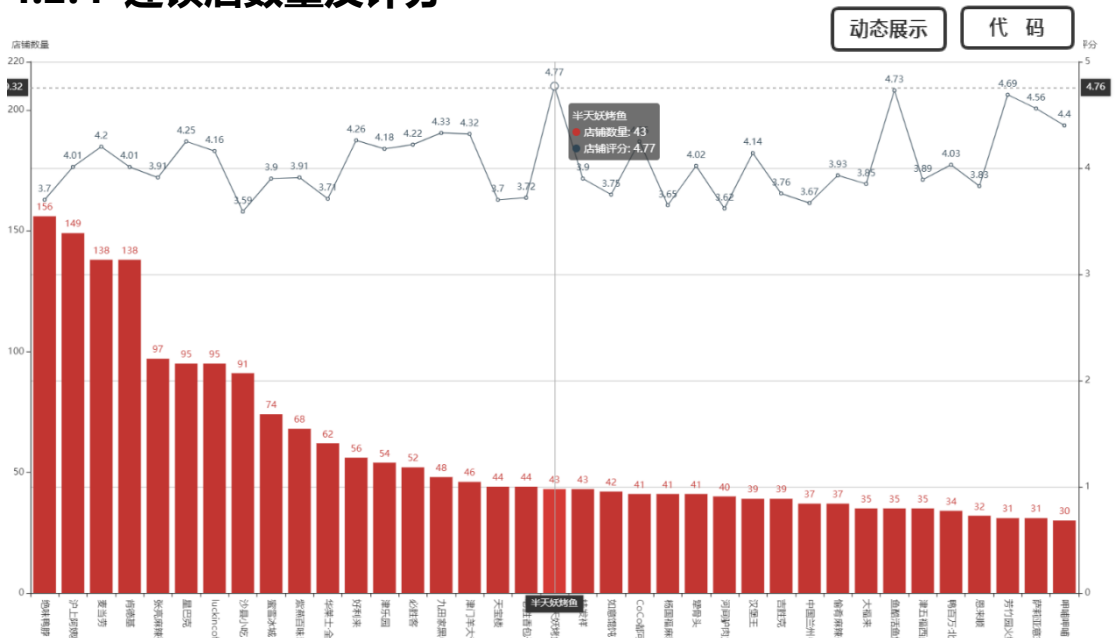


天津数量最多的菜品是“快餐简餐” ，“火锅” 和 “天津菜” 。其中“快餐简餐”普遍为连锁店，成为菜品榜首在意料之中。展现天津特色的“天津菜”也进入了前三；榜单第二名说明，无论在哪个城市“火锅”都十分的受欢迎。

4.2.3 连锁店词云

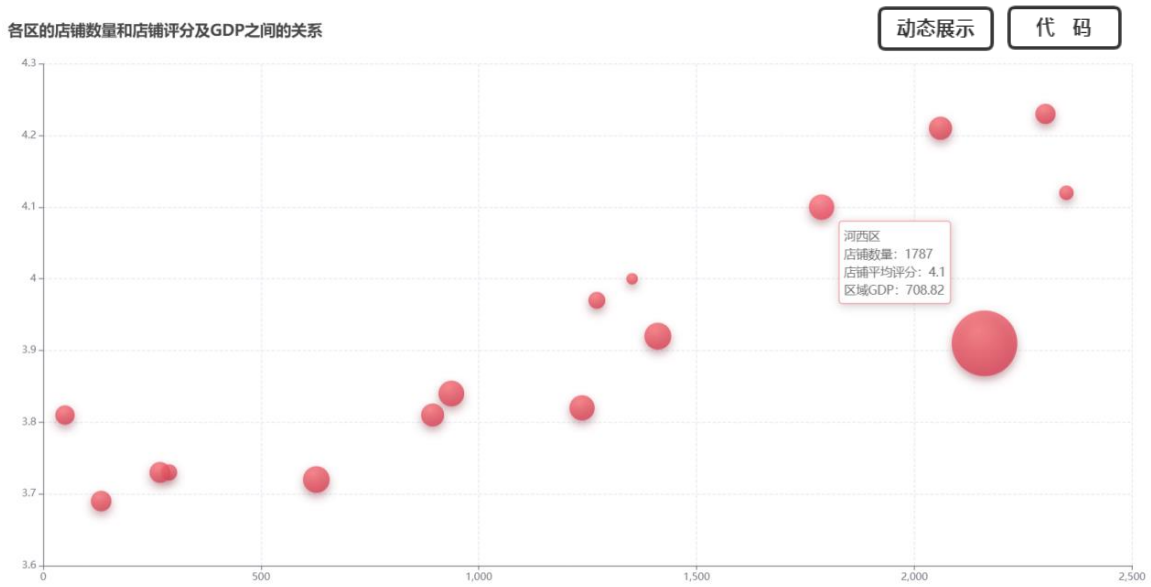
可以看出所有连锁店铺中快餐类和饮品类店铺的规模最大，发展最好。

4.2.4 连锁店数量及评分



图片中横坐标是连锁店铺名称，左侧纵坐标为店铺数量，右侧纵坐标表示店铺评分。从表格中可以看到，连锁店铺的数量与评分之间没有明显的关系；也就是锁，店铺开的越多并不能说明这家店就越好。

4.2.5 各区店铺数量和评分及 GDP 之间的关系



图片中横坐标表示的是店铺数量，纵坐标表示的是店铺评分，每个圆圈代表一个区，且圆圈的大小表示该地区的 GDP。该图标展示了天津各区的店铺数量-店铺平均评分-地区 GDP 之间的关系。

可以从图片中看出，店铺数量与店铺平均评分之间成正比。其中处于右上角（店铺数量多且评分高）的区域，主要包括市中心的地区和 GDP 较高的地区。

5.大数据分析算法

数据挖掘又被称为资料探勘、数据采矿和知识发现(Knowledge Discovery in Database，KDD)。就是从大量的数据中获取有效的、新颖的、潜在有用的、最终可理解的模式的非平凡过程。广义的数据挖掘就是从存放在数据库，数据仓库或其他信息库中的大量的数据中“挖掘”出有趣的知识的过程。知识发现过程由以下步骤组成（1）数据清理（2）数据集成（3）数据选择（4）数据变换（5）数据挖掘（6）模式评估（7）知识表示。

数据挖掘常用的技术有关联规则、回归分析、聚类分析、决策树、随机森林等。本次实验中我们选择了关联规则、决策树及 K-means 算法等 3 种算法对数据进行分析。

5.1 关联规则算法

穆耶赛尔 3019244006

对于本实验，关联规则挖掘的方法可用于发现各种有趣的规则和现象，比如：

（1）利用关联规则来探究菜品（美食种类）与天津各区之间的关系如何？哪一个区有哪种菜品的可能性更高？

（2）或者：对于一个区而言，某种菜品与地点之间的关系。

5.1.1 关联规则挖掘定义

关联规则挖掘可以让我们从数据集中发现项与项（item 与 item）之间的关系，也就是查找项目集合之间的频繁模式、关联、相关性、因果结构等，从而描述一个事件中某些属性同时出现的规律和模式。在本实验中，我们主要探究菜

品（美食种类）与天津各区之间的关系，也就是哪一个区有哪种菜品的可能性更高；对于一个区而言，某种美食种类与地点之间的关系。

5.1.2 关联规则挖掘所要用到的几个基本概念如下

1）事务：每一家店铺称为一个事务，例如和平区中的数据集包含 3280 个事务。

2）项：店铺的每一种属性（星级，地点，菜品等）称为一个项。

3）支持度计数：一个项（项集）出现在几个事务当中，它的支持度计数就是几。例如在下图中有 5 个事务，“天津菜”出现在 1 和 2 号事务中，所以对于这个小数据集而言“天津菜”的支持度计数是 2。

	shop_name	shop_star	shop_comment	shop_price	shop_region	shop_category
2	津门八大碗	4.5	1011	63	小白楼	天津菜
3	陆壹捌餐厅(五大道店)	4.7	5459	134	五大道	天津菜
4	乾宫精致私房菜(五大道店)	4.8	2457	93	五大道	私房菜
5	南洋餐室(开封道店)	4.8	4284	58	小白楼	南洋中菜
6	清鲜境意创意菜(南京路店)	4.9	660	99	海光寺/六里台	创意菜

4）支持度：支持度计数除于总的事务数。例如上例中“天津菜”的支持度计数为 2，所以它的支持度是 2÷5=40%。

5）频繁项集：支持度大于或等于某个阈值的项集就叫做频繁项集。例如阈值设为 30%时，因为“天津菜”的支持度是 40%，所以它是频繁项集。

6）置信度：对于规则“小白楼”→“天津菜”，{小白楼，天津菜}的支持度计数除于{小白楼}的支持度计数，为这个规则的置信度。

7）强关联规则：大于或等于最小支持度阈值和最小置信度阈值的规则叫做强关联规则。关联分析的最终目标就是要找出强关联规则。得到的强关联规则并不一定为真：此时需要计算规则中样本的相关性系数：即提升度。

8）提升度：Lift(A->B) = Confidence (A->B) /support (B)= support(A,B) / (support(A) support(B)) =lift(B->A)=（AB 同时出现的次数）/(包含 A 的事务

数 X 包含 B 的事务数)提升度表示了一条关联规则是否有效 ,如果提升度大于 1，说明规则为有效的强关联规则，提升度小于 1，表示规则为无效的强关联规则。如果提升度为 1 ,则表示两事件相互独立。提升度表示在含有 A 的条件下同时含有 B 的可能性与没有这个条件下项集中含有 B 的可能性之比 ,也可以理解为在 B 自身出现可能性 P（B）的基础上，A 的出现对于 B 的“出镜率”的提升程度。

该指标与置信度同样用于衡量规则的可靠性，可以看做是置信度的一种互补指标。

5.1.3 模型建立及结果分析

1）以和平区为例选择数据集 01_test（由美食种类和和平区内对应的地点构成的数据集）建立该区内地点与餐饮类别的关联分析。选择支持度的最小阈值为 0.01，置信度最小阈值为 0.2。

（1）利用 FP_tree 算法得到了所有频繁项集并计算得到所有的置信度：

```
["('滨江道',)->('面馆',)", 33, 0.2, 1.0525641025641026]
["('面馆',)->('滨江道',)", 33, 0.052884615384615384, 1.0525641025641026]
["('滨江道',)->('茶饮果汁',)", 55, 0.3548387096774194, 1.8674524400330852]
["('茶饮果汁',)->('滨江道',)", 55, 0.08814102564102565, 1.8674524400330852]
["('五大道',)->('西餐',)", 34, 0.34, 2.3506526315789475]
["('西餐',)->('五大道',)", 34, 0.07157894736842105, 2.3506526315789475]
["('和平路',)->('快餐简餐',)", 85, 0.1756198347107438, 1.174614128696706]
["('快餐简餐',)->('和平路',)", 85, 0.17311608961303462, 1.174614128696706]
["('海光寺/六里台',)->('快餐简餐',)", 97, 0.20041322314049587, 1.2656865861411315]
["('快餐简餐',)->('海光寺/六里台',)", 97, 0.18653846153846154, 1.2656865861411315]
["('滨江道',)->('快餐简餐',)", 59, 0.12190082644628099, 0.6415421699512609]
["('快餐简餐',)->('滨江道',)", 59, 0.09455128205128205, 0.6415421699512609]
["('快餐简餐',)->('五大道',)", 46, 0.0968421052631579, 0.6570856894301871]
["('五大道',)->('快餐简餐',)", 46, 0.09504132231404959, 0.6570856894301871]
["('快餐简餐',)->('南市',)", 55, 0.17027863777089783, 1.1553616661975794]
["('南市',)->('快餐简餐',)", 55, 0.11363636363636363, 1.1553616661975794]
["('五大道',)->('日本料理',)", 33, 0.39285714285714285, 2.7160902255639097]
["('日本料理',)->('五大道',)", 33, 0.06947368421052631, 2.7160902255639097]
["('快餐简餐',)->('小白楼',)", 50, 0.18450184501845018, 1.2518678905797322]
["('小白楼',)->('快餐简餐',)", 50, 0.10330578512396695, 1.2518678905797322]
["('快餐简餐',)->('鞍山道沿线',)", 43, 0.17338709677419356, 1.1764529458810984]
["('鞍山道沿线',)->('快餐简餐',)", 43, 0.08884297520661157, 1.1764529458810984]
["('五大道',)->('咖啡厅',)", 69, 0.32242990654205606, 2.229178553861289]
["('咖啡厅',)->('五大道',)", 69, 0.14526315789473684, 2.229178553861289]
["('滨江道',)->('咖啡厅',)", 47, 0.21962616822429906, 1.1558531032830097]
["('咖啡厅',)->('滨江道',)", 47, 0.07532051282051282, 1.1558531032830097]
计算完成置信度 26
```

（2）从其中筛选出置信度大于 0.2，提升度大于 1 的规则得到：

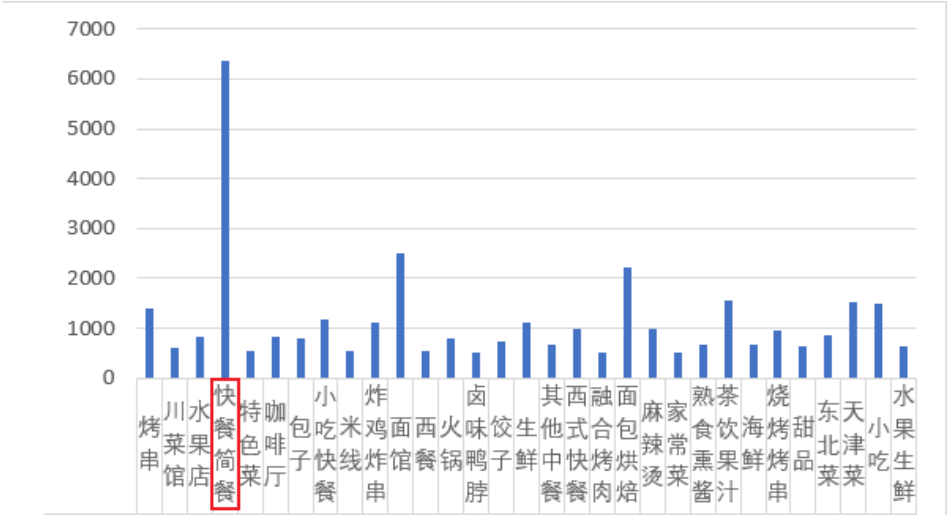
```
["('滨江道',)->('茶饮果汁',)", 55, 0.3548387096774194, 1.8674524400330852]
["('五大道',)->('西餐',)", 34, 0.34, 2.3506526315789475]
["('海光寺/六里台',)->('快餐简餐',)", 97, 0.20041322314049587, 1.2656865861411315]
["('五大道',)->('日本料理',)", 33, 0.39285714285714285, 2.7160902255639097]
["('五大道',)->('咖啡厅',)", 69, 0.32242990654205606, 2.229178553861289]
["('滨江道',)->('咖啡厅',)", 47, 0.21962616822429906, 1.1558531032830097]
```

（3）当再从（2）的结果中筛选置信度大于 0.35 的规则，得到：

```
["('滨江道',)->('茶饮果汁',)", 55, 0.3548387096774194, 1.8674524400330852]
["('五大道',)->('日本料理',)", 33, 0.39285714285714285, 2.7160902255639097]
```

滨江道步行街作为中国最长的步行街，这里的美食商铺是丰富多彩的，“茶饮果汁”类商家喜欢选择这里也是符合步行街特色的。因此得到的两个关联规则是符合常理的。

2）当挖掘数据集 02_test（由美食种类和对应的区名构成的数据集）中的所有频繁项集时得到下图所示的图片，可知所有美食种类商家中“快餐简餐”是最多的，因此我们将分析“快餐简餐”与各区之间的关系。



从中选择“快餐简餐”利用关联规则来探究其与天津各区之间的关系。置信度最小阈值为 0.15，选择提升度大于 1 的规则。得到：

```
["('快餐简餐',)->('津南区',)", 747, 0.16263879817112997, 1.2354676302516183]
["('快餐简餐',)->('北辰区',)", 506, 0.1553100061387354, 1.1797952726918788]
```

该关联规则又能给我们不一样的启发和感受。我们认为津南区和北辰区与“快餐简餐”的关联性比较大并不是因为商家喜欢选择这两个区，而是因为首先“快餐简餐”在各个区分布的比较均匀并且数量多，其次津南区和北辰区美食种类较单一，因此显得“快餐简餐”在该区的占比高。

5.2 决策树算法

吴柯睿 3019244365

决策树是最经典的数据挖掘方法之一，它以树形结构将决策或分类过程展现出来，简单直观、解读性强。下面我们根据大众点评爬取下来的数据中存在的因子变量的值，即是否为连锁店、评论数、价格高低来预测并分类，构造出一个相适应的模型，预测判断店铺的类别。

5.2.1 数据预处理

1. 首先我们将对店铺名（shop_name）、店铺评分（shop_star）、店铺评论数（shop_comment）以及人均消费价格（shop_price）这四个变量进行数据处理。具体如下：

① 对店铺评分（shop_star）该数据进行处理。如果该店铺的评分大于 4，说明该餐厅受顾客欢迎，为受欢迎店铺类。生成的新变量店铺类别取值为 1；反之，该店铺的评分低于 4，消费过的顾客的推荐率不高，我们认为这些餐厅不值得选择，生成新变量店铺类别的取值为 0。新变量列 shop_evaluation 作为该店铺值

不值得推荐的重要指标。

② 对店铺名（shop_name）该数据进行预处理。店铺名（shop_name）如下

shop_name
津门八大碗
陆壹捌餐厅(五大道店)
乾宫精致私房菜(五大道店)

图：

假如 shop_name 内容中存在（ ）标注为什么店，表明该店铺是连锁店。生成一个新列 if_multiple（是否是连锁店），是连锁店则生成的新变量店铺类别取值为 1；反之，取值为 0。

③ 对人均消费价格（shop_price）、店铺评论数（shop_comment）进行处理，将其转换成 double 类型。

2. 然后再对 shop_evaluation 变量建立决策树。处理整理后的数据 :if_multiple、shop_price、shop_comment。如图 1：

shop_evaluation	count
0.0	12100
1.0	7493

评分大于 4 的店铺大约占比 38% ,证明店铺评分高于 4 的店铺还是占少数的。如图 2：

```
# data splitting
(training_df, test_df) = model_df.randomSplit([0.7, 0.3])
```

① 将原数据进行分层抽样 ,70%的数据组成训练集 ,剩 30%的数据组成测试集。

② 测试集的数据量为 250，训练集的数据量为 750。

③ 训练集的数据用于建立模型，之后我们用构建的分类树对测试集中的

df_predictions 变量进行预测并对预测结果进行评价。

5.2.2 决策树建立

我根据大众点评上爬取下来的 if_multiple、shop_price、shop_comment 来分析一个店铺评分是否与这三个变量因素存在关系。因此我们将通过分析数据中的预测因子变量，拟合新变量来建立决策树，分析这三个变量对店铺好坏的关系与其影响程度。如图：

```
# transformer
assembler = VectorAssembler(inputCols=['if_multiple','shop_price','shop_comment'],
                            outputCol='features')
output = assembler.transform(df)

model_df = output.select('features','shop_evaluation')
```

建立的模型如下：

·formula_dazhong = shop_evaluation ~ if_multiple + shop_price + shop_comment

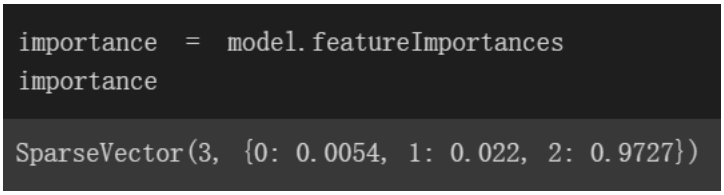
```
# train our model using training data
df_classifier = DecisionTreeClassifier(labelCol='shop_evaluation',featuresCol='features')
model = df_classifier.fit(training_df)

# test our model and make predictions using testing data
df_predictions = model.transform(test_df)
```

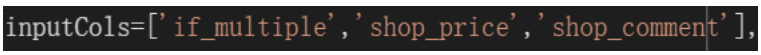
5.2.3 结果分析与结论

我们先用训练集的数据（training_df）建立了决策树，之后我们用构建的决策树对测试集（test_df）中的(df_predictions)变量进行预测并对预测结果进行评价。**1. 结果评估如下所示：**

① 由上图可以看出 df_precision 为 0.8581652869838565。该建立的二分类决策树预测的精确度高于 0.85.

② 由上图可以看出 df_accuracy 为 0.8581652869838565。该建立的二分类决策树预测的准确度高于 0.85.





③分析三个因素影响重要程度，如上图分析可知：

预测因子（if_multiple） < 预测因子（shop_price） < 预测因子（shop_comment）

2. 结论：

我们发现店铺评论数与一个店铺在大众点评上的评价有很大的关系 ;店铺人均消费价格对该店铺评分高低影响很小 ,而是否是连锁店对该店铺评分高低几乎没有影响。

据此分析加上我个人推测，当评论数很多的情况时，至少说明该店铺的顾客量与销量很大，往往这样的店铺都是大众点评上比较受欢迎的店铺。而在大众点评上评论数较少甚至极少的店铺：第一说明该店铺不是很受欢迎，第二评论数本来

就偏少，少量的差评直接拉低的整体评分。

而是不是连锁店与价格高低这两个因素与顾客去给大众点评评分的高低确实没有太大影响。

5.3 K-means 算法

格桑曲珍 3019244018

聚类（clustering）是无监督学习（unsuperviserd learning）中研究和应用最多的一类学习算法，目的是将样本划分成若干个“簇”（cluster），每个“簇”之间尽量相异，每个簇之内的样本尽量相似。K-Means 假设聚类结构能够通过一组原型（点）刻画，K-Means 中的原型是指每个“簇”的质心，这个原型可以使得“簇”内的平方误差的加和达到最小。

5.3.1 实验背景与问题提出

K-Means 算法的目标是最小化聚类后“类”内的平方误差和，因为 E 值越小，“类”内的样本相似性越高。但是这个最小化过程是 NP 难问题，没有有效的算法，只能遍历所有可能的组合（“类”），当样本量较大时就是非常浪费资源甚至无解的。所以考虑使用贪心算法，即通过迭代优化来求近似解。

我们将针对 3 个属性分析三个属性 shop_star shop_comment shop_prize 之间随着价格、评论数以及店铺评分的不同所有店铺聚类的分布图。

shop_star	shop_comment	shop_price
4.5	1011.0	63.0
4.7	5459.0	134.0
4.8	2457.0	93.0
4.8	4284.0	58.0
4.9	660.0	99.0
4.9	4645.0	131.0
4.7	2378.0	77.0
4.4	814.0	61.0
4.9	3550.0	109.0
4.3	89.0	98.0

5.3.2 k-means 聚类算法

k-means 算法接受参数 k ；然后将事先输入的 n 个数据对象划分为 k 个聚类以便使得所获得的聚类满足：同一聚类中的对象相似度较高；而不同聚类中的对象相似度较小。聚类相似度是利用各聚类中对象的均值所获得一个“中心对象”（引力中心）来进行计算的。

K-means 算法是最为经典的基于划分的聚类方法，是十大经典数据挖掘算法之一。K-means 算法的基本思想是：以空间中 k 个点为中心进行聚类，对最靠近他们的对象归类。通过迭代的方法，逐次更新各聚类中心的值，直至得到最好的聚类结果。

1）算法思路

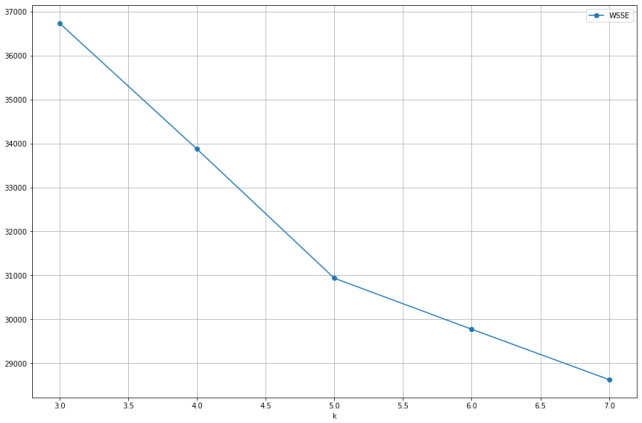
首先从 n 个数据对象任意选择 k 个对象作为初始聚类中心；而对于所剩下其它对象，则根据它们与这些聚类中心的相似度（距离），分别将它们分配给与其最相似的（聚类中心所代表的）聚类；然 后再计算每个所获新聚类的聚类中心（该聚类中所有对象的均值）；不断重复这一过程直到标准测度函数开始收敛为止。一般都采用均方差作为标准测度函数. k 个聚类具有以下特点：各聚类本身尽可能的紧凑，而各聚类之间尽可能的分开。

该算法的最大优势在于简洁和快速。算法的关键在于初始中心的选择和距离公式。

2）聚类数 K 值选取

通过肘部法取得 k 值，肘部法的计算原理是成本函数，成本函数是类别畸变

程度之和，每个类的畸变程度等于每个变量点到其类别中心的位置距离平方和（类内部的成员彼此越紧凑则类的畸变程度越小，越分散越大）。在选择类别数量上，肘部法则会把不同值的成本函数值画出来。随着值的增大，每个类包含的样本数会减少，于是样本离其重心会更近平均畸变程度会减小。随着值继续增大，



平均畸变程度的改善效果会不断减低。值增大过程中，畸变程度的改善效果下降幅度最大的位置对应的值就是肘部。

3）算法步骤

输入：样本集

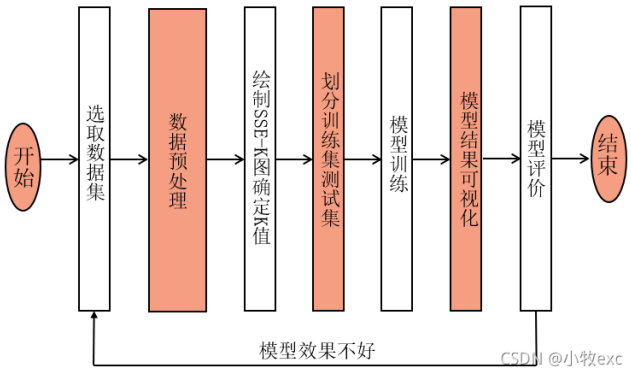
step1: 选择 K 个点作为初始原型；

step2: 计算剩余的点到 K 个点的距离，将点划入距离最近的原型所在的“簇”内；

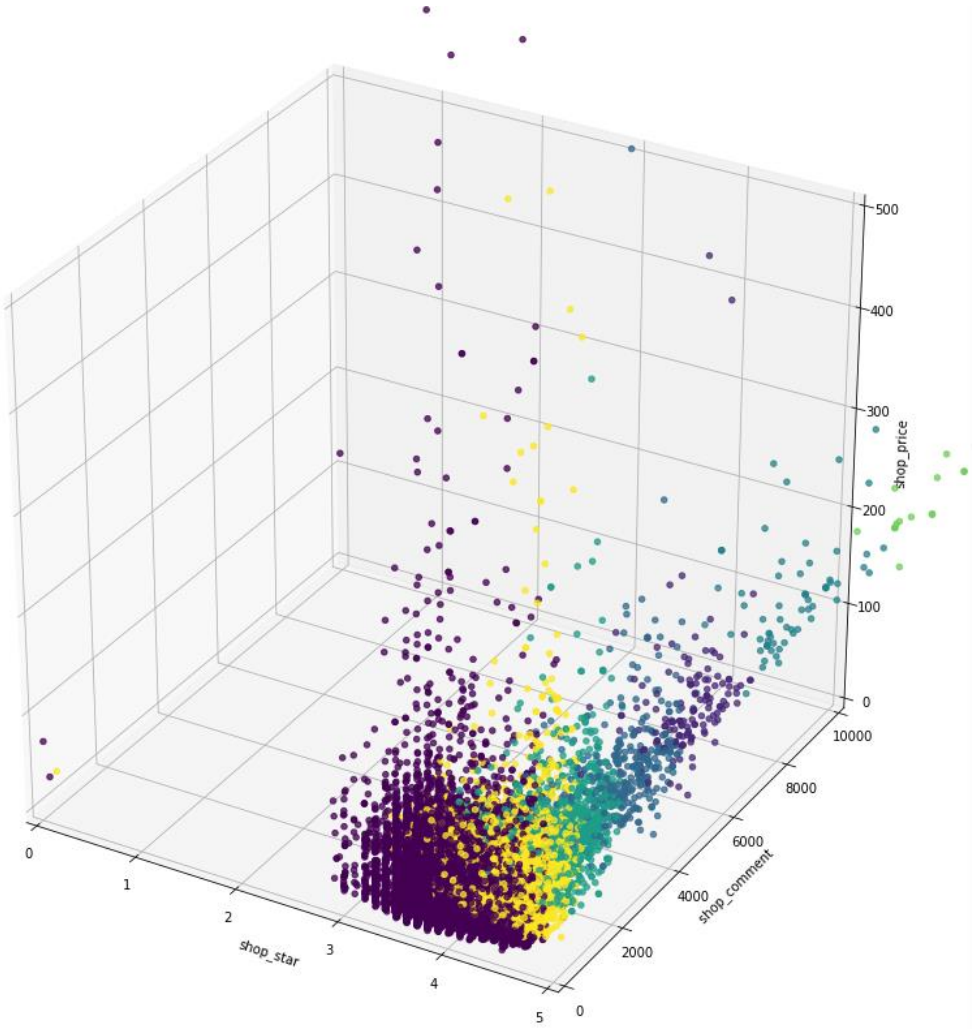
step3: 计算每个“簇”点的均值，并将每个“簇”的均值更新为新的原型；

step4: 重复前面 3 步，直到每个“簇”点的均值不再变化；

step5: 输出每个"簇"的中心。



5.3.3 实验结果及总结



① 不适合圆环形状的簇、不适合交错分布没有明显界限的簇、不适合大小差别

较大的簇

② 初始值的选取会影响最终聚类效果。

③ 聚类数量 K 很重要，K 值可以通过绘制 SSE 图观察肘点得到。在绘制 SSE 图时不对数据归一化处理会比归一化处理后的肘点更加明显。

④ K-Means 不适用于类似圆形的数据集，聚类效果很差，主要是算法的原因。

⑤ 要想得到好的结果，选择合适的数据集至关重要。

⑥ 在选取数据时先查看属性之间的相关程度。

⑦ K-Means 是用来做无标签学习的，它能从数据中提取信息然后按照最小化来对数据聚类，这种原理不一定是贴合实际的，因为事物分类的情况有很多种，并不是简单的距离越小就是一个类别，因此将 K-Means 聚类标签和原始标签去对比在一定程度上重合度不会特别高。

⑧ 在没有标签的情况下可以用 K-Means 对数据进行标记。

6.结语

在整个项目与实验过程中，前期的环境搭建过程花了一定时间，然而在不断的 debug 当中参考了很多网上的资料，可以说是既是帮了很大的忙，又是挖了很大的坑，但总体来说也收获颇多 。

小组内分工明确互帮互助，各司其职在整个小组的共同努力下也按时、按量、安质完成了任务。同时本次项目让我充分懂得如何将理论知识与实际运用相结合，再先学习理论知识与例子后，明确选题与研究方向方法，再进一步学习相关实践运用知识。比如如何根据算法需要对数据进行相应预处理，如何根据自己想分析发现的东西灵活运用算法。将学到的理论知识应用在实际中，解决实际问题，也更深刻地体会到大数据技术在生活中方方面面的应用。

这次大数据大作业对我更好地理解知识、提高对大数据的兴趣有很大的促进作用，我们深切感受到了大数据的魅力。也进一步提高了我的理解与实践能力。

学号	姓名	任务
3019244005	迪丽菲娅	爬虫代码编写，数据可视化，报告撰写
3019244006	穆耶赛尔	关联规则算法，报告撰写
3019244018	格桑曲珍	K-means 算法，报告撰写
3019244365	吴柯睿	决策树算法，报告撰写

7.任务分工