第5章 数据清洗

学习目标

- 掌握重复值处理，能够灵活使用 Kettle 去除原始数据中的重复值；
- 掌握缺失值处理，能够灵活使用 Kettle 删除或填补原始数据中的缺失值；
- 异常值处理，能够灵活使用 Kettle 删除或替换原始数据中的异常值。

数据清洗是一项复杂且烦琐的工作，同时也是 ETL 过程中的关键步骤。数据清洗的目的在于提高数据质量和准确性，将原始数据中的脏数据清洗干净，常见的脏数据包括重复值、缺失值、异常值等。本节将详细讲解如何使用 Kettle 对常见的脏数据进行清洗。

5.1 重复值处理

重复值处理是在原始数据中识别和处理重复的记录。

1. 重复值处理步骤

Kettle 提供了“唯一行（哈希值）”和“去除重复记录”步骤用于重复值处理，它们都可以基于指定字段来识别和删除重复数据，以确保数据的唯一性，具体介绍如下。

（1）“唯一行（哈希值）”步骤。

“唯一行（哈希值）”步骤通过计算指定字段值的哈希值来识别重复数据，将哈希值相同的数据识别为重复。“唯一行（哈希值）”步骤适用于数据量较小的情况，对于大规模数据来说，计算的开销会随之增加。

（2）“去除重复记录”步骤。

“去除重复记录”步骤首先会根据指定字段的值进行排序，然后比较相邻的两行数据，检查这些数据中指定字段的值是否相同。如果字段值相同，那么这些数据就会被识别为重复数据。相较于“唯一行（哈希值）”步骤，“去除重复记录”步骤更适合处理大规模数据。

2. 重复值处理操作

现在有两个 CSV 文件 202307.csv 和 202308.csv，分别记录某公司在 2023 年 7 月份和 2023 年 8 月份的客户信息。这两个 CSV 文件的内容如图 5-1 所示。

从图 5-1 中可以看出，202307.csv 文件和 202308.csv 文件共包含 16 条客户信息，这些信息中存在重复的客户信息。接下来，分别演示如何使用“唯一行（哈希值）”和“去除重复记录”步骤对这两个 CSV 文件进行重复值处理，具体内容如下。

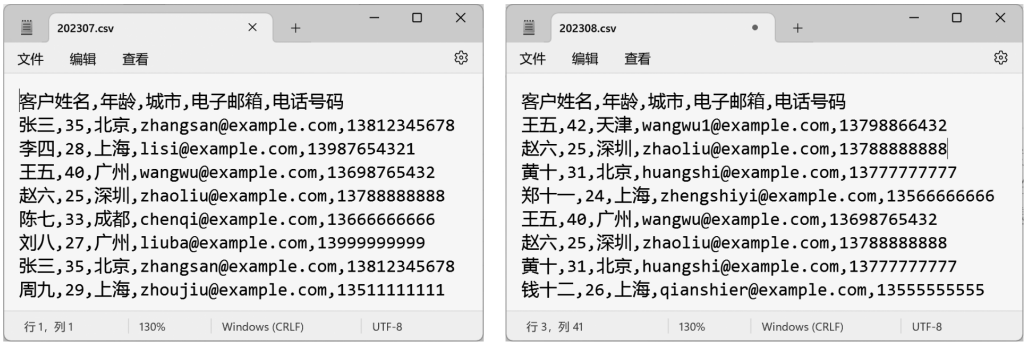


图 5-1 202307.csv 文件和 202308.csv 文件的内容

(1) 使用“唯一行 (哈希值)”步骤进行重复值处理。

要求将 CSV 文件 202307.csv 和 202308.csv 合并后进行重复值处理,去除重复的客户信息,具体操作步骤如下。

① 创建转换。在 Kettle 的图形化界面中创建转换,指定转换的名称为 repeat01。

② 添加步骤。在转换 repeat01 的工作区中添加“文本文件输入”和“唯一行 (哈希值)”步骤,并将这两个步骤通过跳进行连接,用于实现对 CSV 文件 202307.csv 和 202308.csv 进行重复值处理的功能。转换 repeat01 的工作区如图 5-2 所示。

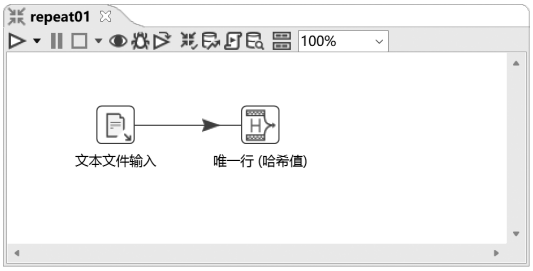


图 5-2 转换 repeat01 的工作区

在图 5-2 中,“文本文件输入”步骤用于从 CSV 文件 202307.csv 和 202308.csv 中抽取数据。之所以选择使用“文本文件输入”步骤而不是“CSV 文件输入”步骤,是因为前者能够同时从多个文件中抽取数据,而后者仅适用于单个文件的数据抽取。

③ 配置“文本文件输入”步骤。在转换 repeat01 的工作区中,双击“文本文件输入”步骤打开“文本文件输入”窗口,在该窗口中分别将 CSV 文件 202307.csv 和 202308.csv 添加到“选中的文件”部分,如图 5-3 所示。

在图 5-3 中,单击“内容”选项卡标签,将“分隔符”输入框的值修改为英文半角逗号。然后在“编码方式”输入框中填写 UTF-8,如图 5-4 所示。

在图 5-4 中,单击“字段”选项卡标签,在该选项卡中单击“获取字段”按钮,在弹出的 Sample data 对话框中不做任何修改,直接单击“确定”按钮。让“文本文件输入”步骤检测 CSV 文件 202307.csv 和 202308.csv,从而推断出字段的信息,如图 5-5 所示。

从图 5-5 中可以看出,“文本文件输入”步骤通过检测这两个 CSV 文件,推断出 5 个字段,这些字段的名称分别为“客户姓名”“年龄”“城市”“电子邮箱”“电话号码”。

在图 5-5 中,单击“确定”按钮保存对当前步骤的配置。

④ 配置“唯一行 (哈希值)”步骤。在转换 repeat01 的工作区中,双击“唯一行 (哈希

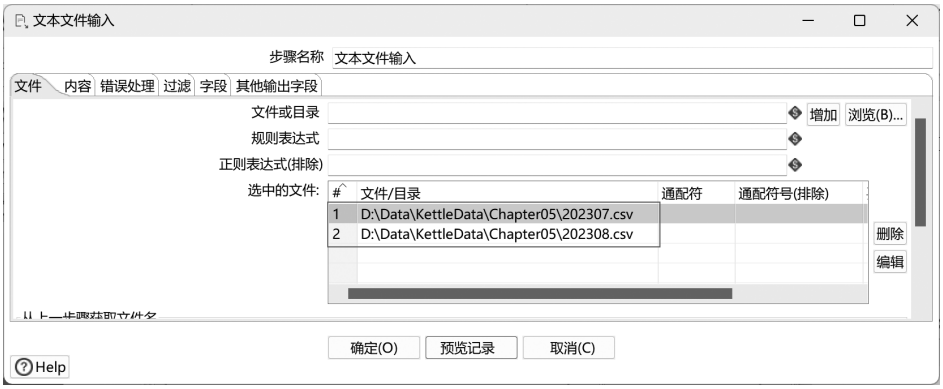


图 5-3　“文本文件输入”窗口(1)

图 5-4　“内容”选项卡(1)

图 5-5　“字段”选项卡(1)

值)”步骤，打开“唯一行（哈希值)”窗口。在该窗口的“字段名称”列添加 5 行内容，分别是“客户姓名”“年龄”“城市”“电子邮箱”“电话号码”，如图 5-6 所示。

图 5-6 中配置的内容意味着通过计算“客户姓名”“年龄”“城市”“电子邮箱”“电话号码”字段的哈希值来识别重复数据，将哈希值相同的数据识别为重复，进而实现去除重复的客户

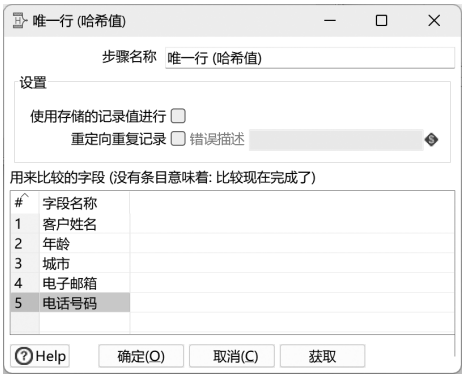
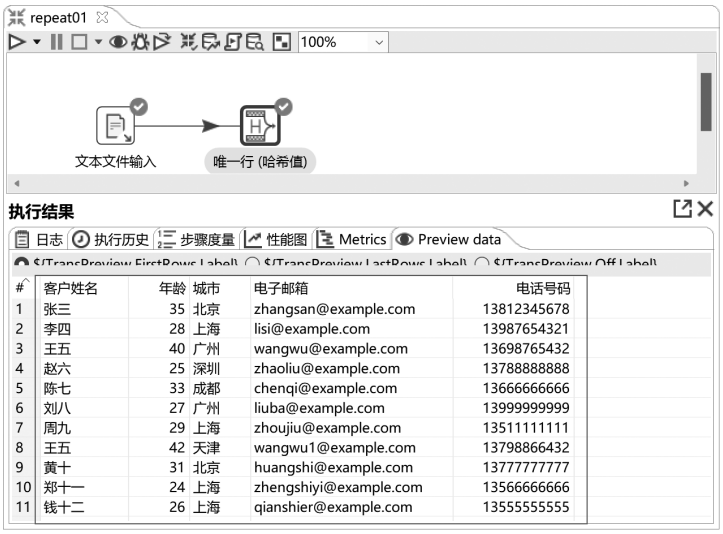


图 5-6 “唯一行(哈希值)”窗口

信息。

在图 5-6 中,单击“确定”按钮保存对当前步骤的配置。

⑤ 运行转换。保存并运行转换 repeat01。当转换 repeat01 运行完成后,在其工作区选择“唯一行(哈希值)”步骤,然后在“执行结果”面板中单击 Preview data 选项卡标签,查看“唯一行(哈希值)”步骤中的数据,如图 5-7 所示。



#	客户姓名	年龄	城市	电子邮箱	电话号码
1	张三	35	北京	zhangsan@example.com	13812345678
2	李四	28	上海	lisi@example.com	13987654321
3	王五	40	广州	wangwu@example.com	13698765432
4	赵六	25	深圳	zhaoliu@example.com	13788888888
5	陈七	33	成都	chenqi@example.com	13666666666
6	刘八	27	广州	liuba@example.com	13999999999
7	周九	29	上海	zhoujiu@example.com	13511111111
8	王五	42	天津	wangwu1@example.com	13798866432
9	黄十	31	北京	huangshi@example.com	13777777777
10	郑十一	24	上海	zhengshiyi@example.com	13566666666
11	钱十二	26	上海	qianshier@example.com	13555555555

图 5-7 查看“唯一行(哈希值)”步骤中的数据

从图 5-7 中可以看出,“唯一行(哈希值)”步骤中的数据不包含重复的客户信息,这说明使用“唯一行(哈希值)”步骤成功去除了 CSV 文件 202307.csv 和 202308.csv 中重复的客户信息。

(2) 使用“去除重复记录”步骤进行重复值处理。

要求将 CSV 文件 202307.csv 和 202308.csv 合并后进行重复值处理,通过去除重复的城市,获取客户覆盖的城市范围,具体操作步骤如下。

① 创建转换。在 Kettle 的图形化界面中创建转换,指定转换的名称为 repeat02。

② 添加步骤。在转换 repeat02 的工作区中添加“文本文件输入”“排序记录”“去除重复记录”“字段选择”步骤,并将这 4 个步骤通过跳进行连接,用于实现对 CSV 文件 202307.csv 和 202308.csv 进行重复值处理的功能。转换 repeat02 的工作区如图 5-8 所示。

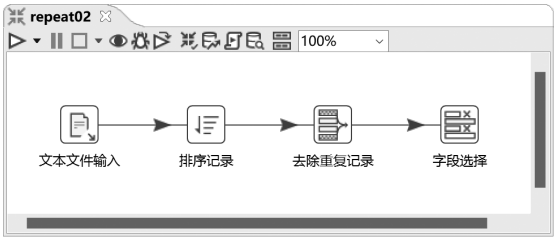


图 5-8　转换 repeat02 的工作区

在图 5-8 中，“文本文件输入”步骤用于从 CSV 文件 202307.csv 和 202308.csv 抽取数据。“排序记录”步骤用于对抽取的数据进行排序。“去除重复记录”步骤用于对排序后的数据进行重复值处理。“字段选择”步骤用于从重复值处理后的数据中选择城市信息。

③ 配置“文本文件输入”步骤。转换 repeat02 和 repeat01 中“文本文件输入”步骤的配置内容一致，这里不再赘述。

④ 配置“排序记录”步骤。在转换 repeat02 的工作区中，双击“排序记录”步骤打开“排序记录”窗口。在该窗口中“字段”部分的“字段名称”和“升序”列分别填写“城市”和“是”，表示根据字段“城市”的值对数据进行升序排序，如图 5-9 所示。

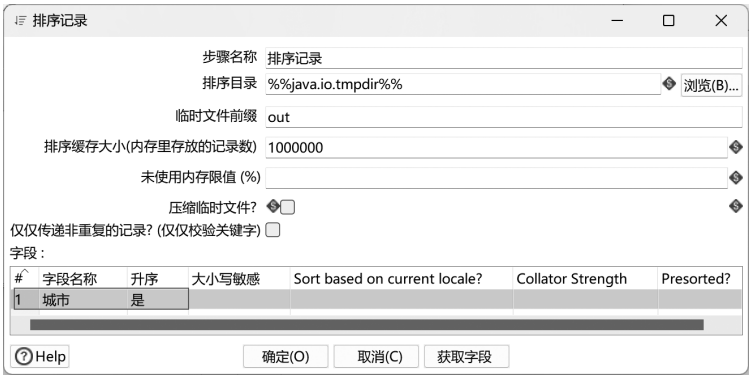


图 5-9　“排序记录”窗口

在图 5-9 中，单击“确定”按钮保存对当前步骤的配置。

⑤ 配置“去除重复记录”步骤。在转换 repeat02 的工作区中，双击“去除重复记录”步骤打开“去除重复记录”窗口，在该窗口的“字段名称”列填写“城市”，如图 5-10 所示。

图 5-10 配置的内容表示比较相邻的两行数据，检查这些数据中字段“城市”的值是否相同。如果字段值相同，那么这些数据就会被识别为重复数据。在图 5-10 中，单击“确定”按钮保存对当前步骤的配置。

⑥ 配置“字段选择”步骤。在转换 repeat02 的工作区中，双击“字段选择”步骤打开“选择/改名值”窗口。在该窗口的“字段名称”列填写“城市”，如图 5-11 所示。

图 5-11 配置的内容表示获取数据中字段“城市”的值。在图 5-11 中，单击“确定”按钮保存对当前步骤的配置。

⑦ 运行转换。保存并运行转换 repeat02。当转换 repeat02 运行完成后，在其工作区选择“字段选择”步骤，然后在“执行结果”面板中单击 Preview data 选项卡标签，查看“字段选择”步骤中的数据，如图 5-12 所示。

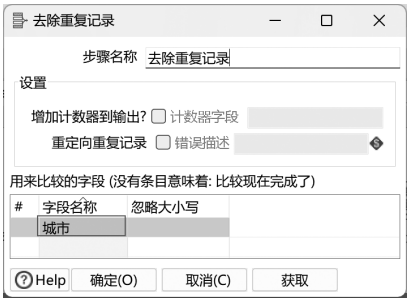


图 5-10 “去除重复记录”窗口

图 5-11 “选择/改名值”窗口(1)

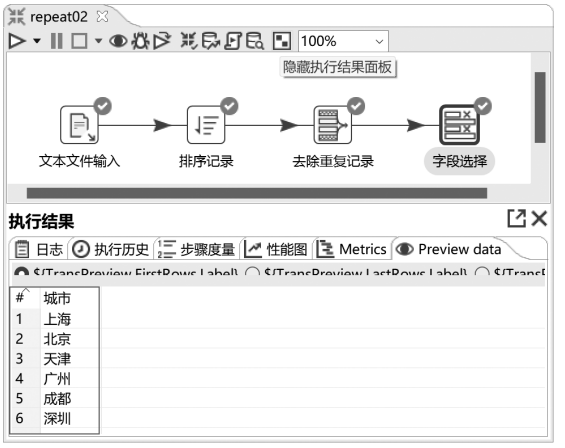


图 5-12 查看“字段选择”步骤中的数据(1)

从图 5-12 中可以看出,客户覆盖的城市包括上海、北京、天津、广州、成都和深圳。说明使用“去除重复记录”步骤成功去除 CSV 文件 202307.csv 和 202308.csv 中重复的城市。

5.2 缺失值处理

在进行数据清洗时,经常会遇到缺失值的情况。这些缺失值可能源于多种原因,例如记录错误、系统故障或样本选择偏差。但不论造成缺失的具体原因是什么,缺失值都在数据中留下了空白,给随后的分析和建模带来了一定挑战。在本节中,将详细介绍如何借助 Kettle 有效处理缺失值问题。

5.2.1 缺失值处理策略

忽略或不适当地处理缺失值可能导致错误的结论,甚至影响对问题的深入理解。因此,制定合理的缺失值处理策略有助于提升数据的可靠性,对分析或建模的结果具有重要影响。缺失值的处理通常可以根据缺失值的重要性和缺失率分为以下 4 种情况,如图 5-13 所示。

在图 5-13 中,每种情况分别对应了不同的缺失值处理策略,具体介绍如下。

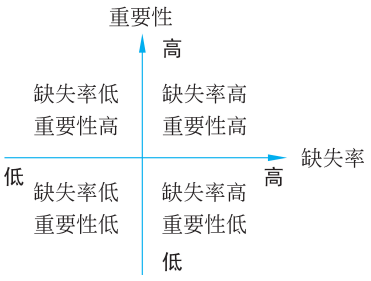


图 5-13 缺失值的重要性和缺失率

1. 缺失率低并且重要性高

当缺失值对于分析或建模结果具有重要影响，但缺失率相对较低时，直接删除缺失值可能会导致信息损失。因此，应该考虑使用更复杂的填补方法，例如插值填补，以更好地保留有价值的信息并减少对分析或建模结果的影响。

2. 缺失率高并且重要性高

当缺失值对分析或建模结果影响较大且缺失率较高时，缺失值处理变得更具挑战性。在这种情况下，需要综合应用多种方法，并根据缺失值的特性和领域知识选择合适的处理策略。在使用填充方法时，需要特别小心，并可能需要使用特定模型来处理缺失值。

3. 缺失率低并且重要性低

当缺失值对分析或建模结果影响较小，且缺失率较低时，可以考虑直接删除缺失值。这样可以简化数据集，减少噪声的引入，并且不会对结果产生显著影响。

4. 缺失率高并且重要性低

当缺失值对分析或建模结果影响较小，但缺失率较高时，删除缺失值可能会导致大量信息丢失。在这种情况下，可以考虑使用填充方法，如使用平均值填充、中位数填充或众数填充。这样可以在保留数据的同时，避免引入过多的偏差。

在实际应用中，决定如何处理缺失值需要权衡缺失值的重要性和缺失率，同时考虑分析或建模的目标。没有一种通用的方法适用于所有情况，因此在选择缺失值处理策略时，需要根据具体情况进行合理的判断和决策。这一点提醒我们要认识并发挥自己的优势，并将其转换为实践。专注于自己的优势领域，有助于我们提高表现，并取得更多的成果。

5.2.2　删除缺失值

删除缺失值是一种简单而直接的方法。在删除缺失值之前，需要仔细考虑数据集的性质以及缺失值的分布情况。如果缺失值的比例相对较小，并且可以合理假设这些缺失值是随机分布的，那么删除它们可能不会对整体分析产生显著影响。然而，如果缺失值的比例较大，或者缺失值的分布与特定模式相关，那么删除缺失值可能会导致信息丧失，甚至可能引入偏见。

1. 删除缺失值的方式

删除缺失值的方式主要分为两种，一种是删除包含缺失值的行，另一种是删除包含缺失值的字段(列)。针对这两种方式的介绍如下。

(1) 删除包含缺失值的行。

删除包含缺失值的行的方式是将包含缺失值的整行数据从数据集中移除，适用于缺失值分布较随机且对分析结果影响有限的情况。然而，这种方式可能导致数据减少，特别是在缺失值较多的情况下。因此，在使用这种方式删除缺失值时，需要权衡数据量减少与分析结果影响之间的关系。

(2) 删除包含缺失值的字段。

删除包含缺失值的字段的方式是将包含缺失值的整个字段从数据集中删除，适用于字段的缺失值较多且对分析结果影响有限的情况。但是，需要注意的是，删除字段可能会丢失某些特征信息，因此应在充分理解分析需求的前提下使用。

例如，现在有一个 TSV 文件 sale.tsv，该文件记录了不同产品的销售信息。由于某种原

因,产品的销售信息中产生了缺失值。TSV 文件 sale.tsv 的内容如图 5-14 所示。

从图 5-14 中可以看出,TSV 文件 sale.tsv 中的产品销售信息存在 3 个缺失值,其中产品 C 和产品 I 的售价缺失,产品 H 的销量缺失。

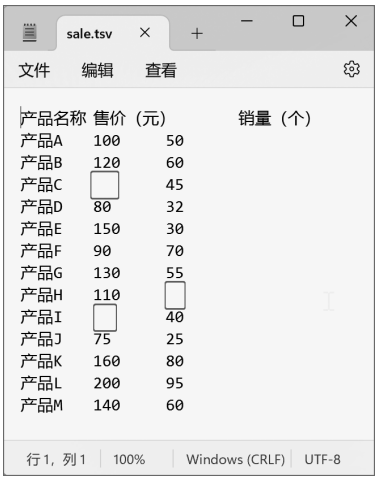


图 5-14 TSV 文件 sale.tsv 的内容

假设,我们的需求是要计算所有产品的平均销量。在这种情况下,产品的售价对于本次计算的结果没有影响。因此,在进行数据清洗时,可以直接删除。除此之外,在所有产品的销量信息中,只有产品 H 的销量缺失。不过,从销售信息的整体角度来看,产品 H 的销量在整个数据集中所占比例较小。鉴于此,可以合理考虑删除产品 H 的销售信息,以确保数据的完整性。

2. 删除缺失值的操作过程

接下来,将演示如何使用 Kettle 删除文件 sale.tsv 中的缺失值,具体操作步骤如下。

(1) 创建转换。

在 Kettle 的图形化界面中创建转换,指定转换的名称为 delete_missing_value。

(2) 添加步骤。

在转换 delete_missing_value 的工作区中添加“文本文件输入”“字段选择”“过滤记录”和两个“空操作 (什么也不做)”步骤,并将这些步骤通过跳进行连接,用于实现删除文件 sale.tsv 中缺失值的功能。转换 delete_missing_value 的工作区如图 5-15 所示。

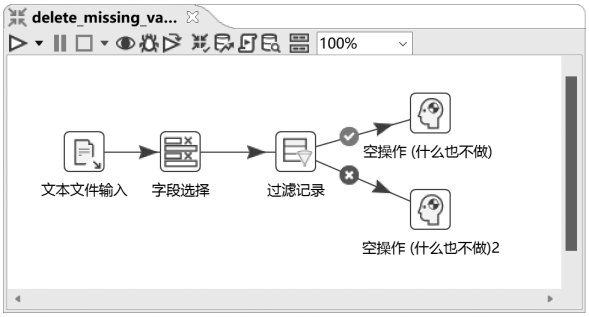


图 5-15 转换 delete_missing_value 的工作区

在图 5-15 中,“文本文件输入”步骤用于从文件 sale.tsv 中抽取数据。“字段选择”步骤用于从销量信息中移除产品的售价。“过滤记录”步骤用于过滤销量信息,分别将不包含缺失值的销售信息和包含缺失值的销售信息输出到“空操作(什么也不做)”和“空操作(什么也不做)2”步骤。

(3) 配置“文本文件输入”步骤。

在转换 delete_missing_value 的工作区中,双击“文本文件输入”步骤打开“文本文件输入”窗口。在该窗口中将 TSV 文件 sale.tsv 添加到“选中的文件”部分,如图 5-16 所示。

在图 5-16 中,单击“内容”选项卡标签,将“分隔符”输入框内的值设为制表符,并且在“编码方式”输入框内填写 UTF-8,如图 5-17 所示。

图 5-16　“文本文件输入”窗口(2)

图 5-17　“内容”选项卡(2)

在图 5-17 中,单击“字段”选项卡标签。在该选项卡中单击“获取字段”按钮让“文本文件输入”步骤检测 TSV 文件 sale.tsv,从而推断出字段的信息,如图 5-18 所示。

图 5-18　“字段”选项卡(2)

从图 5-18 中可以看出,“文本文件输入”步骤推断出 TSV 文件 sale.tsv 包含 3 个字段,

这些字段的名称分别为“产品名称”“售价(元)”“销量(个)”。

在图 5-18 中,单击“确定”按钮保存对当前步骤的配置。

(4) 配置“字段选择”步骤。

在转换 delete_missing_value 的工作区中,双击“字段选择”步骤打开“选择/改名值”窗口。在该窗口单击“移除”选项卡标签,在该选项卡的“字段名称”列填写“售价(元)”,如图 5-19 所示。

图 5-19 中配置的内容表示从数据中移除字段“售价(元)”。在图 5-19 中,单击“确定”按钮保存对当前步骤的配置。

(5) 配置“过滤记录”步骤。

在转换 delete_missing_value 的工作区中,双击“过滤记录”步骤打开“过滤记录”窗口。在该窗口的“条件”部分添加判断条件。首先,单击左侧的<field>选项打开“字段”窗口,在该窗口中双击“销量(个)”选项。然后,单击=选项打开“函数”窗口,在该窗口中双击 IS NOT NULL 选项,如图 5-20 所示。

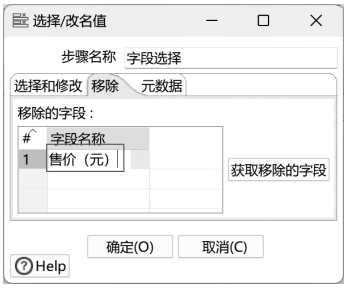


图 5-19 “移除”选项卡

图 5-20 “过滤记录”窗口(1)

图 5-20 中配置的内容意味着,判断字段“销量(个)”的值是否为 NULL。如果判断结果为 true,说明字段“销量(个)”的值不为 NULL,那么将数据输出到“空操作(什么也不做)”步骤。反之,如果判断结果为 false,说明字段“销量(个)”的值为 NULL,那么将数据输出到“空操作(什么也不做) 2”步骤。

在图 5-20 中单击“确定”按钮保存对当前步骤的配置。

(6) 配置“空操作(什么也不做)”步骤。

为了区分两个“空操作(什么也不做)”步骤的含义,这里将该步骤的名称设置为“不包含缺失值”。

(7) 配置“空操作(什么也不做)2”步骤。

为了区分两个“空操作(什么也不做)”步骤的含义,这里将该步骤的名称设置为“包含缺失值”。

(8) 运行转换。

保存并运行转换 delete_missing_value。当转换 delete_missing_value 运行完成后,在其工作区选择“不包含缺失值”步骤,然后在“执行结果”面板中单击 Preview data 选项卡标签,查看“不包含缺失值”步骤中的数据,如图 5-21 所示。

从图 5-21 中可以看出,“不包含缺失值”步骤中的数据不包含缺失值,说明使用 Kettle 成功删除文件 sale.tsv 中的缺失值。