

课程设计报告

**建立人工智能应用系统**

姓 名： 高政

学 号： 19851242

专 业： 计算机科学与技术

系 别： 计算机与信息技术

指导教师： 王志海

**2022年6月19日**

# 《信息系统集成与开发》课程设计报告

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 数据分析网站简介

Weka的全名是怀卡托智能分析环境，是一款免费的，非商业化的，基于Java环境下开源的机器学习以及数据挖掘软件，Weka的主要开发者来自新西兰[1]。

### 1.1 Weka及其团队简介

机器学习方法（ML）的发明和应用是计算机科学中一个激动人心且可能影响深远的发展。这些使计算机程序能够自动分析大量数据，并确定哪些信息最相关。这些具体化的信息可以用来自动做出预测，或帮助人们更快、更准确地做出决策。

Weka的团队已经将几种标准的ML技术集成到一个名为Weka的软件“工作台”中，用于Waikato环境的知识分析。有了它，特定领域的专家能够使用ML从太大而无法手动分析的数据库中获取有用的知识。Weka的用户是ML研究人员和工业科学家，但它也广泛用于教学。最近，此团队还致力于MOA，一种用于挖掘数据流的环境。

Weka是用于数据挖掘任务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。同时Weka是根据GNU通用公共许可证发布的开源软件，并且支持深度学习[2]。

### 1.2 项目目标

1. 使ML技术普遍可用；

2. 将其应用于对新西兰工业至关重要的实际问题；

3. 开发新的机器学习算法并向世界推广；

4. 为该领域的理论框架做出贡献。

## 2 软件系统功能

Weka系统汇集了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成熟处理方法应用于新的数据集。

### 2.1 Weka的历史

1992年末，新西兰怀卡托大学计算机科学系Ian Written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。

1994年发布了第一个Weka的内部版本。

1996年，第一个公开版本Weka2.1发布。Weka的早期版本主要采用C语言编写。

1997年，团队决定使用Java重新改写。Java允许用一个统一的接口来学习方案和方法的预处理和后处理，决定使用Java来代替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。Weka已经测试过的平台包括Linux, Windows和Macintosh操作系统，甚至PDA。最后的可执行程序复制过来即可运行，完全绿色，不需求复杂安装。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程序比相应的C语言程序马上好几倍。

1999年，发布Java的Weka 3版本。

2013年，Weka的最新版本是3.7.8。

### 2.2 Weka简介

Weka作为一个公开的数据挖掘工作平台，集合了大量能承担数据挖掘任务的机器学习算法，包括对数据进行预处理，分类，回归、聚类、关联规则以及在新的交互式界面上的可视化。

如果想自己实现数据挖掘算法的话，可以参考Weka的接口文档。在Weka中集成自己的算法甚至借鉴它的方法自己实现可视化工具并不是件很困难的事情。

2005年8月，在第11届ACM SIGKDD国际会议上，怀卡托大学的Weka小组荣获了数据挖掘和知识探索领域的最高服务奖，Weka系统得到了广泛的认可，被誉为数据挖掘和机器学习历史上的里程碑，是现今最完备的数据挖掘工具之一（已有11年的发展历史）。Weka的每月下载次数已超过万次。

2014年3月起，新西兰怀卡托大学将推出Weka免费网课，课程分为初级和高级两个部分，每个部分时长5周。初级课程将于2014年3月3日开课，高级课程于2014年4月下旬开课。课程具体内容参见怀卡托大学网站Weka MOOC。课程在优酷网站也有专辑。

### 2.3 处理方法

包括处理标准数据挖掘问题的所有方法：回归、分类、聚类、关联规则和属性选择。

### 2.4 输入数据

1. 通过以ARFF格式为代表的文件进行输入

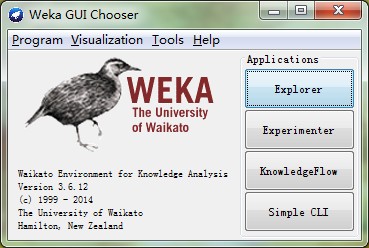
2. 直接读取数据库表

### 2.5 Weka功能简介

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。

Weka主界面为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，如图：

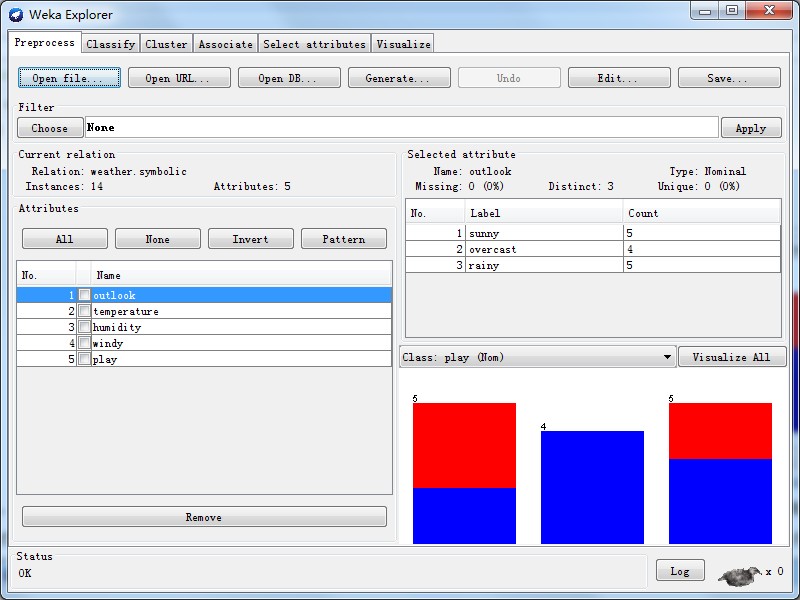
图1.1 Weka软件首页



#### 2.5.1 Explorer（探索者）

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。

图1.2 Explorer页面

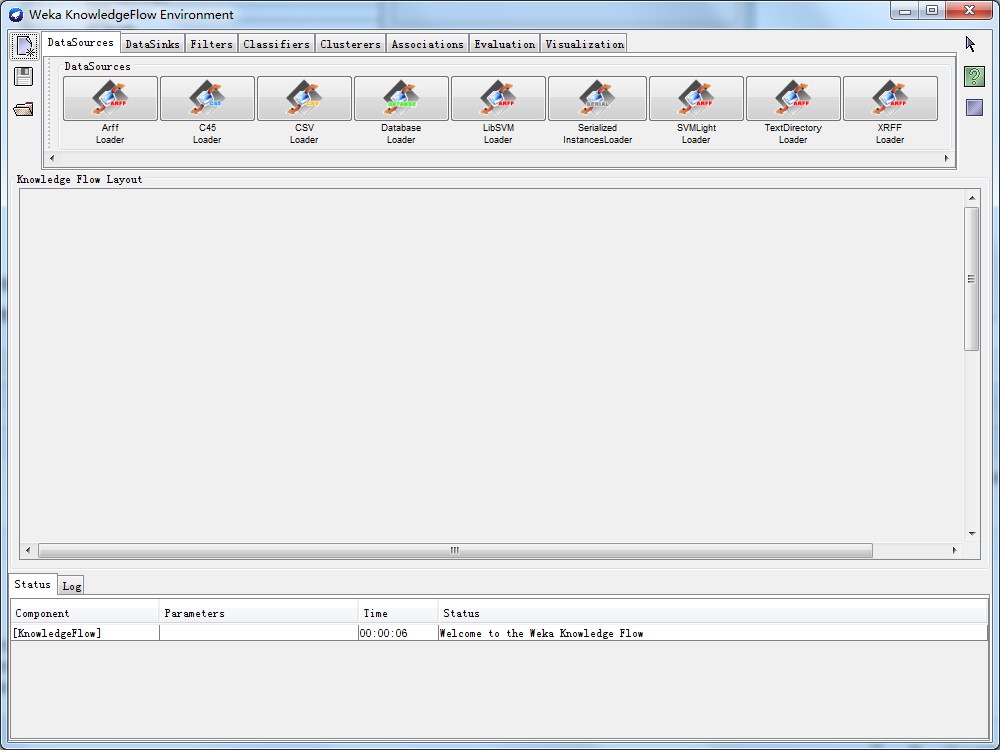


虽然探索者界面使用很方便，但它也存在一个缺陷，要求它将所需数据全部一次读进内存，一旦用户打开某个数据集，就会读取全部数据。因此，这种批量方式仅适合处理中小规模的问题。而**知识流**刚好能够弥补这一缺陷。

#### 2.5.2 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。

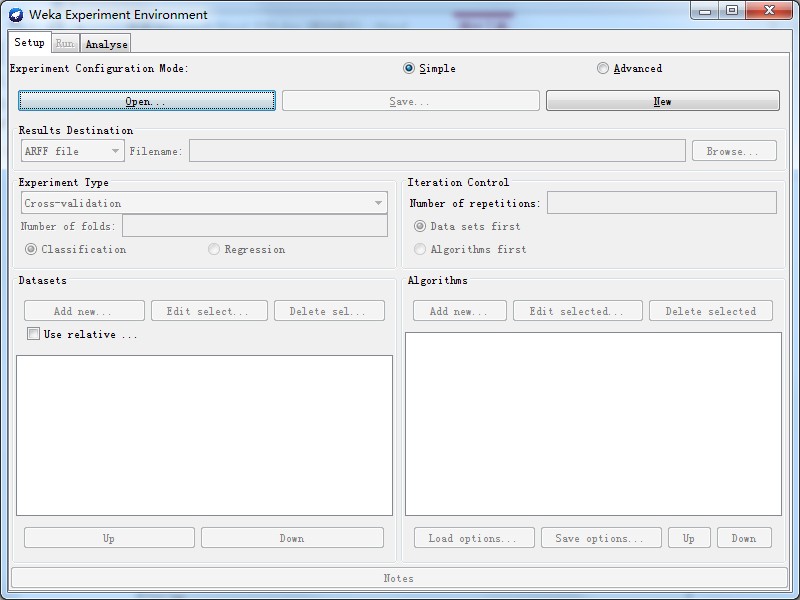
图1.3 Knowledge Flow页面



#### 2.5.3 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。

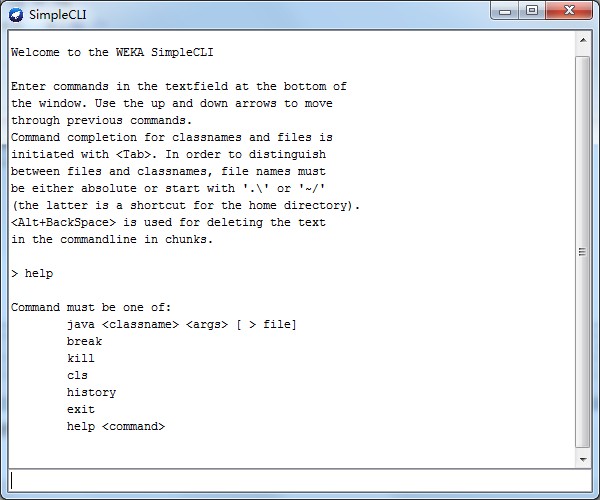
图1.4 Experimenter页面



#### 2.5.4 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]。、

图1.5 Simple CLI页面



## 3 数据集合的定义

Weka支持很多文件格式，包括arff、xrff、csv，甚至有libsvm的格式。其中arff是最常见的格式，arff是最常见的格式，我们在这里仅介绍这一种。arff全称是Attribute-Relation File Format，以下是一个arff格式的文件的例子。

文件中，“%”开头的是注释。剩余的可以分为两大部分，头信息（header information）的数据信息（data information）。

头信息中，“@relation”开头的行代表关系名称，在整个文件的第一行(除去注释)。格式是@relation;“@attribute”开头的代表特征，格式是@attribute；“@data”是实际的数据部分。每一行代表一个实例，可以认为是一个特征向量。各个特征的顺序与头信息中attribute逐个对应，特征值之间用逗号分割。在有监督分类中，最后一列是标注的结果。某些特征的数值如果是缺失的，可以用“?”代替。

Weka所处理的数据集是一个二维的表格。

### 3.1 术语

表格里的一个横行称作一个实例（Instance），相当于统计学中的一个样本，或者数据库中的一条记录。竖行称作一个属性（Attribute），相当于统计学中的一个变量，或者数据库中的一个字段。这样一个表格，或者叫数据集，在Weka看来，呈现了属性之间的一种关系（Relation）。

### 3.2 声明

声明分为关系声明和属性声明

#### 3.2.1 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）。

#### 3.2.2 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。

这些声明语句的顺序很重要。首先它表明了该项属性在数据部分的位置。例如，“humidity”是第三个被声明的属性，这说明数据部分那些被逗号分开的列中，第三[列数据](https://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=74633135&ss_c=ssc.citiao.link" \t "https://baike.sogou.com/_blank)85 90 86 96 ... 是相应的“humidity”值。其次，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

其中<nominal-specification>和<date-format>将在下面说明。还可以使用两个类型“integer”和“real”，但是Weka把它们都当作“numeric”看待。注意 “integer”，“real”，“numeric”，“date”，“string”这些关键字是区分大小写的，而“relation”, “attribute”和“data”则不区分。

### 3.3 属性

属性大致分为五类

#### 3.3.1 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。

#### 3.3.2 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-name1>,<nominal-name2>,<nominal-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。

#### 3.3.3 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

#### 3.3.4 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。

#### 3.3.5 Relational型属性

在Weka3.5版中增加了一种属性类型叫做Relational，有了这种类型我们可以像[关系型数据库](https://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=277136&ss_c=ssc.citiao.link" \t "https://baike.sogou.com/_blank)那样处理多个维度了。但是这种类型目前还不见广泛应用，暂不作介绍。

### 3.4 数据

一般的数据分为数据信息和稀疏数据

#### 3.4.1 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 LCCvsLCSH

@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"

#### 3.4.2 稀疏数据

有的时候数据集中含有大量的0值（比如购物篮分析），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, 0, Y, "class A"

0, 0, W, 0, "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起来。实例中每一个非0的属性值用<index> <空格> <value>表示。<index>是属性的序号，从0开始计；<value>是属性值。属性值之间仍用逗号隔开。这里每个实例的数值必须按属性的顺序来写，如{1 X, 3 Y, 4 "class A"}，不能写成{3 Y, 1 X, 4 "class A"}。

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。

### 3.5 csv转换为arff格式

我们通常接触到的数据为csv格式较多，若想要将数据更改为Weka通用的arff格式，我们可以利用Weka工具进行格式的转换。

#### 3.5.1 csv格式

逗号分隔值（Comma-Separated Values，CSV，有时也称为字符分隔值，因为分隔字符也可以不是逗号），其文件以纯文本形式存储表格数据（数字和文本）。纯文本意味着该文件是一个字符序列，不含必须像二进制数字那样被解读的数据。CSV文件由任意数目的记录组成，记录间以某种换行符分隔；每条记录由字段组成，字段间的分隔符是其它字符或字符串，最常见的是逗号或制表符。通常，所有记录都有完全相同的字段序列。通常都是纯文本文件。建议使用Word或是记事本来开启，再则先另存新档后用Excel开启，也是方法之一[4]。

#### 3.5.2 arff格式

arff是一种Weka专用的文件格式，由Andrew Donkin创立，有传言说arff代表Andrew's Ridiculous File Format(安德鲁的荒唐文件格式)，但在Weka的正式文档中明确说明arff代表Attribute-Relation File Format(属性——关系文件格式)。该文件是ASCII文本文件，描述共享一组属性结构的实例列表，由独立且无序的实例组成，是Weka表示数据集的标准方法，arff不涉及实例之间的关系。

#### 3.5.2 获取csv文件

本次案例我们使用UCI网站中的Soybean(Large)数据进行演示，我们先从网站中下载下来我们所需要的数据，进入网站后选择Download按钮进入下载页面，我们选择soybean-large.data数据进行下载下来发现是一个纯文本文件，用记事本打开，发现此文件大约有300多条实例，属性多达36个，为了方便用表展示，我们只截取了前7个，实例也仅选取了前10个以作示范。

将网站主页中的属性名（features）依次复制到该data文件中，属性之间用逗号隔开，这样我们就构造出来了csv格式的数据文件。保存文件，将该文件的后缀改为csv，我们就得到这个数据集的csv格式文件。

表3-1 Soybean(Large)数据集合

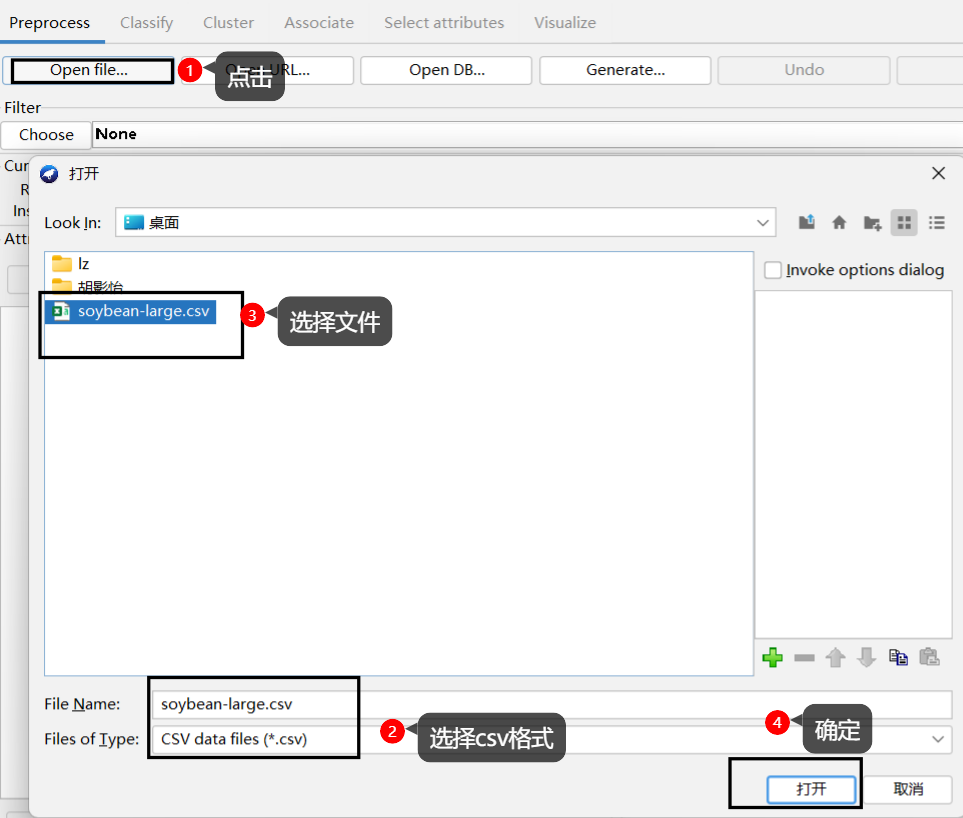
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | class | date | plant-stand | precip | temp | hail | crop-hist |
| 1 | diaporthe-stem-canker | 6 | 0 | 2 | 1 | 0 | 1 |
| 2 | diaporthe-stem-canker | 4 | 0 | 2 | 1 | 0 | 2 |
| 3 | diaporthe-stem-canker | 3 | 0 | 2 | 1 | 0 | 1 |
| 4 | diaporthe-stem-canker | 3 | 0 | 2 | 1 | 0 | 1 |
| 5 | diaporthe-stem-canker | 6 | 0 | 2 | 1 | 0 | 2 |
| 6 | diaporthe-stem-canker | 5 | 0 | 2 | 1 | 0 | 3 |
| 7 | diaporthe-stem-canker | 5 | 0 | 2 | 1 | 0 | 2 |
| 8 | diaporthe-stem-canker | 4 | 0 | 2 | 1 | 1 | 1 |
| 9 | diaporthe-stem-canker | 6 | 0 | 2 | 1 | 0 | 3 |
| 10 | diaporthe-stem-canker | 4 | 0 | 2 | 1 | 0 | 2 |
| 11 | charcoal-rot | 6 | 0 | 0 | 2 | 0 | 1 |
| 12 | charcoal-rot | 4 | 0 | 0 | 1 | 1 | 1 |
| 13 | charcoal-rot | 3 | 0 | 0 | 1 | 0 | 1 |
| 14 | charcoal-rot | 6 | 0 | 0 | 1 | 1 | 3 |
| 15 | diaporthe-stem-canker | 6 | 0 | 2 | 1 | 0 | 1 |

#### 3.5.3 转换csv格式为arff格式

1. 打开Weka，选择explorer模式

2. 这时会打开新的explorer窗口，我们选择Open File选项，可以对我们的数据进行选择。

图3.1 在Weka中打开本地数据



3. 打开文件后，我们点击save选择文件类型为arff，选择文件名称以及存放位置后，点击保存，这样我们的数据就保存为了arff格式

图3.1 在Weka中另存为arff格式数据



## 参考文献

[1] 百度百科, URL: https://baike.baidu.com/item/weka [2022.6.16].

[2] Weka 官网, URL: https://www.cs.waikato.ac.nz/ml/weka [2022-6-16].

[3] 博客, URL: http://t.zoukankan.com/chamie-p-4551514.html [2022-6-16].

[4] 博客, URL: https://baike.baidu.com/item/CSV/10739 [2022-6-18].