【61】

上节视频介绍了数据预处理的数据归约，这节视频介绍预处理最后一个任务数据变换以及在数据预处理工作中，有哪里需要注意事项。最后将给大家介绍下数据预处理的工具。

【62】

【62.1】如果原始数据的形式不适合信息处理算法的需要，就要进行数据变换。数据变换的目的是将数据转换或统一成**易于进行数据挖掘**的数据存储形式，使得挖掘过程可能更有效。

【62.2】数据变换的方法有很多种，包括光滑、属性构造、聚集、数据规范化、数据离散化、概念分层等。

光滑就是去掉数据中的噪声，前面已经介绍。

属性构造是由给定的属性构造新的属性并添加到属性集中，帮助数据分析和挖掘

聚集是对数据进行汇总和聚集。例如，可以聚集日销售数据，计算月和年销售量。

数据规范化是把属性数据按比例缩放，使之落入一个特定的小区间，比如将企业员工工资[200-20000]缩放到[0.0-1.0]之间

数据离散化，是将数值属性用区间标签或概念标签替换。例如年龄的原始值用【0-10】，【11-20】区间标签或youth，adult 、senior这种概念标签来替换。

由标称数据产生概念分层是指将一些属性向上泛化到较高的概念层，比如street街道的属性可以向上泛化到城市或国家。

在数据变换的这些方法中，光滑、属性构造、聚集、离散化、前面视频已做介绍。接下我们将介绍规范化。

。

【63】

我们知道，规范化是将数据按比例进行缩放，使之落入一个特定的区域。那么，有什么算法来实现数据的规范化呢？

常用的算法有最大-最小规范化、零-均值规范化和小数定标规范化。

【64】

首先最小最大规范化算法，它的思想是，假定minA和maxA分别为属性A的最小和最大值，通过这个公式将属性A的取值映射到一个小的特定区间[new\_min,new\_max]中。举例说明，假定属性income的最小与最大值分别为$12000和$98000，缩放后的特定区间为[0,1]。则当属性income取值为73600时，根据公司可计算得到0.716（绿色为规范化前的区间，红色为规范化后的区间，蓝色是将属性值规范化）

【65】

第二个常用的算法是零均值规范化也叫z-score规范化，它的原理是将属性A的值根据其平均值和标准差进行规范化，这种算法常用于属性的最大值与最小值未知，或使用最小最大规范化方法会出现异常数据的情况。

还是通过举例来介绍零均值规范化的过程。根据数值income所有的取值计算出平均值和标准差，比如平均值为54000，标准差为16000，现在要将income值为73600的数值进行规范化，则将这些数据代入公式可得规范化后的数值为1.225(绿色为平均值，黄色为标准差)。

【66】

还有一个规范化的算法是效数定标算法，它是通过移动属性A的小数点位置进行规范化的，小数点的移动依赖于A的最大绝对值。比如属性A的取值范围为[-986, 917]，A的最大.绝对值是986，小数点需要向左移动3位才能使986变成0.986，所以j取值为3，那么当属性A取值为73时，经过小数定标规范化后的取值为0.073

【67】

数据变换和数据归约所用的方法大多是相同的。有时候、数据归约也等同于数据变换。例如，冗余数据的删除既是一种数据清理形式，也是一种数据归约。

在本节中，重点介绍了最小-最大规范化，零-均值规范化，小数定标规范化。

【68】

【68.1】我们已经了解数据预处理的四大处理流程步骤：数据清理，采集，归约，变换。这四个流程在实际工作中并不是完全分开的，在某种场景下是可以一起使用的。例如，数据清理可能涉及纠正错误数据的变换，如把一个数据字段的所有项都变换成统一的格式，然后进行数据清理；冗余数据的删除既是一种数据清理形式，也是一种数据归约。

【68.2】另外，我们也需注意，在实际工作中，应该针对具体所要研究的问题通过详细分析后再进行预处理方案的选择，毕竟，方案的选择的优劣决定了整个预处理的数据质量效果。

【68.3】此外，整个预处理过程要尽量人机结合，尤其要注重和客户以及专家多交流。因为，预处理是一个广泛的领域，数据可能涉及到医疗、互联网、科研等等应用领域。和相关客户及专家多多交流，能使得我们少走弯路。

【68.4】数据预处理完毕后，若挖掘结果显示和实际差异较大，在排除源数据的问题后，则有必要考虑数据的二次预处理，以修正初次数据预处理中引入的误差或方法的不当，若二次挖掘结果仍然异常，则需要另行斟酌以实现达到较好的挖掘效果。

【69】

介绍完毕数据处理的流程步骤后，我们将介绍到数据预处理的工具。一个称心如意的工具在手，处理起数据来当然是事半功倍。据预处理的工具及手段都是多样化的。但总体归纳起来，可以分为工具类手段及编程类手段。在这里，针对这两类手段，我们推荐Kettle工具和Python语言进行数据预处理。

【70】

【70.1】Kettle是一款开源的软件工具，可以为企业提供灵活的数据抽取和数据处理的功能。

【70.2】Ketlle除了支持各种关系型数据库、 HBase、 MongoDB这样的 NoSQL数据源外，它还支持 Excel，Access这类小型的数据源。并且通过插件扩展， Kettle可以支持各类数据源。本书详细介绍了Kettle可以处理的数据源，而且详细介绍了如何使用Kettle抽取增量数据。

【70.3】Kettle的数据处理功能也很强大，除了选择、过滤、分组、连接、排序这些常用的功能外， Kettle里的Java表达式、正则表达式、Java脚本、Java类等功能都非常灵活而强大，都非常适合于各种数据处理功能。

【71】

【71.1】另外，我们选择Python作为本教材数据预处理的工具，最主要的原因是随着人工智能疯狂的浪潮，新生代的工具Python得到了前所未有的爆发。也是极其适合初学者入门的编程语言，同时它又是万能的胶水语言，可以胜任很多领域的工作，是人工智能和大数据时代的明星，可以说是未来学习编程的首选语言。

【71.2】Python，是一种面向对象的解释型计算机程序设计语言，具有丰富和强大的库，Python已经成为继JAVA，C++之后的的第三大语言。它具备有简单易学、免费开源、高层语言、可移植性强、面向对象、可扩展性、可嵌入型、丰富的库、规范的代码等特点。其中，Pandas, Numpy是数据预处理中常用到的库。在本书的最后两章，我们将介绍如何调用这些库，完成数据的导入导出和清洗工作。

【72】

本章分别从背景、目的、流程、注意事项、工具等5个主题介绍了数据预处理。其中，我们要了解数据预处理的问题背景，由背景引申出目的。为了完成目的，从而有了一整套的流程步骤，在运用这些流程步骤中，有一些注意事项，处理得当，会起到事半功倍的作用。

在接下来的章节中，我们将重点学习如何使用Kettle和Python去处理数据的质量问题，从而满足数据分析数据挖掘的前提需求。