

课程设计报告

**姓 名：** 赵璐彤

**学 号：** 19851070

**专 业：**计算机科学与技术

**系 别：**计算机科学与技术

**指导教师：** 王志海

**2022年6月**

# 机器学习应用系统与大数据分析

Weka是用于数据挖掘认为的机器学习算法的集合。它包含用于数据准备、分类、回归、聚类、关联规则挖掘和可视化的工具。

Weka只在新西兰的岛屿上发现，是一种不会飞的鸟，具有好奇的天性。这个名字的发音是这样的，鸟儿听起来像这样。

Weka是根据GNU通用公共许可证发布的开源软件。

## 1 Weka数据分析网站

Weka是怀卡托智能分析环境(Waikato Environment for Knowledge Analysis)的英文字首缩写，官方网址为：<http://www.cs.waikato.ac.nz/ml/weka>，在该网站可以免费下载可运行软件和源代码，还可以获得说明文档、常见问题解答、数据集和其他文献等资源。Weka的发音类似新西兰本土一种不会飞的鸟，因此Weka系统使用该鸟作为其徽标。

Weka是新西兰怀卡托大学用Java开发的数据挖掘著名开源软件，该系统自1993年开始由新西兰政府资助，至今已经历了20年的发展，其功能已经十分强大和成熟。Weka集合了大量的机器学习和相关技术，受领域发展和用户需求所推动，代表了当今数据挖掘和机器学习领域的最高水平。Weka是一种使用Java语言编写的数据挖掘机器学习软件，是GNU协议下分发的开源软件。Weka主要用于科研、教育和应用领域。

国内外很多著名大学都采用Weka作为数据挖掘和机器学习课程的实践工具。Weka还有另外一个名字叫Pentaho Data Mining Community Edition(Pentaho数据挖掘社区版)，此外，Pentaho的网站(<http://weka.pentaho.com/>)还维护一个称为Pentaho Data Mining Enterprise Edition(Pentaho数据挖掘企业版)的版本，它主要提供技术支持和管理升级。另一个用Java编写的著名数据挖掘工具RapidMiner通过Weka Extension(Weka扩展)支持Weka，以充分利用Weka的“约100个额外的建模方案，其中包括额外的决策树、规则学习器和回归估计器”，参见网址<http://rapid-i.com/content/view/202/206/>[5]。

### Weka历史

1992年末，新西兰怀卡托大学计算机科学系Ian Witten博士申请基金，1993年获新西兰政府资助，并于同年开发出接口和基础架构。次年发布了第一个Weka的内部版本，两年后，在1996年10月，第一个公开版本(Weka 2.1)发布。Weka早期版本主要采用C语言编写，1997年初，团队决定使用Java重新改写，并在1999年中期发布纯Java的Weka 3版本。选定Java来实现Ian Witten著作《Data Mining》的配套机器学习技术是有充分理由的，作为一个著名的面向对象的编程语言，Java允许用一个统一的接口来进行学习方案和方法的预处理和后处理。决定使用Java来替代C++或其他面向对象的语言，是因为Java编写的程序可以运行在绝大部分计算机上，而无须重新编译，更不需要修改源代码。已经测试过的平台包括Linux、Windows和Macintosh操作系统，甚至包括PDA。最后的可执行程序复制过来即可运行，完全绿色，不要求复杂安装。当然，Java也有其缺点，最大的问题是它在速度上有缺陷，执行一个Java程序比对应的C语言程序要慢上好几倍。综合来看，对于Weka来说，Java“一次编译，到处运行”的吸引力远远超出对性能的渴望[5]。

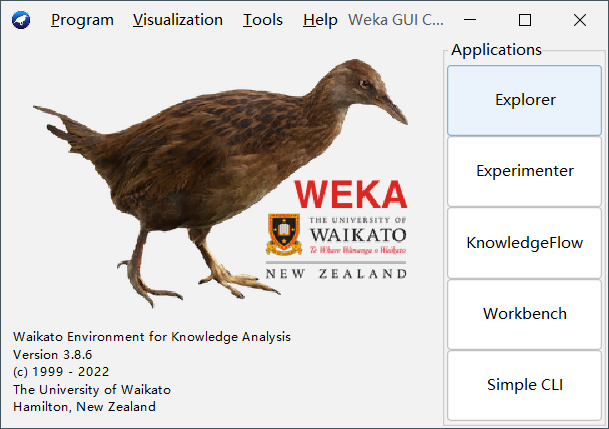
## 2 Weka软件系统功能

Weka系统汇集了最前沿的机器学习算法和数据预处理工具，以便用户能够快速灵活地将已有的处理方法应用于新的数据集。它为数据挖掘的整个过程提供全面的支持，包括准备输入数据、统计评估学习方案、输入数据和学习效果的可视化。Weka除了提供大量学习算法之外，还提供了适应范围很广的预处理工具，用户通过一个统一界面操作各种组件，比较不同的学习算法，找出能够解决问题的最有效的方法。

**Weka系统包括处理标准数据挖掘问题的所有方法：回归、分类、聚类、关联规则以及属性选择**。分析要进行处理的数据是重要的一个环节，Weka提供了很多用于数据可视化和预处理的工具。**输入数据可以有两种形式，第一种是以ARFF格式为代表的文件；另一种是直接读取数据库表。**

**使用Weka的方式主要有三种**：**第一种**是将学习方案应用于某个数据集，然后分析其输出，从而更多地了解这些数据；**第二种**是使用已经学习到的模型对新实例进行预测；**第三种**是使用多种学习器，然后根据其性能表现选择其中的一种来进行预测。用户使用交互式界面菜单中选择一种学习方法，大部分学习方案都带有可调节的参数，用户可通过属性列表或对象编辑器修改参数，然后通过同一个评估模块对学习方案的性能进行评估。

**Weka主界面称为Weka GUI选择器**，它通过**右边的四个按钮提供四种主要的应用程序供用户选择**，如下图所示，**用鼠标单击按钮进入到相应的图形用户界面**。

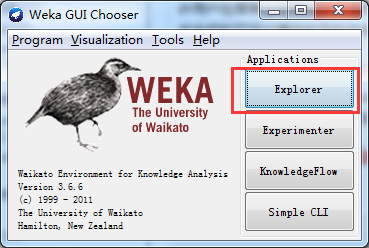


其中，Weka系统提供的最容易使用的图形用户接口称为探索者(Explorer)。通过选择菜单和填写表单，可以调用Weka的所有功能。例如，用户用鼠标仅仅单击几个按钮，就可以完成从ARFF文件中读取数据集，然后建立决策树的工作。

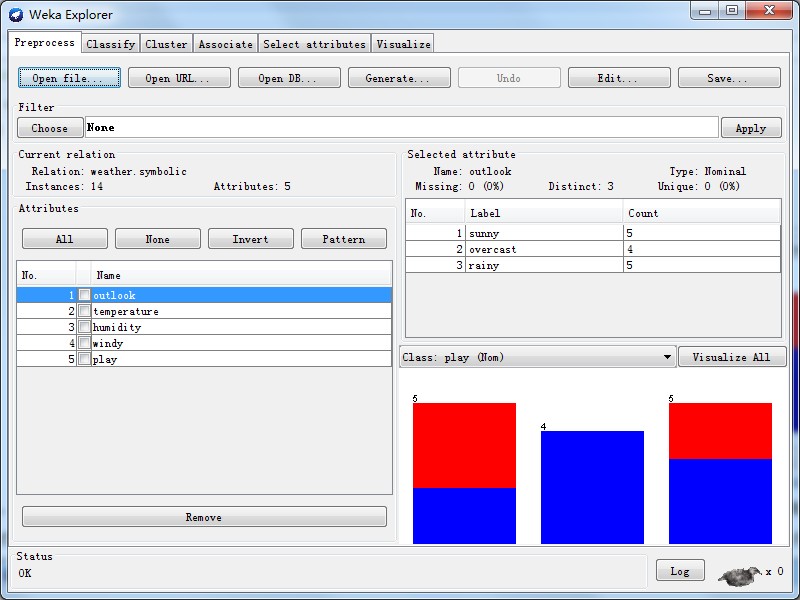
Weka界面十分友好，能适时地将不宜用的功能选项设置为不可以选；将用户选项设计为表格方式以方便填写；当鼠标移动到界面工具上短暂停留时，会给出用法提示；对算法都给出较为合理的默认值，这样，帮助用户尽量少花精力进行配置就可取得较好的效果等。

### 2.1 **Explorer（探索者）**

虽然探索者（Explorer）界面使用很方便，但它也存在一个缺陷，要求它将所需数据全部一次读进内存，一旦用户打开某个数据集，就会读取全部数据。因此，这种批量方式仅适合处理中小规模的问题。知识流刚好能够弥补这一缺陷。



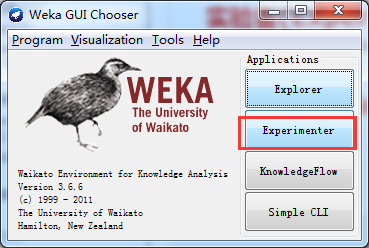
系统提供的最容易使用的图像用户接口。通过选择菜单和填写表单，可以调用Weka的所有功能。



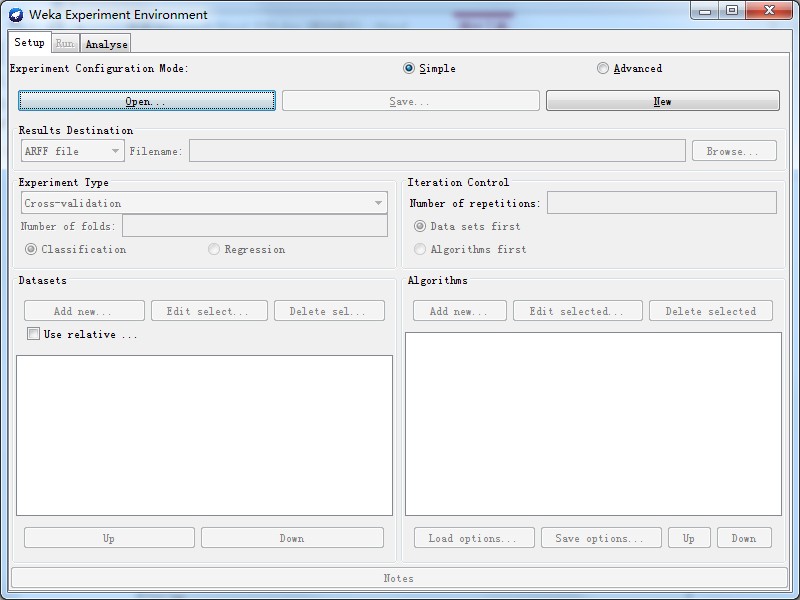
虽然探索者界面使用很方便，但它也存在一个缺陷，要求它将所需数据全部一次读进内存，一旦用户打开某个数据集，就会读取全部数据。因此，这种批量方式仅适合处理中小规模的问题。而**知识流**刚好能够弥补这一缺陷。

### 2.2 **Experimenter（实验者界面）**

**实验者(Experimenter)界面**用于帮助用户解答实际应用分类和回归技术中遇到的一个基本问题——对于一个已知问题，哪种方法及参数值能够取得最佳效果？通过Weka提供的实验者工作环境，用户可以比较不同的学习方案。尽管探索者界面也能通过交互完成这样的功能，但通过实验者界面，用户可以让处理过程实现自动化。实验者界面更加容易使用不同参数去设置分类器和过滤器，使之运行在一组数据集中，收集性能统计数据，实现重要的测试实验。

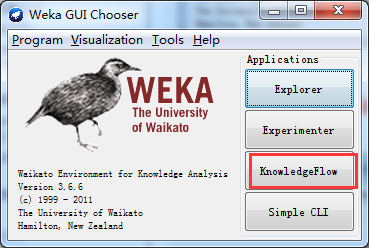


**Experimenter（实验者）**，用于帮助用户解答实际应用分类和回归技术中遇到的一个基本问题-----对于一个已知问题，哪种方法及参数值能够取得最佳效果？尽管**探索者界面**也能通过交互完成这样的功能，但通过**实验者界面**，用户可以让处理过程实现自动化。实验者界面更加容易使用不同参数去设置分类器和过滤器，使之运行在一组数据集中，收集性能统计数据，实现重要的测试实验。

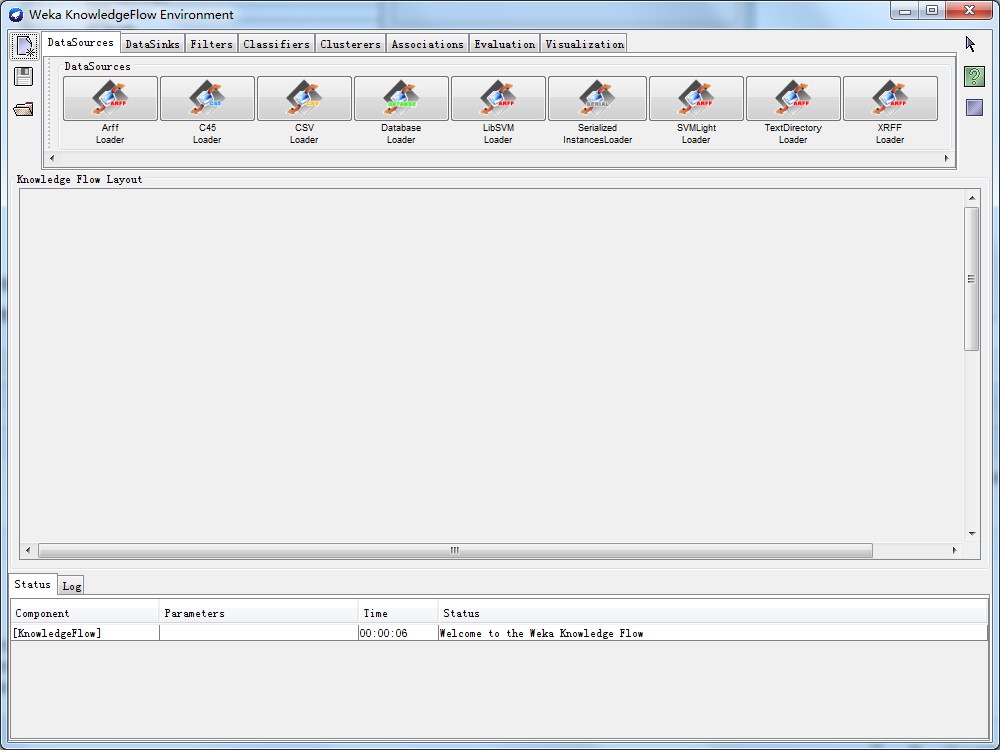


### 2.3 **Knowledge Flow（知识流界面）**

**知识流(Knowledge Flow)界面**可以使用增量方式的算法来处理大型数据集，用户可以定制处理数据流的方式和顺序。知识流界面允许用户在屏幕上任意拖曳代表学习算法和数据源的图形构件，并以一定的方式和顺序组合在一起。也就是，按照一定顺序将代表数据源、预处理工具、学习算法、评估手段和可视化模块的各构件组合在一起，形成数据流。如果用户选取的过滤器和学习算法具有增量学习功能，那就可以实现大型数据集的增量分批读取和处理。

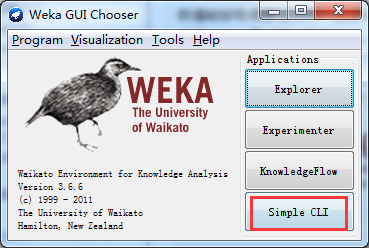


**Knowledge Flow（知识流）**，使用增量方式的算法来处理大型数据集，用户可以定制处理数据流的方式和顺序。按照一定顺序将代表数据源、预处理工具、学习算法、评估手段和可视化模块的各构件组合在一起，形成数据流。

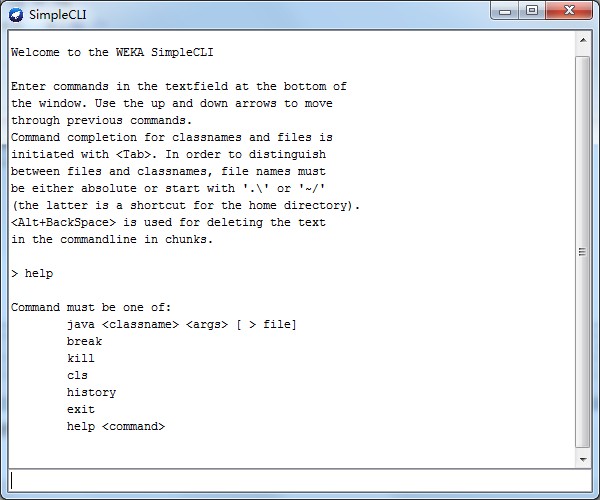


### 2.4 **Simple CLI（简单命令行）**

**简单命令行(Simple CLI)界面**是为不提供自己的命令行界面的操作系统提供的，该简单命令行界面用于和用户进行交互，可以直接执行Weka命令。



**Simple CLI（简单命令行）**，这个界面是为不提供自己的命令行界面的操作系统提供的，该简单命令行界面用户和用户进行交互，可以直接执行Weka命令。



## 3 数据集合格式

### 术语

Weka存储数据的格或是ARFF(Attribute-Relation File Format)文件，是一种ASCLL文本文件。如下例，Weka自带的文件。

Weka数据以“%”开始的行是注释, Weka将忽略这些行。除去注释后，整个ARFF文件可以分为两个部分。

第一部分给出了头信息(Head information),包括了对关系的声明和对属性的声明。

第二部分给出了数据信息(Data information)，即数据集中给出的数据。从“@data”标记开始，后面的就是数据信息了。关系声明

关系名称在ARFF文件的第一个有效行来定义，格式为@relation <relation-name>

<relation-name>是一个字符串。如果这个字符串包含空格，它必须加上引号（指英文标点的单引号或双引号)。

### 属性声明

属性声明用一列以“@attribute”开头的语句表示。数据集中的每一个属性都有它对应的“@attribute”语句，来定义它的属性名称和数据类型。

声明语句的顺序很重要:它表明了该项属性在数据部分的位置;最后一个声明的属性被称作class属性，在分类或回归任务中，它是默认的目标变量。

属性声明的格式为

@attribute <attribute-name> <datatype>

其中<attribute-name>是必须以字母开头的字符串。和关系名称一样，如果这个字符串包含空格，它必须加上引号。

Weka支持的<datatype>有四种，分别是

Numeric 数值型

<nominal-specification> 分类(nominal)型

String 字符串型

date [<date-format>] 日期和时间型

注意“integer”, “real”, “numeric”, “date”, “string”这些关键字是区分大小写的，而“relation”“attribute”和“date”叫不区分。

数值属性

数值型属性可以是整数或者实数，但Weka把它们都当作实数看待。

### 分类属性

分类属性由<nominal-specification>列出一系列可能的类别名称并放在花括号中:{nominal-name l>,<nominal-name2>, cnomina-name3>,..}。数据集中该属性的值只能是其中一种类别。

例如如下的属性声明说明“outlook”属性有三种类别: “sunny”, “ overcast”和“rainy”。而数据集中每个实例对应的“outlook”值必是这三者之一。

@attribute outlook {sunny, overcast, rainy}如果类别名称带有空格，仍需要将之放入引号中。

字符串属性

字符串属性中可以包含任意的文本。这种类型的属性在文本挖掘中非常有用。示例: @ATTRIBUTE LCC string

日期和时间属性

日期和时间属性统一用“date”类型表示，它的格式是@attribute <name> date [<date-format>]

其中<name>是这个属性的名称,<date-format>是一个字符串，来规定该怎样解析和显示日期或时间的格式，默认的字符串是ISO-8601所给的日期时间组合格式“yyyy – MM - ddTHH:mm:ss”。

数据信息部分表达日期的字符串必须符合声明中规定的格式要求。

### 数据信息

数据信息中“@data”标记独占一行，剩下的是各个实例的数据。

每个实例占一行。实例的各属性值用逗号“”隔开。如果某个属性的值是缺失值(missing value)，用问号“?”表示，且这个问号不能省略。例如:

@data

sunny,85,85,FALSE,no,78,90,,yes

字符串属性和分类属性的值是区分大小写的。若值中含有空格，必须被引号括起来。例如;

@relation LCC vs LCSH @attribute LCC string @attribute LCSH string @data

AG5, 'Encyclopedias and dictionaries. ;Twentieth century.'AS262, 'Science -- Soviet Union -- History.'

日期属性的值必须与属性声明中给定的相一致。例如:@RELATION Timestamps

@ATTRIBUTE timestamp DATE " yyyy -MM-dd HH: mm: ss" @DATA

"2001-04-03 12:12:12""2001-05-03 12:59:55"

### 稀疏数据

有的时候数据集中含有大量的O值（比如购物分析)，这个时候用稀疏格式的数据存贮更加省空间。稀疏格式是针对数据信息中某个实例的表示而言，不需要修改ARFF文件的其它部分。看如下的数据:@data

0,X,O,Y, "class A"0,o,W,O, "class B"用稀疏格式表达的话就是@data

{1 X,3 Y,4 "class A"{2 w,4 "class B""}

每个实例用花括号括号起来。实例中每一个非O的属性值用<index <空格> <value>表示。<index>是属性的序号，从O开始计; <value>是属性值。属性值之间仍用逗号隔开。

注意在稀疏格式中没有注明的属性值不是缺失值，而是0值。若要表示缺失值必须显式的用问号表示出来。

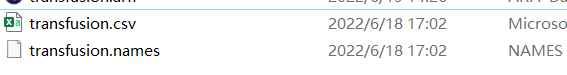
表1-1 weather数据集合

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | outlook | temperature | humidity | windy | play |
| 1 | sunny | hot | high | FALSE | no |
| 2 | sunny | hot | high | TRUE | no |
| 3 | overcast | hot | high | FALSE | yes |
| 4 | rainy | mild | high | FALSE | yes |
| 5 | rainy | cool | normal | FALSE | yes |
| 6 | rainy | cool | normal | FALSE | yes |
| 7 | overcast | cool | normal | TRUE | yes |
| 8 | sunny | mild | high | FALSE | no |
| 9 | sunny | cool | normal | FALSE | yes |
| 10 | rainy | mild | normal | FALSE | yes |
| 11 | Sunny | mild | normal | TRUE | yes |
| 12 | overcast | mild | high | TRUE | yes |
| 13 | overcast | hot | normal | FALSE | yes |
| 14 | rainy | mild | high | TRUE | no |

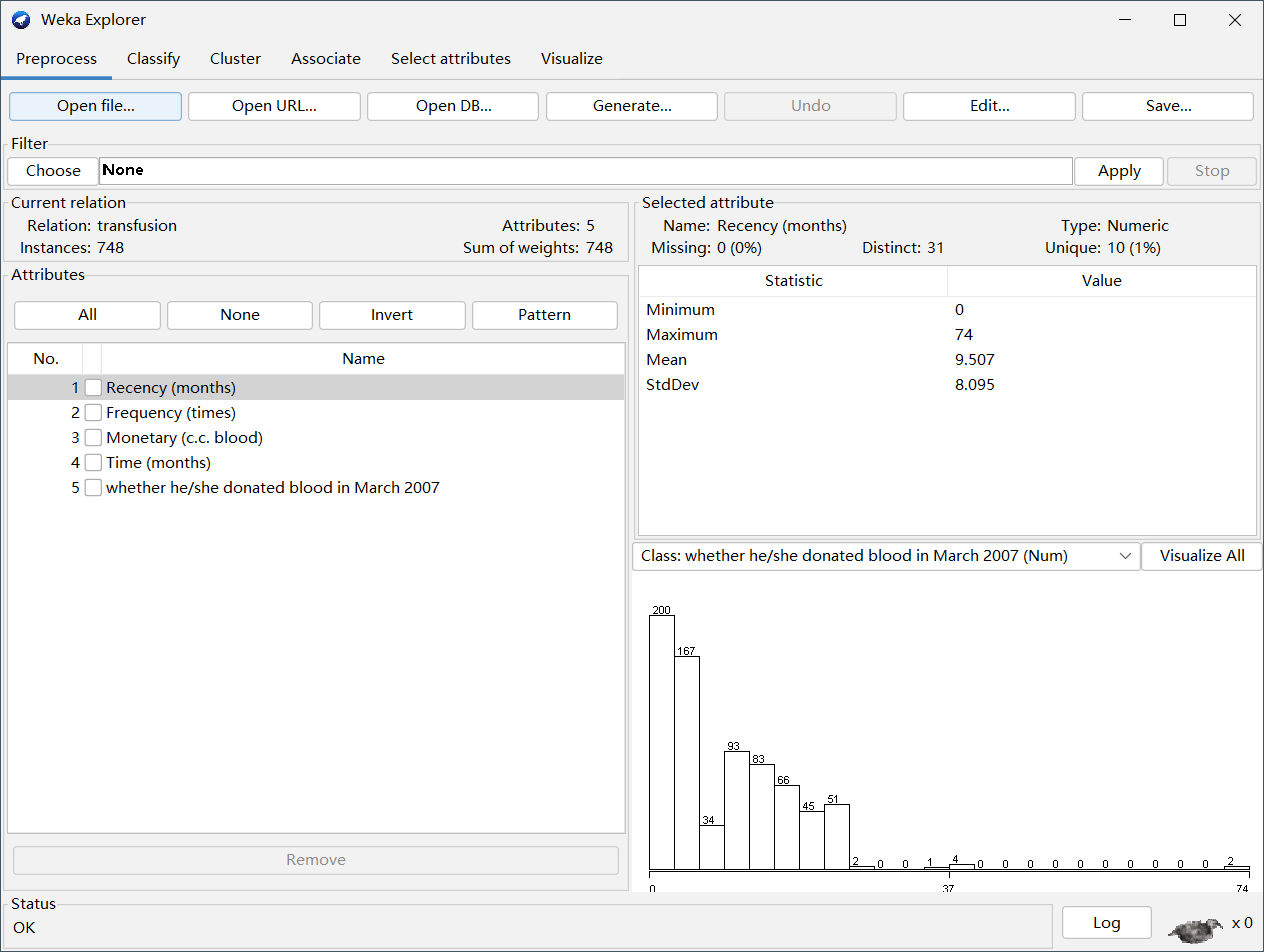
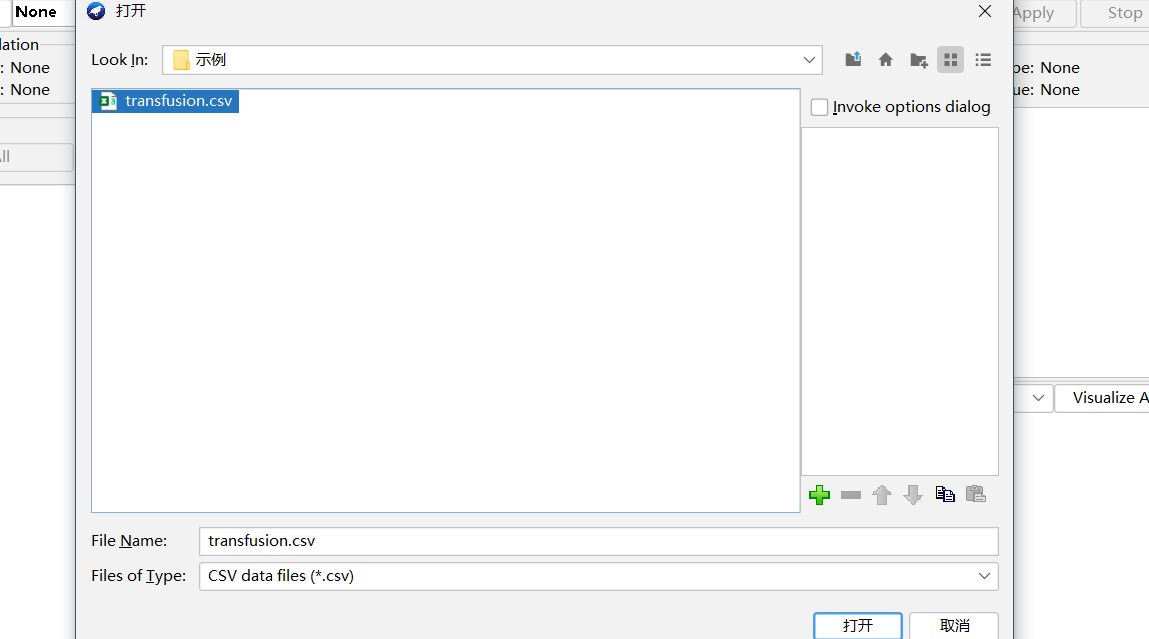
### 示例

将从UCI下载的数据集中的 .data 文件转换为和 .names 文件

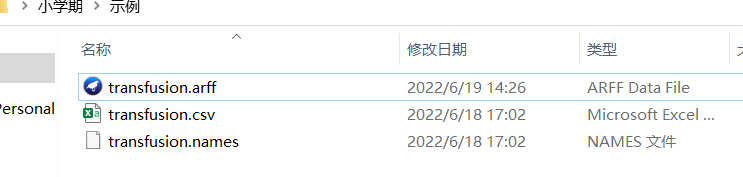
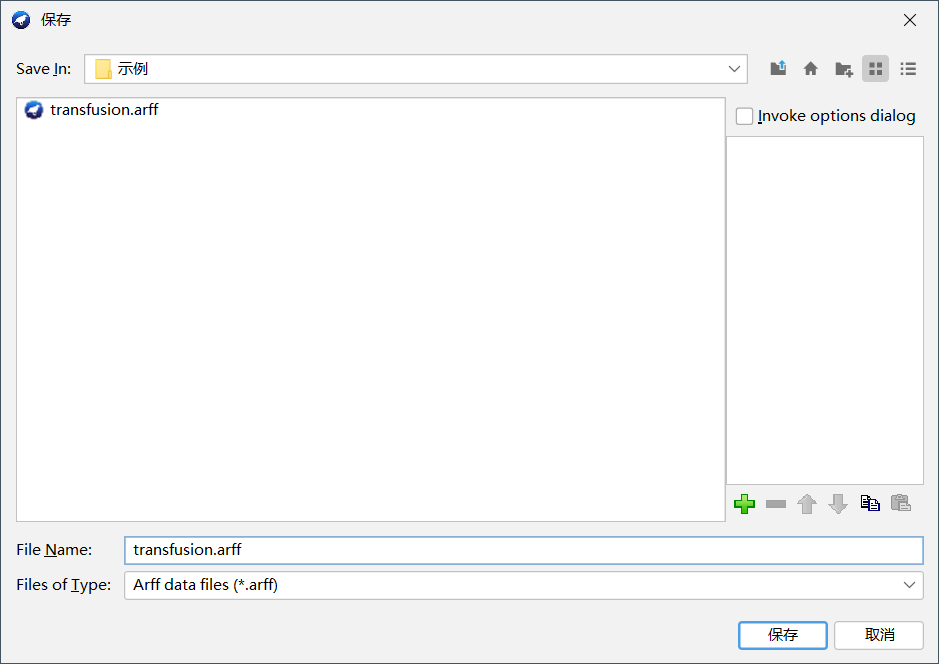
之后将 .data 文件转换为 .csv 文件



打开Weka软件，并且导入该 .csv 文件



之后进行导出 .arff 文件



导出成功

R（最近-自上次捐赠后的月数）

F（频率-捐赠总数）

M（货币-以c.c.表示的献血总量）

T（首次捐赠后的时间-月）

V（表示他/她是否在2007年3月献血（1代表献血；0表示不献血））

表1-2transfusion数据集合

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | R | F | M | T | V |
| 1 | 0 | 13 | 3250 | 28 | 1 |
| 2 | 1 | 16 | 4000 | 35 | 1 |
| 3 | 2 | 20 | 5000 | 45 | 1 |
| 4 | 1 | 24 | 6000 | 77 | 0 |
| 5 | 4 | 4 | 1000 | 4 | 0 |
| 6 | 2 | 7 | 1750 | 14 | 1 |
| 7 | 1 | 12 | 3000 | 35 | 0 |
| 8 | 1 | 12 | 3000 | 35 | 0 |
| 9 | 2 | 9 | 2250 | 22 | 1 |

## 参考文献

[1] Weka官网, URL: https://sourceforge.net/projects/weka/ [2022-6-16]

[2] 百度百科, URL: https://baike.baidu.com/item/weka/10701215 [2022-6-16].

[3] 知乎,URL: https://zhuanlan.zhihu.com/p/57663416 [2022-6-16]

[4] CSDN,URL: https://blog.csdn.net/tao\_wei162/article/details/84881457 [2022-6-16]

[5] 百度百科,URL: https://www.shuzhiduo.com/A/1O5EAQob57/ [2022-6-17]

[6] 百度百科, URL: https://baike.baidu.com/item/weka/10701215 [2022-6-16].