



D B D

Mining
大数据挖掘平台
使用手册

声明

本手册的用途在于帮助您正确地使用曙光瑞翼教育合作中心产品(以下称“本产品”),在安装和第一次使用本产品前,请您务必先仔细阅读随机配送的所有资料,特别是本手册中所提及的注意事项。这会有助于您更好和安全地使用本产品。请妥善保管本手册,以便日后参阅。

本手册的描述并不代表对本产品规格和软硬件配置的任何说明。有关本产品的实际规格和配置,请查阅相关协议、装箱单、产品规格配置描述文件,或向产品的销售商咨询。

如您不正确地或未按本手册的指示和要求安装、使用或保管本产品,或让非曙光瑞翼教育合作中心授权的技术人员修理、变更本产品,曙光瑞翼教育合作中心将不对由此导致的损害承担任何责任。

本手册中所提供照片、图形、图表和插图,仅用于解释和说明目的,可能与实际产品有些差别,另外,产品实际规格和配置可能会根据需求不时变更,因此与本手册内容有所不同。请以实际产品为准。

本手册中所提及的非曙光瑞翼教育合作中心网站信息,是为了方便起见而提供,此类网站中的信息不是曙光瑞翼教育合作中心产品资料的一部分,也不是曙光瑞翼教育合作中心服务的一部分,曙光瑞翼教育合作中心对这些网站及信息的准确性和可用性不做任何保证。使用此类网站带来的风险将由您自行承担。

本手册不用于表明曙光瑞翼教育合作中心对其产品和服务做了任何保证,无论是明示的还是默示的,包括(但不限于)本手册中推荐使用产品的适用性、安全性、适销性和适合某特定用途的保证。对本产品及相关服务的保证和保修承诺,应按可适用的协议或产品标准保修服务条款和条件执行。在法律法规的最大允许范围内,曙光瑞翼教育合作中心对于您的使用或不能使用本产品而发生的任何损害(包括,但不限于直接或间接的个人损害、商业利润的损失、业务中断、商业信息的遗失或任何其他损失),不负任何赔偿责任。

对于您在本产品之外使用本产品随机提供的软件,或在本产品上使用非随机软件或经曙光瑞翼教育合作中心认证推荐使用的专用软件之外的其他软件,曙光瑞翼教育合作中心对其可靠性不做任何保证。

曙光瑞翼教育合作中心已经对本手册进行了仔细的校勘和核对,但不能保证本手册完全没有任何错误和疏漏。为更好地提供服务,曙光瑞翼教育合作中心可能会对本手册中描述的产品软件和硬件及本手册的内容随时进行改进或更改,恕不另行通知。

目录

声明	I
目录	II
插图目录	VI
1 数据	1
1.1 文件	2
1.2 SQL 表	4
1.3 数据表	5
1.4 绘图数据	8
1.5 数据信息	10
1.6 数据采集器	12
1.7 选择属性	16
1.8 选择数据	20
1.9 分级	24
1.10 合并数据	27
1.11 连接	31
1.12 预处理	34
1.13 估算	37
1.14 离群点	40
1.15 编辑域	44
1.16 Python 脚本	46

1.17 图片浏览.....	53
1.18 颜色.....	56
1.19 连续化.....	62
1.20 离散化.....	66
1.21 特性构造函数.....	69
1.22 清除域.....	72
1.23 保存.....	76
2 可视化.....	79
2.1 分类树图.....	79
2.2 属性统计.....	84
2.3 分布.....	88
2.4 散点图.....	91
2.5 筛法图.....	99
2.6 马赛克图.....	104
2.7 线性投影.....	107
2.8 热图.....	109
2.9 维恩图.....	112
2.10 轮廓图.....	116
2.11 毕达哥拉斯树.....	119
2.12 毕达哥拉斯森林.....	124
2.13 CN2 规则查看器.....	127
2.14 散点图.....	129

3 分类.....	133
3.1 多数学习法	133
3.2 CN2 规则学习法.....	135
3.3 K 最近邻学习法	138
3.4 分类树学习法	141
3.5 随机森林.....	145
3.6 SVM 学习法	148
3.7 逻辑回归学习法.....	152
3.8 朴素贝叶斯学习法.....	154
3.9 Adaboost 学习法	157
3.10 保存分类器.....	160
3.11 加载分类器.....	162
4 回归.....	165
4.1 均值学习法	165
4.2 K 近邻学习法.....	166
4.3 回归树.....	169
4.4 随机森林回归	173
4.5 支持向量机	176
4.6 线性回归学习法.....	179
4.7 Adaboost 学习法	181
4.8 随机梯度下降学习法.....	185
4.9 多项式回归	190

5 评估.....	195
5.1 测试学习法.....	195
5.2 预测.....	200
5.3 混淆矩阵.....	204
5.4 ROC 分析.....	208
5.5 生命曲线.....	214
5.6 校准图.....	217
6 无监督.....	220
6.1 距离文件.....	220
6.2 距离矩阵.....	222
6.3 距离图.....	224
6.4 层次聚类.....	229
6.5 K 均值聚类.....	233
6.6 流形学习.....	238
6.7 主成分分析.....	242
6.8 一致性分析.....	245
6.9 示例距离.....	247
6.10 距离转换.....	250
6.11 MDS.....	253
6.12 保存距离矩阵.....	257

插图目录

1.1-1 File 窗口	3
1.2-1 SQL Table 窗口	5
1.3-1 Data Table 窗口	6
1.3-2 示例图片	8
1.4-1 Paint Data 窗口	9
1.4-2 示例图片	10
1.5-1 Data info 窗口	11
1.5-2 示例图片	12
1.6-1 Data Sampler 窗口	14
1.6-2 示例图片	15
1.6-3 示例图片	16
1.7-1 Select Columns 窗口	17
1.7-2 示例图片	19
1.7-3 示例图片	20
1.8-1 Select Rows 窗口	21
1.8-2 示例图片	23
1.8-3 示例图片	24
1.9-1 Rank 窗口	25
1.9-2 示例图片	26

1.9-3 示例图片	27
1.10-1 Merge Data 窗口	28
1.10-2 示例图片	30
1.10-3 示例图片	30
1.10-4 示例图片	31
1.11-1 Concatenate 窗口	32
1.11-2 示例图片	33
1.12-1 Preprocess 窗口	35
1.12-2 示例图片	36
1.13-1 Impute 窗口	38
1.13-2 示例图片	40
1.14-1 Outliers 窗口	42
1.14-2 示例图片	43
1.15-1 Edit Domain 窗口	44
1.15-2 示例图片	46
1.16-1 Python Script 窗口	48
1.16-2 示例图片	50
1.16-3 示例图片	51
1.16-4 示例图片	52
1.16-5 示例图片	53
1.17-1 Image Viewer 窗口	54
1.17-2 示例图片	55

1.17-3 示例图片	56
1.18-1 Color 窗口.....	57
1.18-2 Select Color 窗口.....	58
1.18-3 Color Palette 窗口.....	59
1.18-4 示例图片	61
1.18-5 示例图片	61
1.19-1 Continuize 窗口	63
1.19-2 示例图片	65
1.19-3 示例图片	66
1.20-1 Discretize 窗口	67
1.20-2 示例图片	69
1.21-1 Feature Constructor 窗口 (连续变量)	70
1.21-2 Feature Constructor 窗口 (离散变量)	71
1.21-3 示例图片	72
1.22-1 Purge Domain 窗口.....	74
1.22-2 示例图片	76
1.23-1 Save Data 窗口.....	77
1.23-2 示例图片	78
2.1-1 Tree Viewer 窗口.....	80
2.1-2 示例图片.....	82
2.1-3 示例图片.....	83
2.1-4 示例图片.....	84

2.2-1 Box Plot 窗口.....	85
2.2-2 Box Plot 窗口 (离散属性)	86
2.2-3 示例图片.....	87
2.2-4 示例图片.....	88
2.3-1 Distributions 窗口 (离散属性)	89
2.3-2 Distributions 窗口 (连续属性)	90
2.3-3 Distributions 窗口 (无类域)	91
2.4-1 Scatter Plot 窗口.....	92
2.4-2 Scatter Plot 窗口 (离散属性)	94
2.4-3 Scatter Plot 窗口 (显示分类密度)	95
2.4-4 示例图片.....	96
2.4-5 示例图片.....	97
2.4-6 示例图片.....	98
2.4-7 示例图片.....	99
2.5-1 Sieve Diagram 窗口	100
2.5-2 示例图片.....	101
2.5-3 示例图片.....	102
2.5-4 示例图片.....	103
2.5-5 示例图片.....	104
2.6-1 Mosaic Display 窗口	105
2.6-2 示例图片.....	106
2.7-1 Linear Projection 窗口.....	108

2.7-2 示例图片.....	109
2.8-1 Heat Map 窗口.....	110
2.8-2 示例图片.....	112
2.9-1 Venn Diagram 窗口.....	113
2.9-2 示例图片.....	115
2.9-3 示例图片.....	116
2.10-1 Silhouette Plot 窗口.....	117
2.10-2 示例图片.....	119
2.11-1 Pythagorean Tree 窗口.....	120
2.11-2 示例图片.....	122
2.11-3 示例图片.....	123
2.11-4 示例图片.....	124
2.12-1 Pythagorean Forest 窗口.....	125
2.12-2 示例图片.....	127
2.13-1 Rule Viewer 窗口.....	128
2.13-2 示例图片.....	129
2.14-1 Scatter Map 窗口.....	130
2.14-2 示例图片.....	132
3.2-1 CN2 Rule Induction 窗口.....	135
3.2-2 示例图片.....	137
3.2-3 示例图片.....	138
3.3-1 kNN 窗口.....	139

3.3-2 示例图片.....	140
3.3-3 示例图片.....	141
3.4-1 Tree 窗口.....	142
3.4-2 示例图片.....	143
3.4-3 示例图片.....	144
3.4-4 示例图片.....	144
3.5-1 Random Forest 窗口.....	146
3.5-2 示例图片.....	147
3.5-3 示例图片.....	148
3.6-1 SVM 窗口.....	149
3.6-2 示例图片.....	152
3.7-1 Logistic Regression 窗口.....	153
3.7-2 示例图片.....	154
3.8-1 Naive Bayes 窗口.....	155
3.8-2 示例图片.....	156
3.8-3 示例图片.....	157
3.9-1 AdaBoost 窗口.....	158
3.9-2 示例图片.....	159
3.9-3 示例图片.....	160
3.10-1 Save Model 窗口.....	161
3.10-2 示例图片.....	162
3.11-1 Load Model 窗口.....	163

3.11-2 示例图片	164
4.1-1 Mean 窗口.....	166
4.2-1 KNN 窗口.....	167
4.2-2 示例图片.....	168
4.2-3 示例图片.....	169
4.3-1 Tree 窗口.....	170
4.3-2 示例图片.....	171
4.3-3 示例图片.....	172
4.3-4 示例图片.....	172
4.4-1 Random Forest 窗口.....	174
4.4-2 示例图片.....	175
4.4-3 示例图片.....	176
4.5-1 SVM 窗口.....	177
4.6-1 Linear Regression 窗口.....	180
4.6-2 示例图片.....	181
4.7-1 AdaBoost 窗口.....	182
4.7-2 示例图片.....	184
4.7-3 示例图片.....	184
4.8-1 Stochastic Gradient Descent 窗口.....	186
4.8-2 示例图片.....	189
4.8-3 示例图片.....	190
4.9-1 Polynomial Regression 窗口.....	191

4.9-2 示例图片	192
4.9-3 示例图片	193
4.9-4 示例图片	194
5.1-1 Test & Score 窗口	196
5.1-2 Test & Score 窗口 (分类)	197
5.1-3 Test & Score 窗口 (回归)	198
5.1-4 示例图片	199
5.2-1 Predictions 窗口	200
5.2-2 示例图片	202
5.2-3 示例图片	203
5.2-4 示例图片	204
5.3-1 Confusion Matrix 窗口	205
5.3-2 示例图片	207
5.3-3 示例图片	208
5.4-1 ROC Analysis 窗口	209
5.4-2 示例图片	214
5.5-1 Lift Curve 窗口	215
5.5-2 示例图片	217
5.6-1 Calibration Plot 窗口	218
5.6-2 示例图片	219
6.1-1 Distance File 窗口	221
6.1-2 示例图片	222

6.2-1 Distance Matrix 窗口	223
6.2-2 示例图片	224
6.3-1 Distance Map 窗口	226
6.3-2 示例图片	227
6.3-3 示例图片	228
6.3-4 示例图片	229
6.4-1 Hierarchical Clustering 窗口	230
6.4-2 示例图片	232
6.4-3 示例图片	232
6.5-1 K-means 窗口	233
6.5-2 示例图片	236
6.5-3 示例图片	237
6.5-4 示例图片	238
6.6-1 Manifold Learning 窗口	239
6.6-2 示例图片	242
6.7-1 PCA 窗口	243
6.7-2 示例图片	244
6.7-3 示例图片	245
6.8-1 Correspondence Analysis 窗口	246
6.8-2 示例图片	247
6.9-1 Distances 窗口	248
6.9-2 示例图片	250

6.10-1 Distance Transformation 窗口.....	251
6.10-2 示例图片	252
6.11-1 MDS 窗口.....	254
6.11-2 示例图片	257
6.12-1 Save Distance Matrix 窗口	258
6.12-2 示例图片	259

1 数据

 文件	 SQL 表	 数据表	 绘图数据
 数据信息	 数据采集器	 选择属性	 选择数据
 分级	 合并数据	 连接	 预处理
 估算	 离群点	 编辑域	 Python 脚本
 图片浏览	 颜色	 连续化	 离散化
 特性构造函数	 清除域	 保存	

1.1 文件

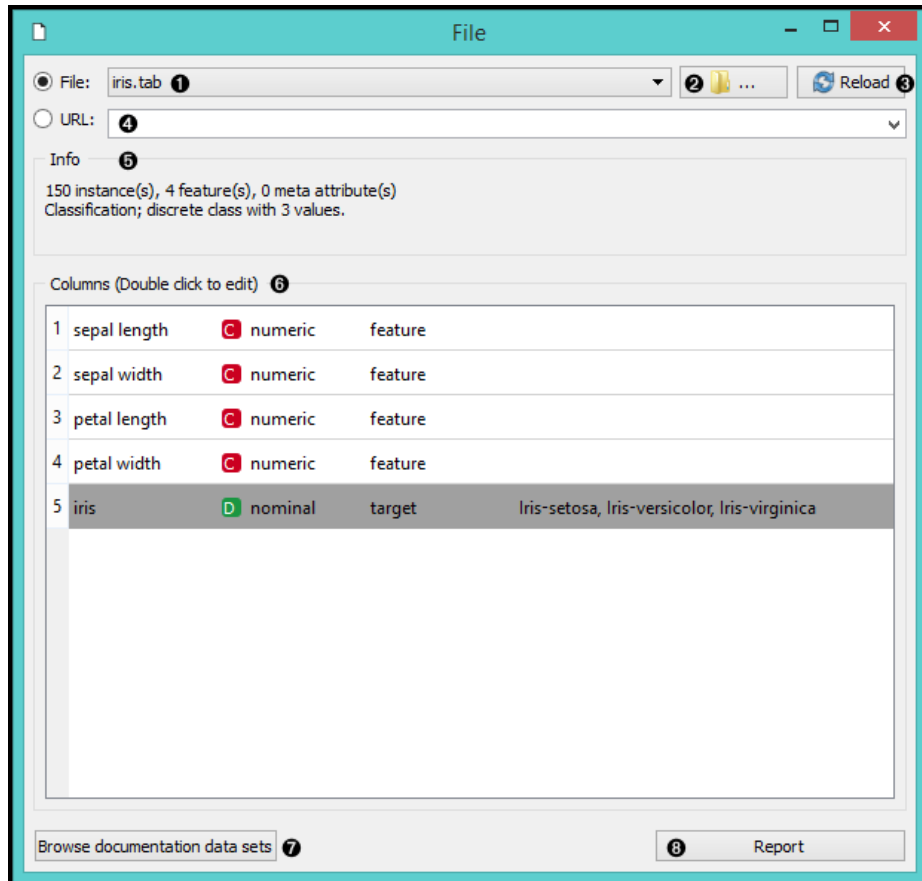


从输入文件读取属性值数据

1.1.1 描述

文件 (File) 组件读取输入数据文件 (具有数据实例的数据表) 并将数据集发送到它的输出通道。它保留最近打开的文件的历史记录。为了方便起见,历史记录还包含一个与 Mining 一起预先安装的带有样本数据集的目录。

这个组件可以从 Excel 文件, txt 文件, csv 格式文件或者通过 URL 来读取数据。

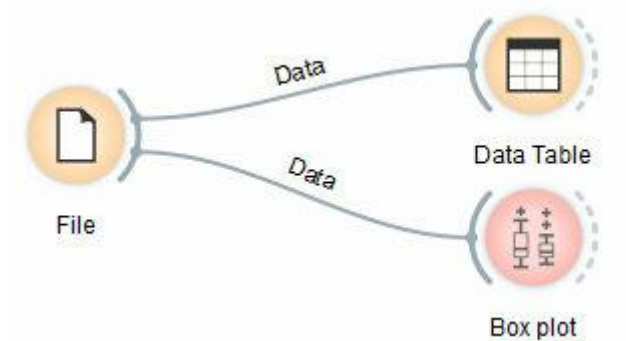


1.1-1 File 窗口

1. 浏览之前打开的数据文件，或加载任何样本数据文件。
2. 浏览数据文件。
3. 重新加载当前选择的数据文件。
4. 通过 URL 查找文件。
5. 有关加载的数据集的信息（数据集大小、数据特征的数量及类型）。
6. 文件列表（双击打开编辑）。
7. 浏览之前保存的文档。
8. 添加有关数据集信息（大小、特征）的报告。

1.1.2 示例

大多数 Mining 工作流程可能都从文件 (File) 组件开始。在下面的方案中，这个组件用于读取发送到 Data Table 组件以及显示 Box plot 的组件的数据。



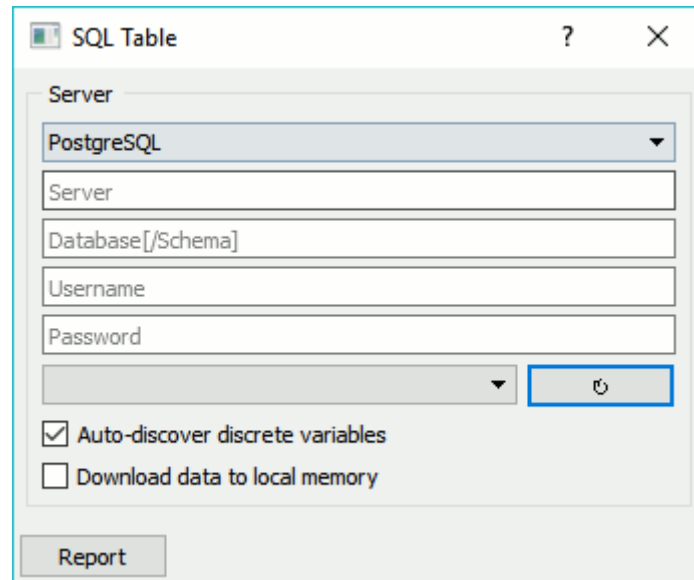
1.2 SQL 表



从 SQL 数据库读取数据

1.2.1 描述

SQL 组件访问存储在 SQL 数据库中的数据。它可以连接到 PostgreSQL (需要 psycopg2 模块) 或 SQL Server (需要 pymssql 模块)。



1.2-1 SQL Table 窗口

1.3 数据表



采用电子表格显示属性值数据

1.3.1 描述

数据表 (Data Table) 组件在其输入上接收一个或多个数据集并采用电子表格的形式呈现它们。数据实例可以按属性值进行排序。这个组件还支持手动选择数据实例。

The screenshot shows the 'Data Table' component window. On the left, there is a control panel with the following sections:

- Info:** 150 instances (no missing values), 4 features (no missing values), Discrete class with 3 values (no missing values), No meta attributes.
- Variables:**
 - Show variable labels (if present)
 - Visualize continuous values
 - Color by instance classes
- Selection:**
 - Select full rows
- Buttons:** Restore Original Order, Report, Send Automatically (checked).

The main table displays data for the Iris dataset, sorted by 'sepal length'. The columns are: iris, sepal length, sepal width, petal length, and petal width. Each numerical value is accompanied by a horizontal bar chart. The bars are color-coded by instance class: green for 'Iris-virginica' and red for 'Iris-versicolor'. The table shows 15 rows of data, with the first row (111) being 'Iris-virginica' and the last row (121) being 'Iris-virginica'.

	iris	sepal length	sepal width	petal length	petal width
111	Iris-virginica	6.500	3.200	5.100	2.000
117	Iris-virginica	6.500	3.000	5.500	1.800
148	Iris-virginica	6.500	3.000	5.200	2.000
59	Iris-versicolor	6.600	2.900	4.600	1.300
76	Iris-versicolor	6.600	3.000	4.400	1.400
66	Iris-versicolor	6.700	3.100	4.400	1.400
78	Iris-versicolor	6.700	3.000	5.000	1.700
87	Iris-versicolor	6.700	3.100	4.700	1.500
109	Iris-virginica	6.700	2.500	5.800	1.800
125	Iris-virginica	6.700	3.300	5.700	2.100
141	Iris-virginica	6.700	3.100	5.600	2.400
145	Iris-virginica	6.700	3.300	5.700	2.500
146	Iris-virginica	6.700	3.000	5.200	2.300
77	Iris-versicolor	6.800	2.800	4.800	1.400
113	Iris-virginica	6.800	3.000	5.500	2.100
144	Iris-virginica	6.800	3.200	5.900	2.300
53	Iris-versicolor	6.900	3.100	4.900	1.500
121	Iris-virginica	6.900	3.200	5.700	2.300

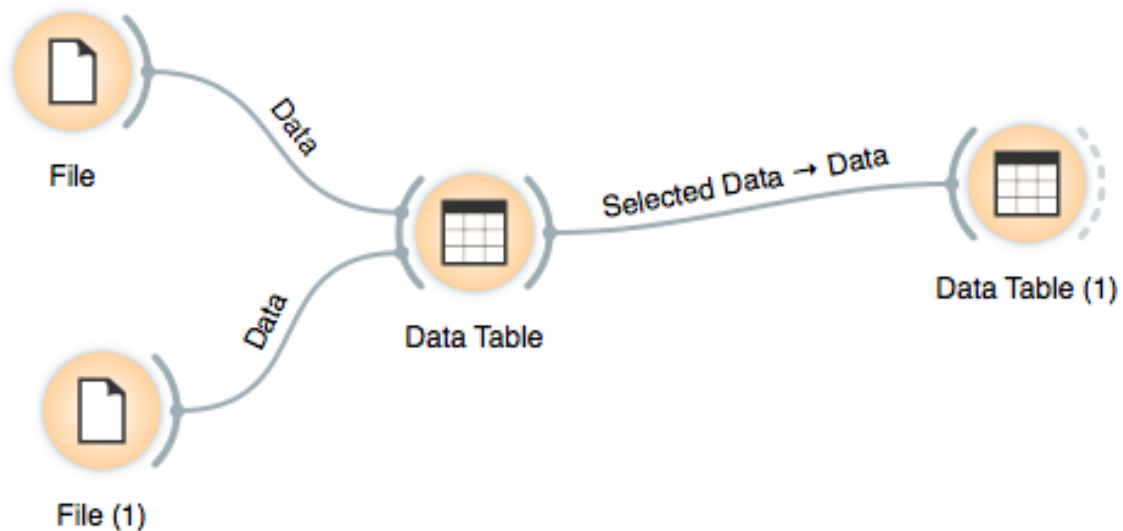
1.3-1 Data Table 窗口

1. 带有包含数据实例 (在行中) 及其属性值 (在列中) 的表的数据集的名称 (例如, 从输入数据文件的名称继承的名称)。在所示的示例中, 数据集按属性 “sepal length” 排序。
2. 有关当前数据集大小以及属性数量及类型的信息。
3. 连续属性的值可以通过下划线直观的表现出来, 不同的类可以设置不同的颜色。

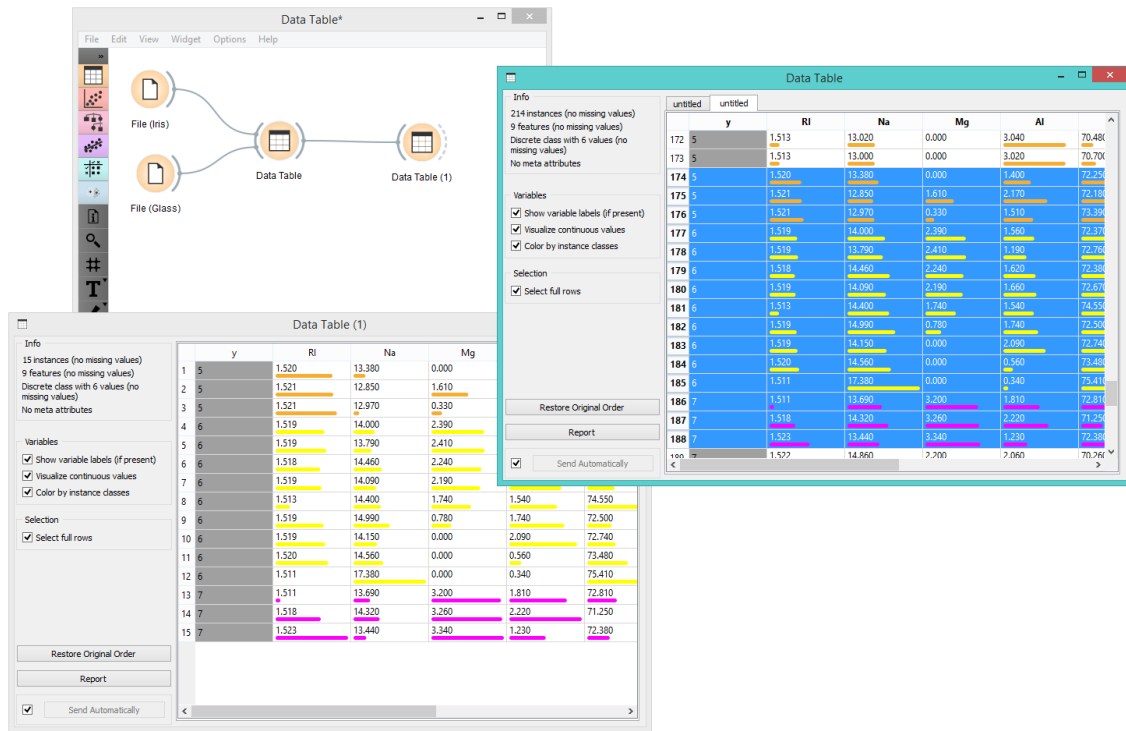
4. 数据实例（行）可以被选择并发送到组件的输出通道。
5. 在基于属性的分类之后，使用 Restore Original Order 按钮来重新排序数据实例。
6. 向当前报告中添加一个包含整个数据表的条目。
7. 在选项框打钩来自动地向其他组件提交更改。或者点击 send 来提交。

1.3.2 示例

我们使用了两个 File 组件，读取了 Iris 和 Glass 数据集（在 Mining 分布中提供）并将它们发送到数据表（Data Table）组件。



在第一个数据表（Data Table）中选择的数据实例传递到第二个数据表（Data Table）。注意到，我们可以选择要查看的数据集（Iris 或 Glass）；如果选择“Send automatically”将一个数据集更改为另一个数据集会改变该数据实例的通信选择。



1.3-2 示例图片

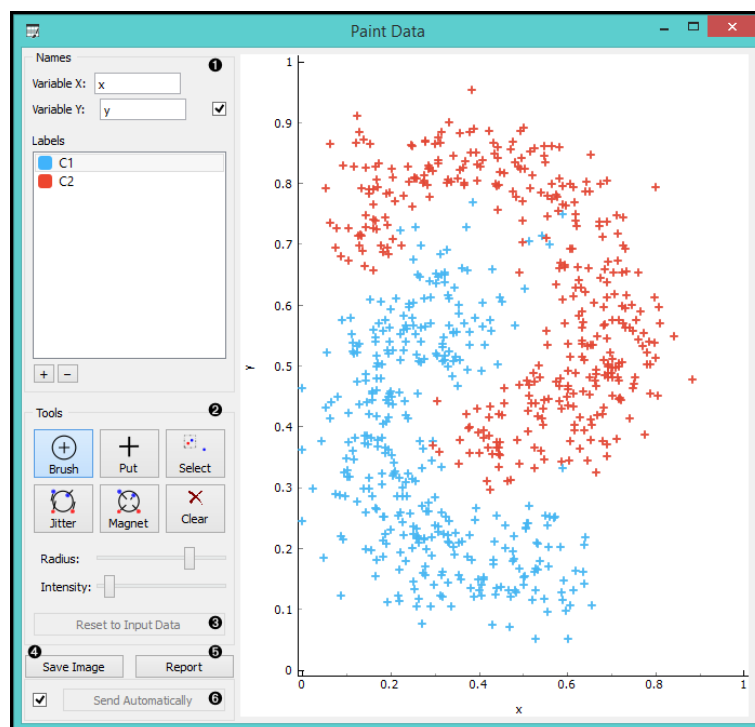
1.4 绘图数据



在 2D 平面上绘制数据。放置各个数据点或使用画刷绘制较大的数据集

1.4.1 描述

这个组件支持通过直观地将数据点放置在二维平面上来创建新的数据集。数据点可以逐个放置在平面上 (Put) , 也可以通过刷新大批量地放置 (Brush)。如果数据旨在用于监督学习 , 则数据点可以属于类。

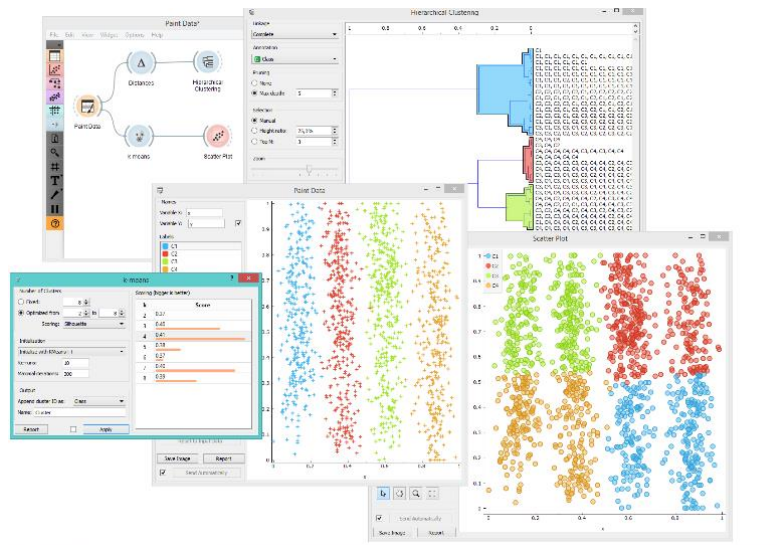


1.4-1 Paint Data 窗口

1. 设置纵横坐标变量。添加或删除类。
2. 绘图组件。用 Brush 和 Put 绘制数据点。用 Select 选择 (然后删除或重新定位) 数据点。用 Jitter 或 Magnet 重新定位数据点。用 Clear 清除数据。
3. 重新输入数据。
4. 保存图片。
5. 添加有关数据集信息 (大小、特征) 的报告。
6. 在选项框打钩来自动地向其他组件提交更改。或者点击 send 来提交。

1.4.2 示例

在下面的示例中，我们绘制了一个具有 4 个类的数据集。这样的数据集更擅长于展示 K-means 算法和层次聚类算法。在下图中，我们可以看到，总的来说，K-means 算法在识别集群上要优于层次聚类算法。它返回一个得分等级，其中最好的分数（最大值的那个）意味着最有可能的几组数值。然而，层次聚类算法不能很好地把它们归类出来。总的来说，这是一个学习和探索统计知识的好组件



1.4-2 示例图片

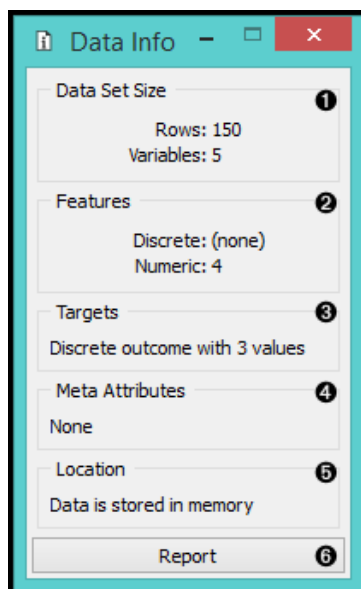
1.5 数据信息



显示所选数据集的信息

1.5.1 描述

一个简单的组件，提供有关数据集大小，特征，目标，元属性和位置的信息。



1.5-1 Data info 窗口

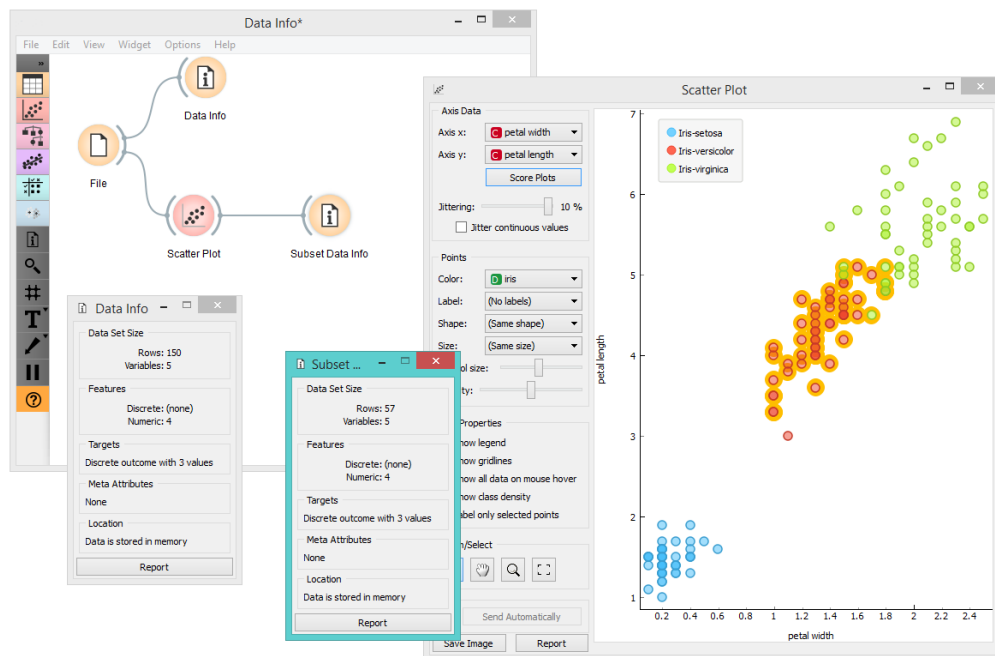
1. 数据集大小信息。
2. 离散和连续特征信息。
3. 目标信息。
4. 元属性信息。

5. 数据存储位置的信息。

6. 制作报告。

1.5.2 示例

下面，我们比较两个 Data Info 组件的基本统计信息，一个是关于整个数据集的信息，另一个与 Scatter plot 组件中的（手动）选定子集的信息。我们使用 Iris 数据集。



1.5-2 示例图片

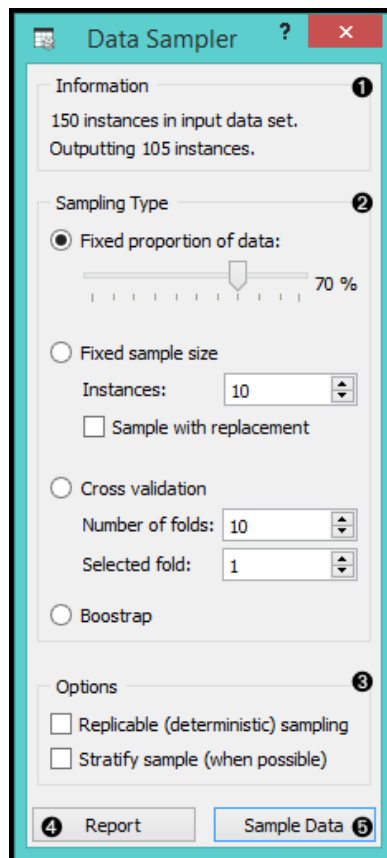
1.6 数据采集器



从输入数据集中选择数据实例的一个子集

1.6.1 描述

数据采样器 (Data Sampler) 实现采用多种方法从输入通道采集数据。它输出采样的数据集和补充数据集 (具有输入集中未包含在采样数据集中的实例)。当提供输入数据集并且按下 Sample Data 之后才设置输出。



1.6-1 Data Sampler 窗口

1. 有关输入和输出数据集的信息。

2. 期望采样法：

数据固定比例 (Fixed proportion of data) 返回整个数据的一个选定的百分比 (例如所有数据的 70%)

固定样本大小 (Fixed sample size) 返回一个选定的数据实例的值，并且可以设置样品与置换 (Sample with replacement)，这个总是从整个数据集中选取样品 (不取出已经在子集中的例子)。

交叉验证 (Cross Validation) 把数据实例分割成互补的子集，你可以选择分割 (子集) 的数目，并且作为样本使用。

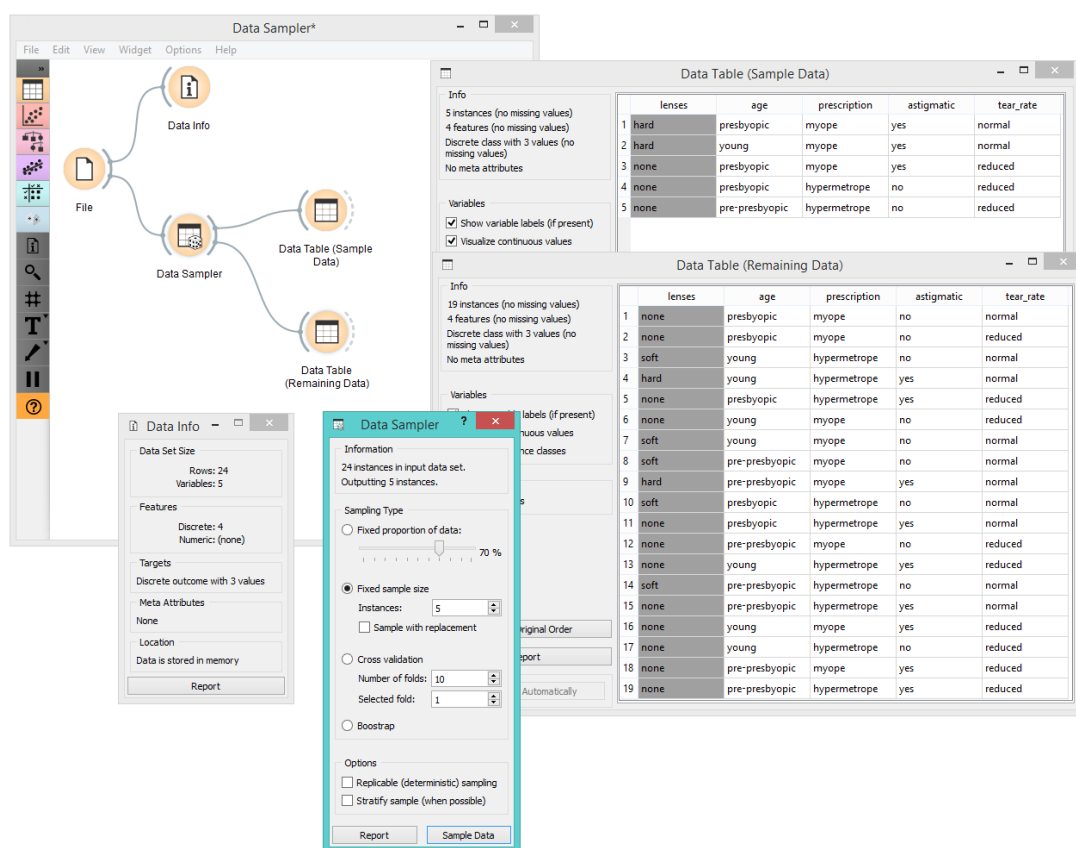
3. 重复抽样 (Replicable sampling) 保持采样模式，就可以通过用户，而分层模拟输入数据集的组成。

4. 制作报告。

5. 按照抽样资料 (Sample data) 来输出数据样本。

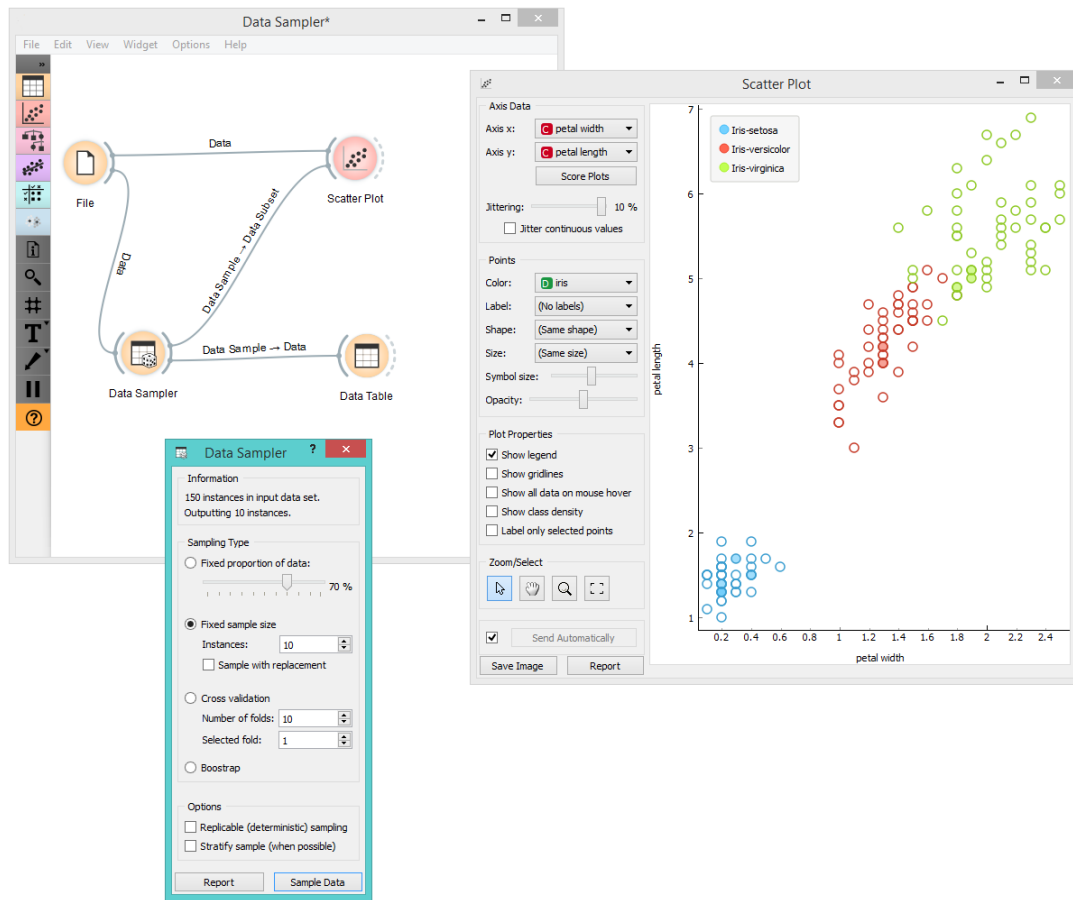
1.6.2 示例

首先，让我们看看数据采样器 (Data Sampler) 是怎样工作的。从 Data Info 组件中的原始数据集的信息可以看到数据中有 24 个实例 (在 lenses.tab 文件中)。我们使用 Data Sampler 组件来抽取样本，从而选择了固定样本大小的 5 个简单的实例。我们可以在 Data Table 组件中观察采样数据。第二个 Data Table 则显示剩余的不在样本中的 19 个实例。



1.6-2 示例图片

在下面的工作流程方案中，我们从 Iris 数据集中采集了 10 个数据实例并将原始数据和样本发送给散点图组件。我们用实心圆圈绘制了采集的数据实例。



1.6-3 示例图片

1.7 选择属性

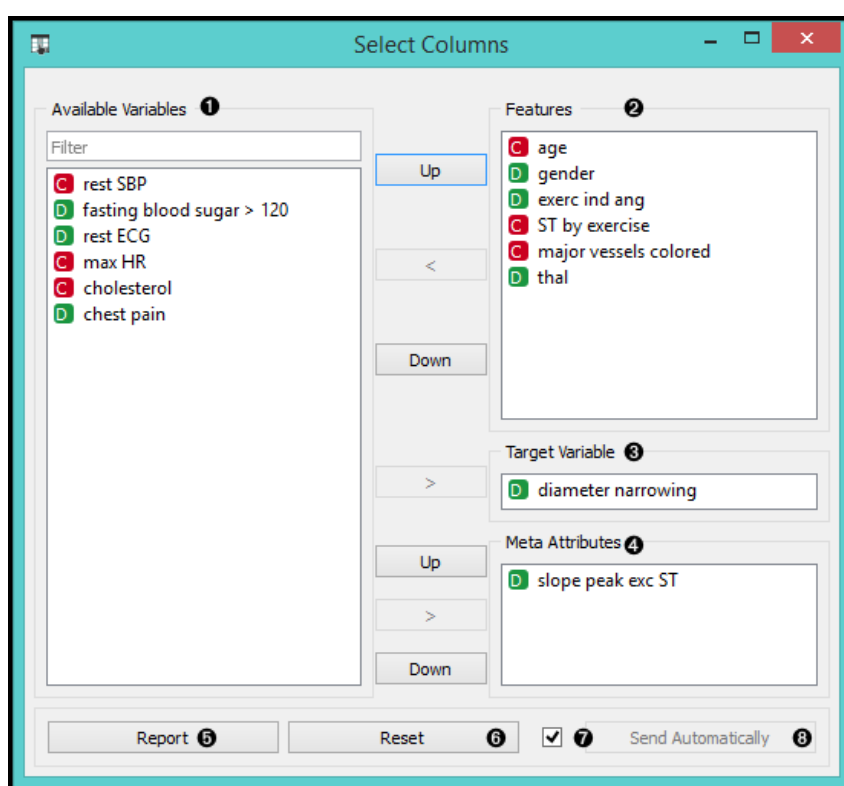


手动选择属性数据以及数据域的组成

1.7.1 描述

选择属性 (Select Attributes) 组件用于手动构建您的数据域。用户可以决定使用哪些属性以及如何使用。Mining 对普通属性、(可选)类属性和元属性进行了区分。例如,对于构建分类模型,那么域将由一组属性和一个离散类属性组成。建模时不使用元属性,但多个组件可以使用元属性为实例提供可选标签。

Mining 属性是类型化的属性,并且也是离散、连续或字符串。属性类别通过属性名称前的符号来标记(分别为 D、C、S)。



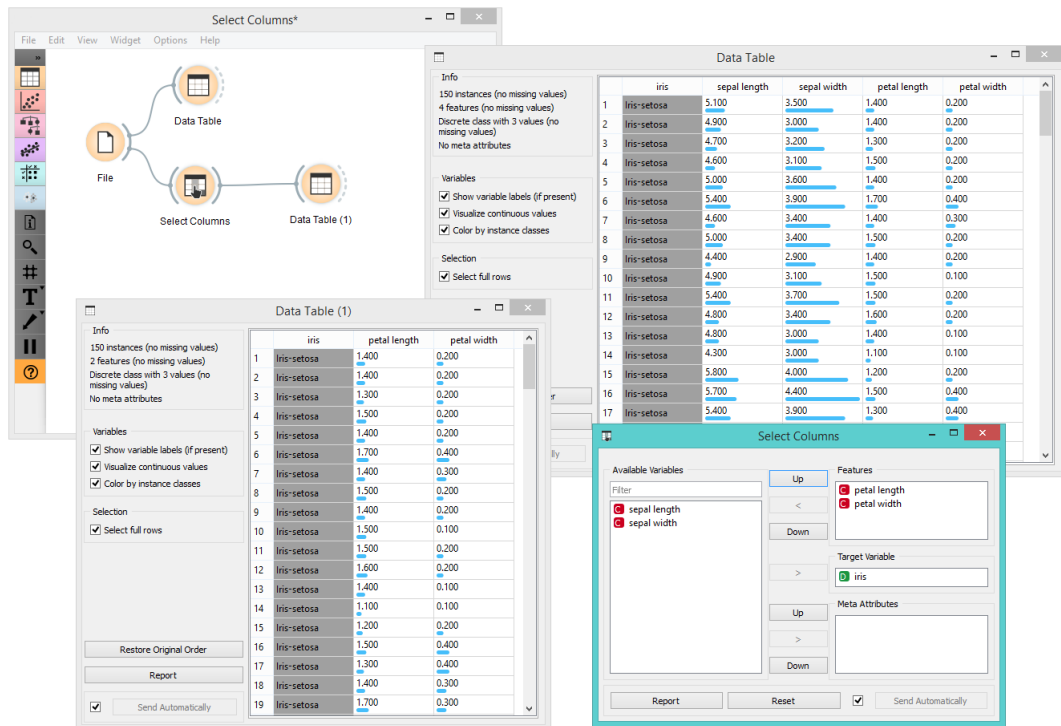
1.7-1 Select Columns 窗口

1. 删去输入数据文件中将不在输出数据文件的数据域中的数据属性。
2. 新的数据文件中的数据属性。
3. 目标变量。如果没有,那么新的数据集将是没有目标变量的数据集。

4. 新的数据文件的元属性。这些属性包含在数据集中，但对于大多数方法来说，在数据分析时不考虑这些属性。
5. 将域数据组成上的条目添加到当前报告。
6. 将域组合重置为输入数据文件的域组合。
7. 如果您希望自动应用数据域的更改，就在选项框中打钩。
8. 应用数据域的更改并发送新的数据文件到组件的输出通道。

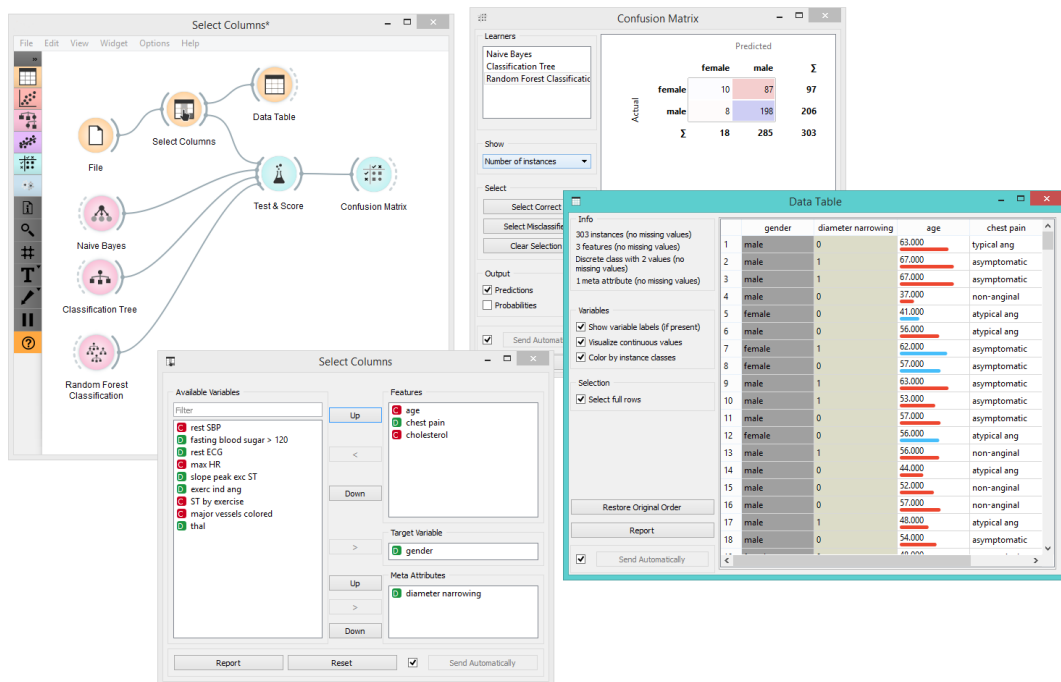
1.7.2 示例

在下面的工作流程中，File 组件中的 iris 数据进入到 Select Columns 组件，在这里我们选择只输出两个属性（即 petal width 和 petal length）。我们同时观察原始数据集和 Data Table 组件中选定列的数据集。



1.7-2 示例图片

对于这个组件的更复杂的使用，我们组建了一个工作流程来定义心脏疾病数据集的分类问题。最初，这项任务是用来预测病人是否有冠状动脉直径变窄的问题。我们将此问题更改为基于年龄、胸痛和胆固醇水平的性别分类问题，并通过这些信息将直径变窄属性保持为元属性。



1.7-3 示例图片

1.8 选择数据



根据数据特征上的条件选择数据实例

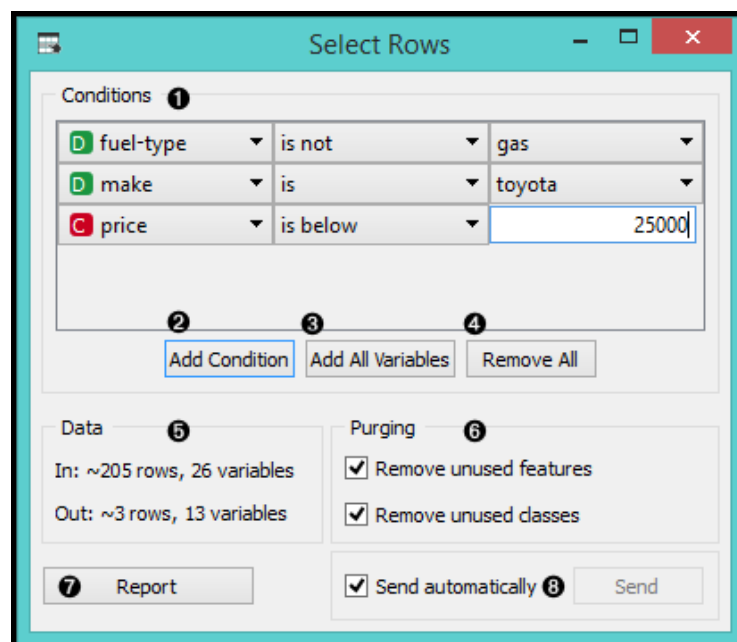
1.8.1 描述

20 · 数据中国“百校工程”

这个组件允许用户根据一组数据属性上定义的条件从输入数据集中选择数据的一个子集。与选择规则匹配的数据实例放置在输出 Matching Data 通道上。

数据选择的条件采用析取范式的形式给出，作为合取数据（换句话说，所选择的项目是与条件中所有术语匹配的项目）。

定义条件术语的方法是：选择属性，从适用于属性类型的运算符列表中选择运算符，如果需要的话，还要定义要在条件术语中使用的值。离散、连续和字符串属性的运算符是不同的。



1.8-1 Select Rows 窗口

1. 你想要应用的条件以及他们的算子和相关的值。
2. 向条件列表中添加新的条件。
3. 立即添加所有可能的变量。
4. 立即删除所有列出的变量。

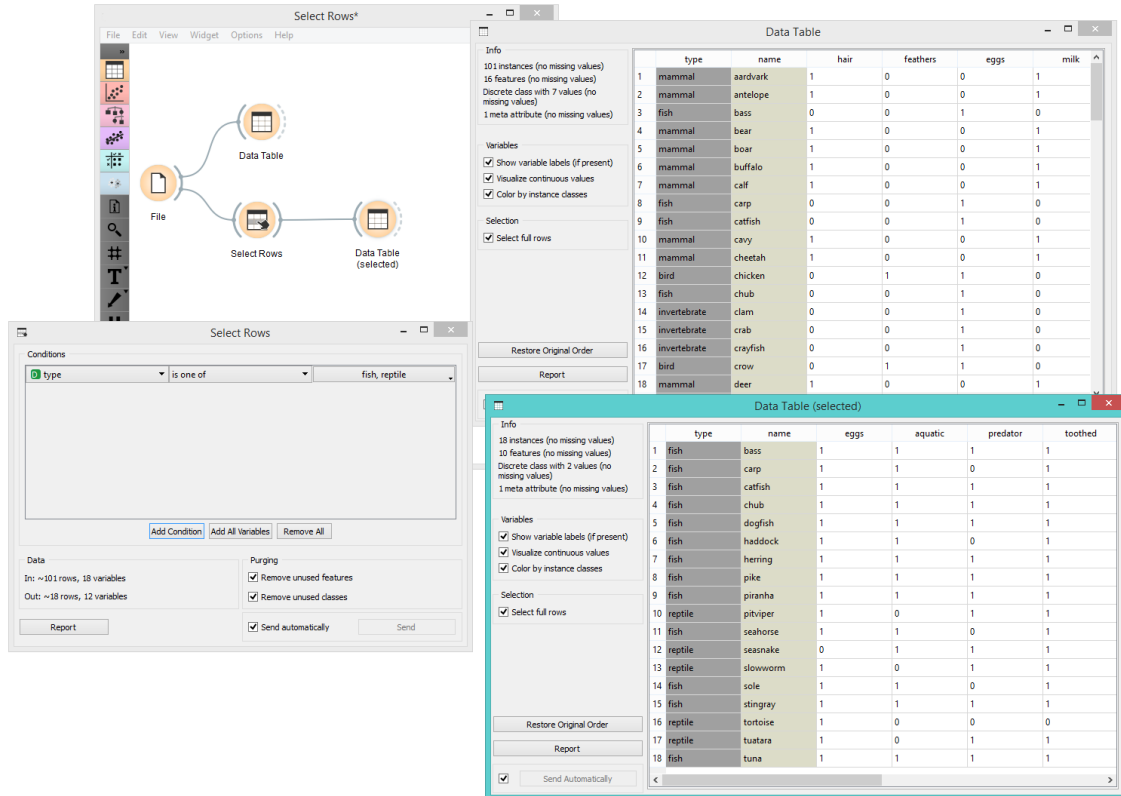
5. 输入数据集的信息和与条件匹配的实例的信息。
6. 清除一些类型的输出数据。
7. 制作一个报告。
8. 当 Send automatically 的选项框被打钩时，所有的变更会自动的传达给其他的组件。

注意,条件组合的任何更改都会触发显示所选择的数据实例数量的信息窗格 (Data Out) 的更新。

如果选中 Send automatically，则当条件组合或任何术语发生任何更改时会更新输出。

1.8.2 示例

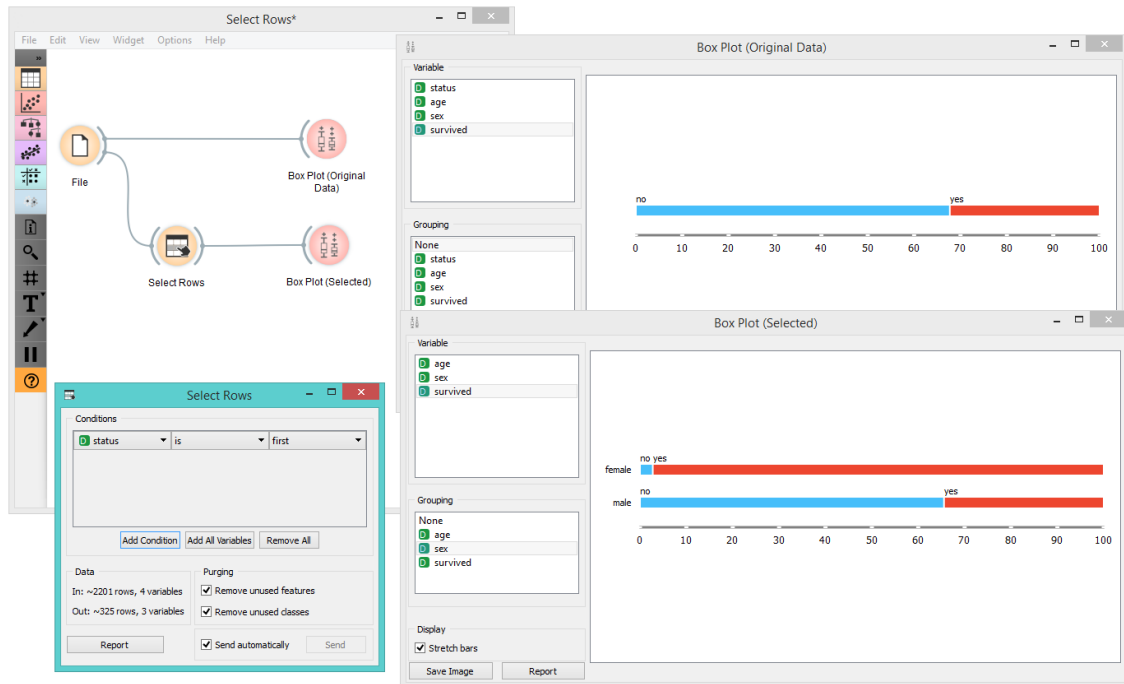
在下面的工作流程中,我们使用 File 组件的 Zoo 数据,并把它提交到 Select Rows 组件里。在这个组件里,我们选择只输出两种动物类型:鱼类和爬行动物类。我们可以在这个组件里同时观察原始数据集和所挑选类型的数据集。



1.8-2 示例图片

在下个例子中，我们使用 Titanic 数据集中的数据，类似的将它添加进 Box Plot 组件中。

我们首先观察生存类型的所有数据集。然后我们在 Select Rows 组件中只选择头等舱的乘客，并再一次把它添加到 Box Plot 组件中。这样我们可以根据性别分组观察到头等舱乘客的生存率。



1.8-3 示例图片

1.9 分级

213

对分类或回归数据集中属性进行分级

1.9.1 描述

24 · 数据中国“百校工程”

分级 (Rank) 组件考虑类标签数据集 (分类或回归) 并且根据它们与该类的相关性给属性评分。

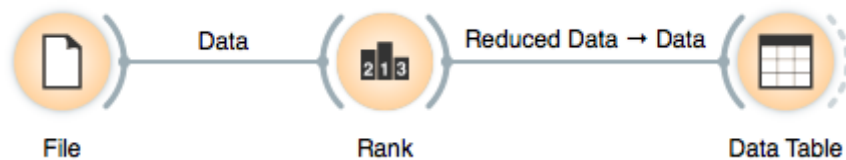
	#	Inf. gain	Gain Ratio	Gini	ANOVA	Chi2	ReliefF	FCBF
C petal length	C	1.112	0.557	0.217	847.977	76.218	0.409	0.618
C petal width	C	1.077	0.541	0.208	764.858	71.357	0.414	0.599
C sepal length	C	0.549	0.276	0.110	78.627	45.082	0.138	0.000
C sepal width	C	0.375	0.191	0.076	33.663	31.390	0.135	0.212

1.9-1 Rank 窗口

1. 从数据表中选择属性。
2. 属性 (行) 及其不同评分方法的得分 (类)。
3. 向当前报告中添加分数表。
4. 如果这项选中, 该组件会自动向其他组件传输更改的数据。

1.9.2 示例: 属性排名和选择

下面我们在紧接着 File 组件之后使用，用于减少数据属性集，使其只包含信息最丰富的属性：



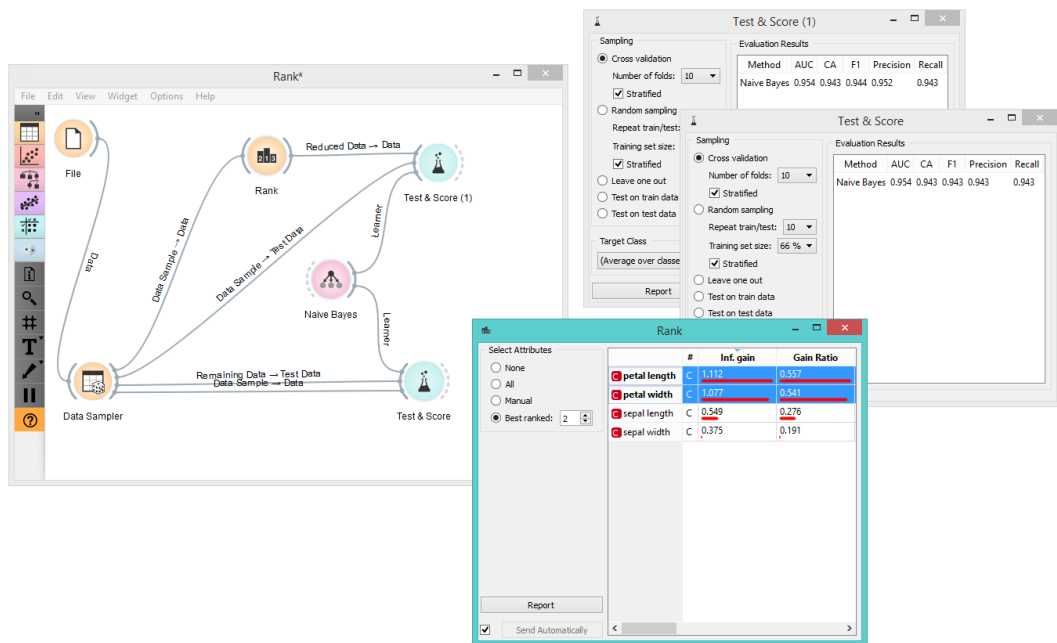
注意，这个组件如何输出只包含分数最佳属性的数据集：

Feature	#	Inf. gain	Gain Ratio	Gini
thal	3	0.208	0.167	0.068
chest pain	4	0.205	0.118	0.067
major vessels colored	C	0.180	0.115	0.059
ST by exercise	C	0.145	0.074	0.047
exerc ind ang	2	0.139	0.153	0.046
max HR	C	0.123	0.062	0.040
slope peak exc ST	3	0.112	0.087	0.038
age	C	0.058	0.029	0.020
gender	2	0.057	0.063	0.019
rest ECG	3	0.024	0.022	0.008
cholesterol	C	0.016	0.008	0.006
rest SBP	C	0.015	0.008	0.005
fasting blood sugar > 120	2	0.000	0.001	0.000

1.9-2 示例图片

1.9.3 示例：机器学习的特征子集选择

下面是一个比较复杂的示例。在下面的工作流程中，我们首先将数据拆分成训练集和测试集。在上面的分支中，训练数据通过 Rank 组件传递，以选择信息最丰富的属性，而在下面的分支中，没有选择任何特征。所选择的特征以及原始数据集传递到它自己的 Test & Score 组件，这个组件会连接一个朴素贝叶斯分类器并在测试集上为该分类器评分。



1.9-3 示例图片

对于具有很多特征以及朴素贝叶斯分类器特征选择的数据集，如上所示，通常预测准确性会更好。

1.10 合并数据

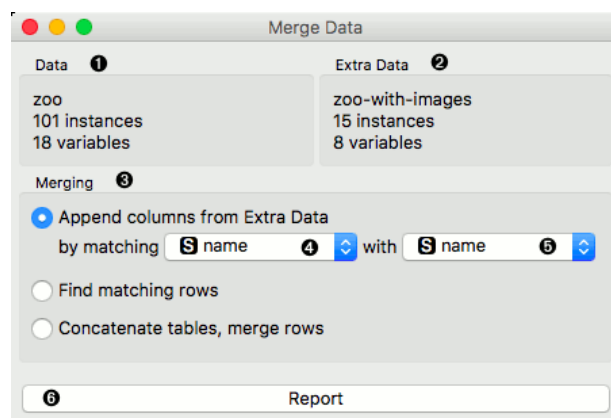


基于所选属性的值合并两个数据集

1.10.1 描述

合并数据 (Merge Data) 组件用于根据所选属性的值水平合并两个数据集。在输入时，需要两个数据集，data 和 extra data。这个组件允许从每个域中选择将用于执行合并的属性。选择之后，这个组件产生一个输出。它对应于从输入数据到被添加输入的额外数据的实例。

合并由选定(合并)属性的值完成。首先，从 Data 里面获取合并属性的值，然后从 Extra Data 里面搜索匹配的值。如果从 Extra Data 里找到了不止一个单实例，则从可用的合并属性中移除该属性。



1.10-1 Merge Data 窗口

1. Data 的信息。
2. Extra Data 的信息。
3. 合并类型。

4. Data 中可比较的列表。

5. Extra Data 中可比较的列表。

6. 制作一份合并数据的报告。

1.10.2 示例

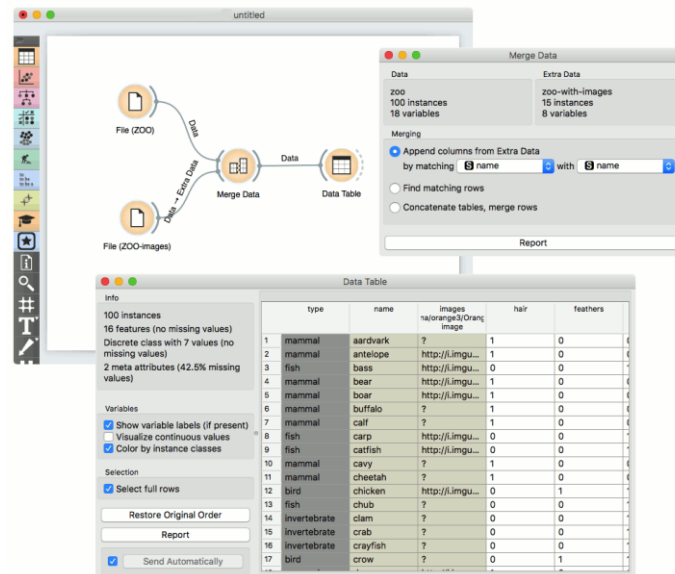
合并两个数据集，结果将根据选定的公共属性向原始文件添加新属性。在下面的例子里，我们希望合并只包含实际数据的 zoo.tab 文件和只包含图片的 zoo-with-images.tab 文件。两个文件共享一个公共字符串属性名称。现在，我们创建一个连接两个文件的工作流。zoo.tab 连接到 Merge Data 的输入数据（Data），而 zoo-with-images.tab 连接到额外输入数据（Extra Data）。Merge Data 组件的输出数据会被连接到 Data Table 组件。在后者中显示出 Merge Data 的输出，其中图像文件已经被加进原始文件了。

The screenshot shows the Orange3 interface with a workflow and two dialog windows. The workflow consists of 'File (ZOO)' and 'File (ZOO-images)' connected to 'Merge Data', which is then connected to 'Data Table'. The 'Merge Data' dialog is open, showing 'Data' as 'zoo' (100 instances, 18 variables) and 'Extra Data' as 'zoo-with-images' (15 instances, 8 variables). The 'Merging' section has 'Find matching rows' selected, with 'where name equals name'. The 'Data Table' window shows the following data:

	type	name	images na/orange3/Oran image	hair	feathers
1	mammal	antelope	http://i.lingu...	1	0
2	fish	bass	http://i.lingu...	0	1
3	mammal	bear	http://i.lingu...	1	0
4	mammal	boar	http://i.lingu...	1	0
5	fish	carp	http://i.lingu...	0	1
6	fish	cattfish	http://i.lingu...	0	1
7	bird	chicken	http://i.lingu...	0	1
8	mammal	deer	http://i.lingu...	1	0
9	mammal	dolphin	http://i.lingu...	0	0
10	bird	duck	http://i.lingu...	0	1
11	bird	gull	http://i.lingu...	0	1
12	fish	haddock	http://i.lingu...	0	1
13	mammal	hamster	http://i.lingu...	1	0
14	bird	kiwi	http://i.lingu...	0	1
15	mammal	mink	http://i.lingu...	1	0

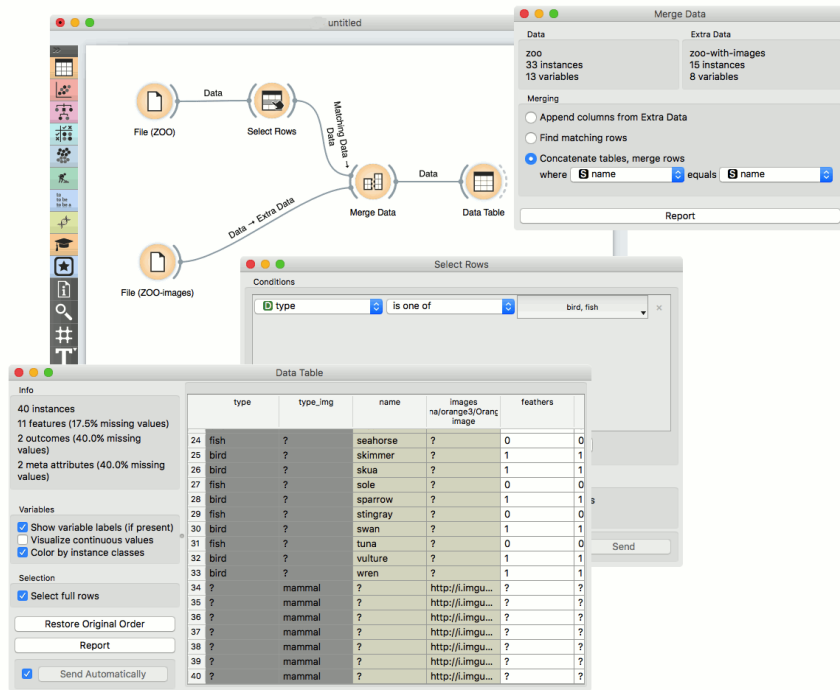
1.10-2 示例图片

我们希望在输出中包含所有实例的情况，即使没有找到与属性名称匹配的情况，也将在下面的工作流程中显示。



1.10-3 示例图片

第三种合并类型显示在下一个工作流程中。输出由两个输入组成，未分配的值在未匹配的情况下分配。



1.10-4 示例图片

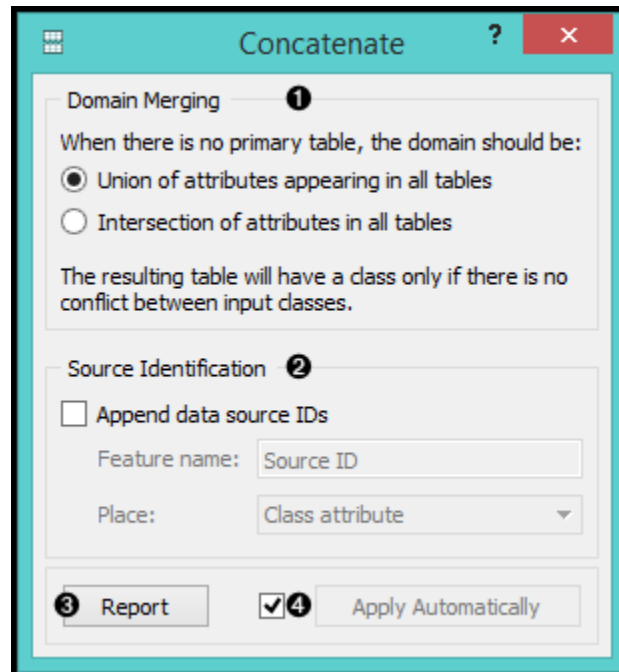
1.11 连接



连接来自多个源的数据

1.11.1 描述

这个组件连接多个示例集。这种合并是“垂直的”，这意味着如果有分别含有 10 个示例和 5 个示例的两个示例集，则会产生一个含有 15 个示例的示例集。



1.11-1 Concatenate 窗口

1. 设置属性合并的方法。
2. 将源数据集的标识添加到输出数据集。
3. 制作一份连接数据的报告。
4. 如果 Apply automatically 被勾选了，更改会自动传输。否则，需要点击 Apply。

如果一个表作为主表连接到这个组件，那么得到的表将包含这些相同的属性。如果没有主表，那么属性可能是指定为“Additional Tables”的表中所出现的所有属性的并集，或者它们的交集，即，出现在所有连接表中的属性列表。

1.11.2 示例

如下图所示，该部件可用于从两个单独的文件合并数据。假设我们有两个具有相同属性的数据集，一个包含来自第一个实验的实例和另一个包含第二个实验的实例，我们希望把两个数据集合并到一起。我们使用 Concatenate 组件，根据属性来合并数据（在现有的属性下添加新的行）。

接下来，我们使用一个修正过的 Zoo 数据集。在第一个 File 组件里，我们只加载首字母是 A 和 B 的动物，在第二个里只加载首字母是 C 的动物。连接后，我们可以在 Data Table 组件的新数据集中看到首字母从 A 到 C 的完整的动物列表。

The screenshot shows the Orange3 software interface. The main window displays a workflow with the following components:

- Concatenate** widget: Receives input from two Data Table widgets (A+B and C) and outputs to a Data Table widget (A+B+C).
- Data Table (A+B)** widget: Displays a table with 6 instances (aardvark, antelope, bear, boar, buffalo).
- Data Table (C)** widget: Displays a table with 11 instances (calf, carp, catfish, cavy, cheetah, chicken, chub, clam, crab, crayfish, crow).
- Data Table (A+B+C)** widget: Displays a combined table with 17 instances, including all animals from the previous two tables.

The Data Table (A+B+C) widget shows the following data:

type	name	hair	feathers	eggs	milk
mammal	aardvark	1	0	0	1
mammal	antelope	1	0	0	1
fish	bass	0	0	1	0
mammal	bear	1	0	0	1
mammal	boar	1	0	0	1
mammal	buffalo	1	0	0	1
fish	carp	0	0	1	0
fish	catfish	0	0	1	0
mammal	cavy	1	0	0	1
mammal	cheetah	1	0	0	1
bird	chicken	0	1	1	0
fish	chub	0	0	1	0
invertebrate	clam	0	0	1	0
invertebrate	crab	0	0	1	0
invertebrate	crayfish	0	0	1	0
bird	crow	0	1	1	0

1.11-2 示例图片

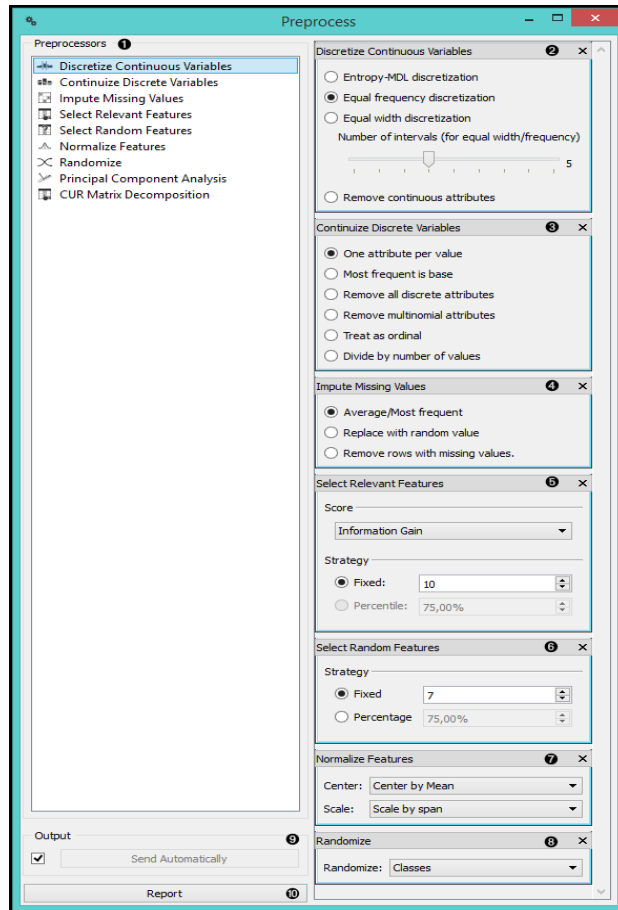
1.12 预处理



使用选定的方法对数据进行预处理

1.12.1 描述

预处理对于实现更好质量的分析结果至关重要。Preprocess 组件提供五种预处理方法来提高数据质量。在这个组件中，您可以立即离散连续值或将连续值离散化，插补缺失值，选择相应的边界或中心并拓展它们。基本上，这个组件组合了 4 个独立的组件来更简便的进行预处理。



1.12-1 Preprocess 窗口

1. 预处理器列表。将要使用的预处理器拖动到组件的右侧。
2. 连续值的离散化。
3. 离散值的连续化。
4. 插补缺失值或删除它们。
5. 通过信息增益，增益比，基尼指数来选择最相关的特征。
6. 选择随机范围。
7. 规范化的范围。
8. 随机化。

9. 此项 (Send Automatically) 被选中, 组件会自动上传更改。否则, 点击 send。

10. 制作报告。

1.12.2 示例

在下面的例子中, 我们使用 adult 数据集并预处理它。我们将离散值 (年龄, 教育和婚姻状况...) 连续化作为每个价值的一个属性, 我们估算缺失值 (用平均值替换), 通过信息增益选择 10 个最相关的属性, 以平均值为中心, 按跨度缩放。我们可以在 Data Table 中观察数据变化, 并将其与未处理的数据进行比较。

The screenshot displays the Orange3 software interface for data preprocessing. It includes a workflow canvas with 'File', 'Preprocess', and 'Data Table (Preprocessed)' widgets. A 'Preprocess' widget configuration window is open, showing settings for 'Continue Discrete Variables' (set to 'One attribute per value'), 'Impute Missing Values' (set to 'Average/Most frequent'), 'Select Relevant Features' (set to 'Information Gain' with a score of 10), and 'Normalize Features' (set to 'Center: Center by Mean' and 'Scale: Scale by span'). The 'Send Automatically' checkbox is checked. Below the configuration, two 'Data Table' windows are shown. The top window, 'Data Table', displays the original dataset with columns: y, marital-status, age, workclass, fnlwgt, and education. The bottom window, 'Data Table (Preprocessed)', shows the transformed data with columns: y, marital-status:Married-civ-spouse, relationship:Husband, marital-status:Never-married, and age. The 'Send Automatically' checkbox is also checked in this window.

1.12-2 示例图片

1.13 估算

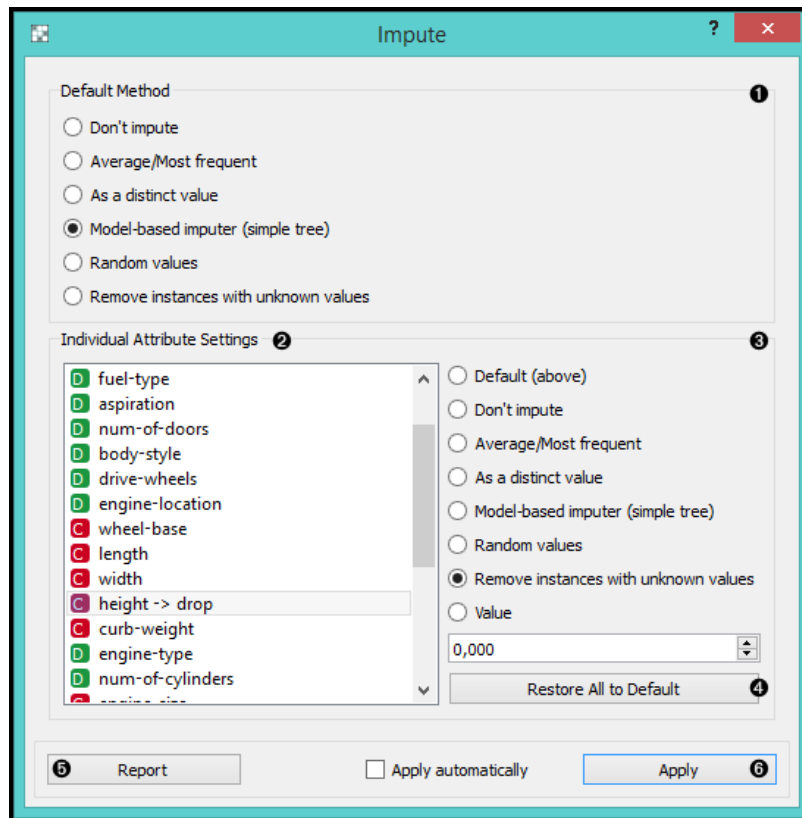


替换数据中的未知值

1.13.1 描述

某些 Mining 算法和可视化无法处理数据中的未知值。这个组件进行统计学家所谓的估算：

它将未知值替换成从数据中计算的值或用户设置的值。



1.13-1 Impute 窗口

1. 在最上面的框中，Default method，用户可以为所有属性指定一个通用的估值方法。

- Don't Impute 不对缺失值进行任何操作。
- Average/Most-frequent 使用平均值（对于连续属性）或最常见的值（对于离散属性）。
- Model-based imputer 构建一个用于根据其他属性的值预测缺失值的模型；为每个属性构建一个单独的模型。默认模型为 1-NN learner，该模型从最相似的示例（这有时被称为 hot deck 估算）中获取值。该算法可以由用户连接到输入信号 Learner for Imputation 的算法来代替。但是，请注意，如果数据中有离散属性和

连续属性，那么该算法需要能够处理这两种属性；目前只有 kNN learner 才能实现该操作。（将来，当 Mining 具有更多回归量时，估算 (Impute) 组件可能会拥有分别用于离散模型和连续模型的单独输入信号。）

- Random values 计算每个属性值的分布，然后通过从它们中选取随机值进行估算。
- Remove examples with missing values 删除包含缺失值的示例，但保留按照如下方式定义特定操作的属性。该检查也适用于类属性（如果选中 Impute class values）。

2. 也可以为每个属性指定单独的处理，以替代上面设置的默认处理。也可以指定用于估算的手动定义的值。在这个图片中，我们决定不估算 “normalized-losses” 和 “make” 的值，“aspiration” 的缺失值将替换为随机值，而 “body-style” 和 “drive-wheels” 的缺失值则分别替换为 “hatchback” 和 “fwd”。如果 “length”、“width” 或 “height” 的值缺失，则丢弃该示例。所有其他属性的值使用上面设置的默认方法（在我们的示例中为基于模型的估算方法）。

3. 对于个别属性来说，归责方法与默认方法相同。

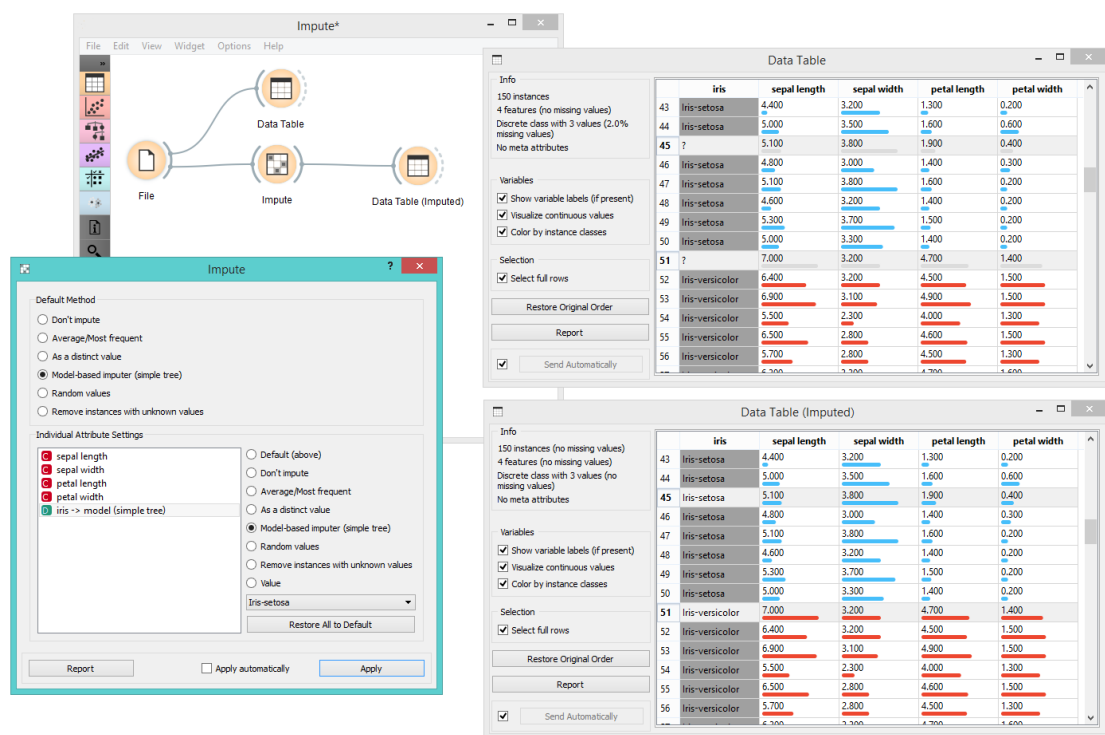
4. 按钮 Restore All to Default 将各个属性处理重置为默认值。

5. 制作报告。

6. 立即提交所有更改是选中 Send automatically。否则需要按 Apply 才能应用任何新的设置。

1.13.2 示例

为了演示如何使用 Impute 组件，我们使用了 Iris 数据集，并删除了一些数据。我们使用 Impute 组件并选择 Model-based imputer 来估算删除的数据。在另一个数据表,我们看到问号变成了不同的值(Iris-versicolor “Iris-setosa,)



1.13-2 示例图片

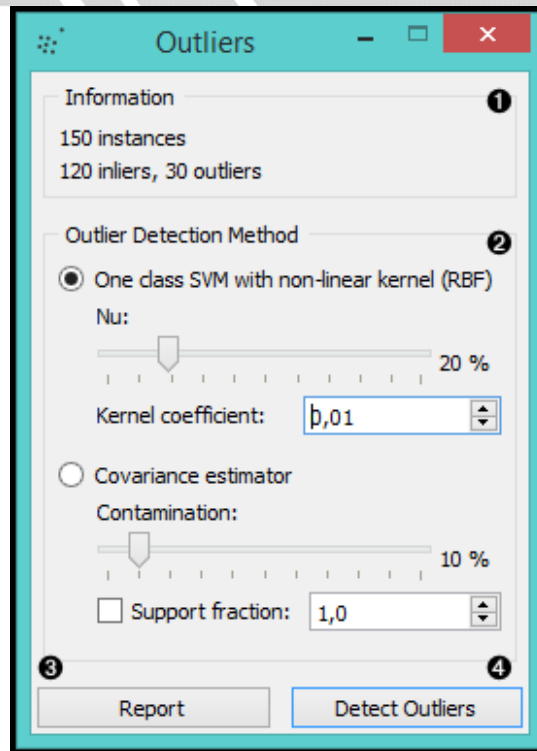
1.14 离群点



通过比较示例之间的距离来进行简单离群点检测。

1.14.1 描述

离群点 (Outliers) 组件应用两种方法中的一种来进行异常检测。这两种方法都将分类应用于数据集,一个是用 SVM(多核),另一个是用椭圆包络。One-class SVM with non-linear kernels (RBF)与非高斯分布的数据契合,而 Covariance estimator 则只适用于高斯分布的数据。



1.14-1 Outliers 窗口

1. 根据所选模型的输入数据、输入端数和离群值的信息。。
2. 选择孤立点检测方法 (Outlier detection method) 。

One class SVM with non-linear kernel (RBF) : 将数据分类为与核心类相似或不同的数据。

Nu 是训练误差分数上界和支持向量分数下界的一个参数。

Kernel coefficient 是一个伽马参数，它指定单个数据实例有多大的影响。

Covariance estimator: 用马氏距离度量符合中心点的点。

Contamination 是数据集中的异常值的比例。

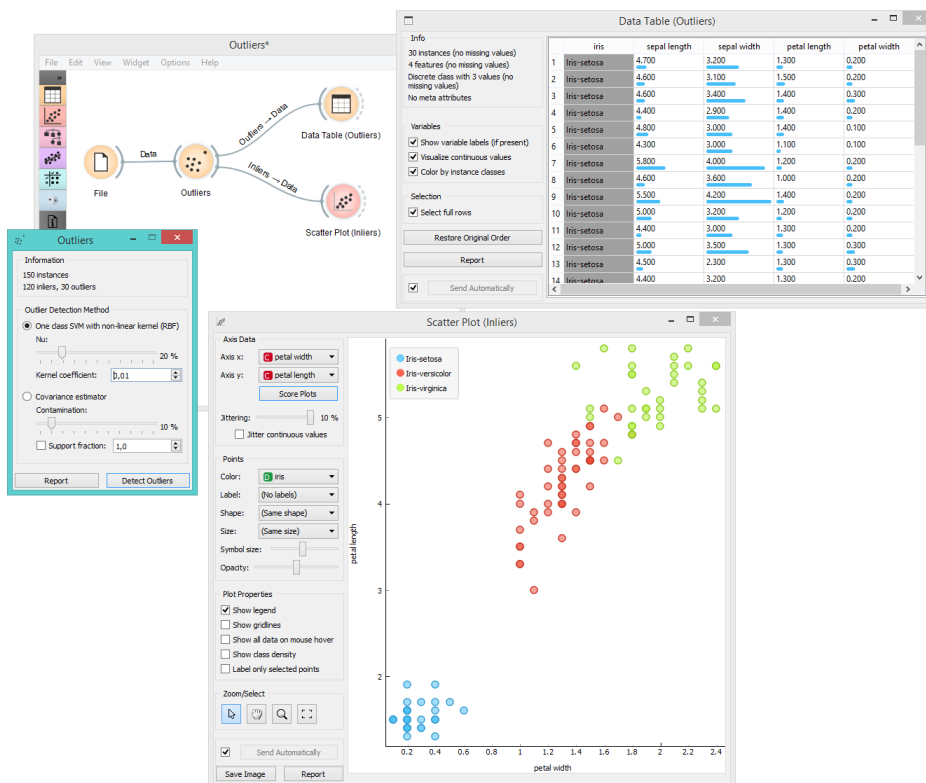
Support fraction 指定估算中包含的点的比例。

3. 制作一个报告。

4. 点击 Detect outliers 来输出数据。

1.14.2 示例

下面是使用这个组件的一个简单的例子。我们使用 Iris 数据集来检测异常值。选择 one class SVM with non-linear kernel (RBF)方法，Nu 设为 20% (较少的训练错误，更多的支持向量)。然后我们在 Data Table 组件中观察离群值，同时把正常值传输到 Scatter Plot 组件中。



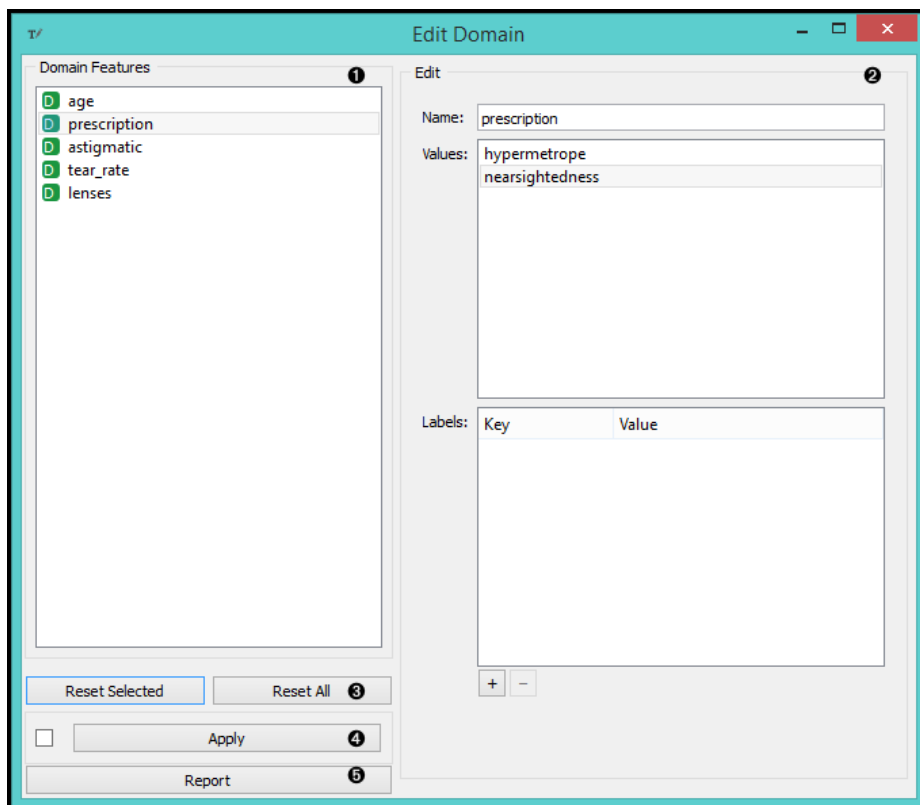
1.14-2 示例图片

1.15 编辑域



1.15.1 描述

这个组件可用于编辑/更改数据集的域



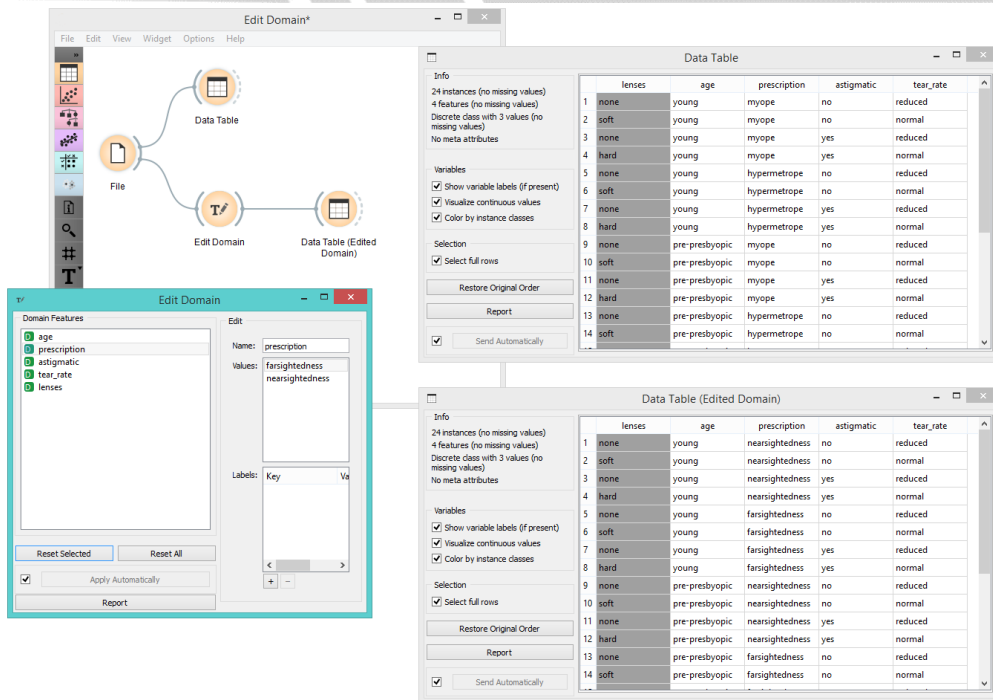
1.15-1 Edit Domain 窗口

1. 数据 (Data) 输入数据集中的所有特征 (包括元属性) 都在左侧 Domain Features 列表框中列出。选择一个特征会在右侧显示编辑器。

2. 该特征的名称可以在 Name 行编辑中更改。对于 Discrete 特征，也可以在 Values 列表框中更改值名称。可以在 Labels 框中添加/删除/编辑其他特征注释。若要添加新标签，请单击 + 按并添加这个新条目的 Key 和 Value 列。选择一个现有标签并按 - 将删除注释。
3. 若要还原对某个特征进行的更改，请在 Domain Features 列表中选中该特征的同时按 Reset 框中的 Reset Selected 按钮。按 Reset All 会同时重置域中的所有特征。
4. 按 Apply 按钮会在数据 (Data) 输出通道上发送更改后的域数据集。
5. 制作报告。

1.15.2 示例

下面，我们演示如何简单的编辑一个现有的域。我们选择了 lens.tab 数据集并编辑了 perscription 属性。在原始数据中，我们有 myope 和 hypermetrope 的值，把它改为 nearsightedness 和 farsightedness。为了便于比较，我们把原始的和编辑的数据都添加到 Data Table 组件中。



1.15-2 示例图片

1.16 Python 脚本



通过 Python 脚本扩展功能。

1.16.1 描述

当在现有组件中未实现适当的功能时，可以使用 Python Script 组件在输入上运行 python 脚本。脚本在其本地命名空间中包含 in_data、in_distance、in_learner、in_classifier 和 in_object 变量（来自输入信号）。如果某个信号未连接或者它尚未收到任何数据，则这些变量包含 None。

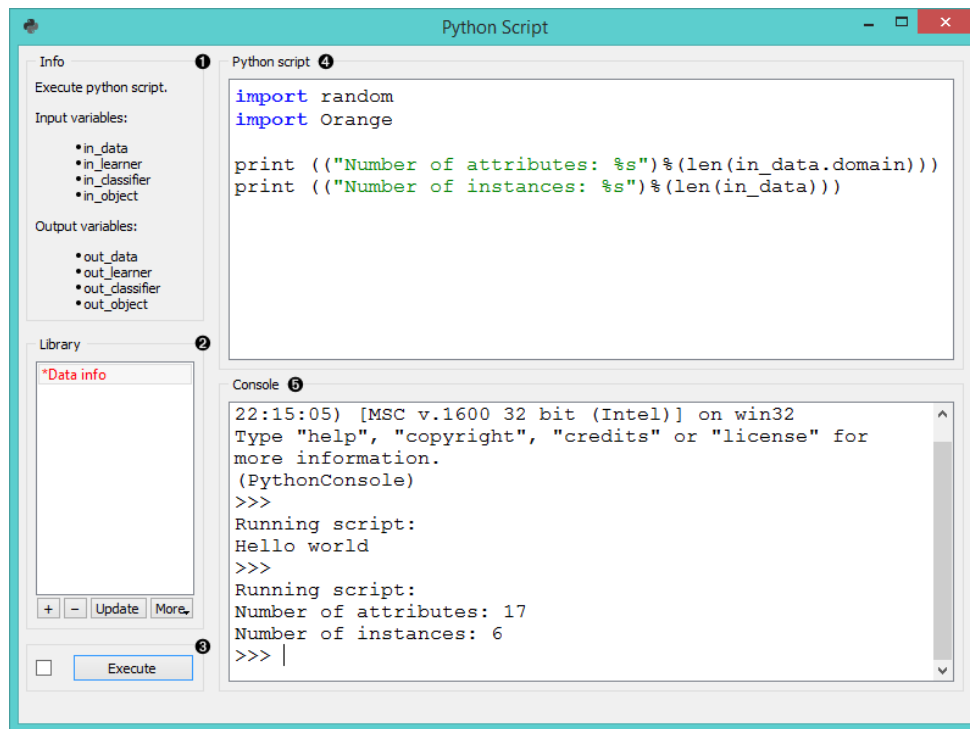
执行脚本之后，会从脚本的本地命名空间提取 out_data、out_distance... 变量并用作这个组件的输出。这个组件可以进一步连接到其他部件可视化输出。

例如下面的脚本将简单地传递它接收到的所有信号：

```
out_data = in_data  
  
out_distance = in_distance  
  
out_learner = in_learner  
  
out_classifier = in_classifier  
  
out_object = in_object
```

Note

You should not modify the input objects in place.



1.16-1 Python Script 窗口

1. 信息框中包含 Mining Python 脚本的基本操作符的名称。
2. Library 可以用于控制多个脚本。按 “+” 将添加一个新条目，并在 Python 脚本编辑器中打开它。修改脚本时，Library 中的条目将会更改，以指示它具有未保存的更改。按 Updat 将保存脚本（键盘快捷键 Ctrl + s）。可以通过选择脚本并按 “-” 按钮来删除脚本。
3. 在运行框中点击 Execute 来运行脚本（使用 exec）。任何脚本输出（从 print）被读取并显示在脚本下面的控制台（Console）中。如果选中自动执行（选项框打钩），脚本将随时随地输入到组件进行更改。
4. 可以使用上方的 Python 脚本编辑器来编辑脚本（它支持一些基本的语法高亮）。
5. 控制台（Console）显示脚本的输出。

1.16.2 示例

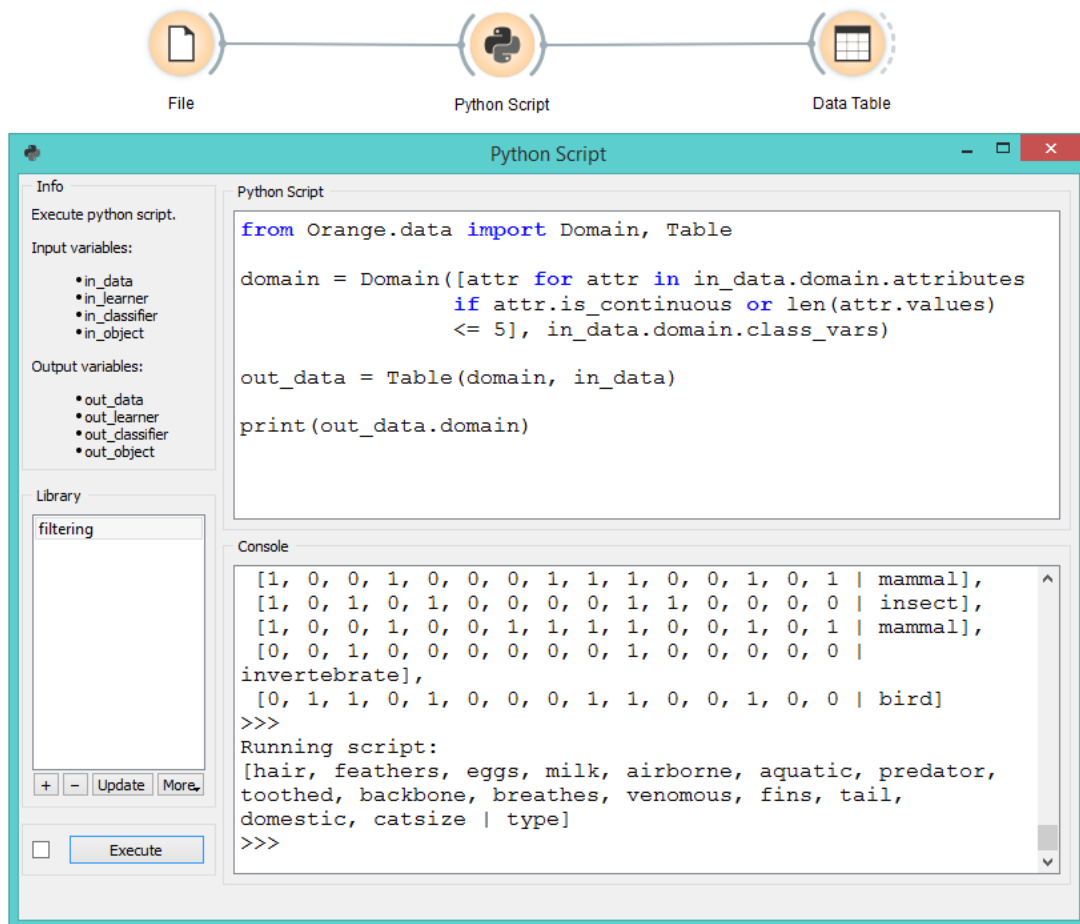
Python Script 组件旨在扩展高级用户的功能。我们使用 zoo.tab 作为示例，我们滤除了所有具有超过 5 个离散值的属性。

例如，可以通过属性进行批量过滤。我们使用 zoo.tab 作为示例，我们滤除了所有具有超过 5 个离散值的属性。在这个例子中，我们只删除了“leg”属性，但是想象一个例子，其中有许多这样的属性。

```
from Orange.data import Domain, Table

domain = Domain([attr for attr in in_data.domain.attributes
                 if attr.is_continuous or len(attr.values) <= 5],
               in_data.domain.class_vars)

out_data = Table(domain, in_data)
```



1.16-2 示例图片

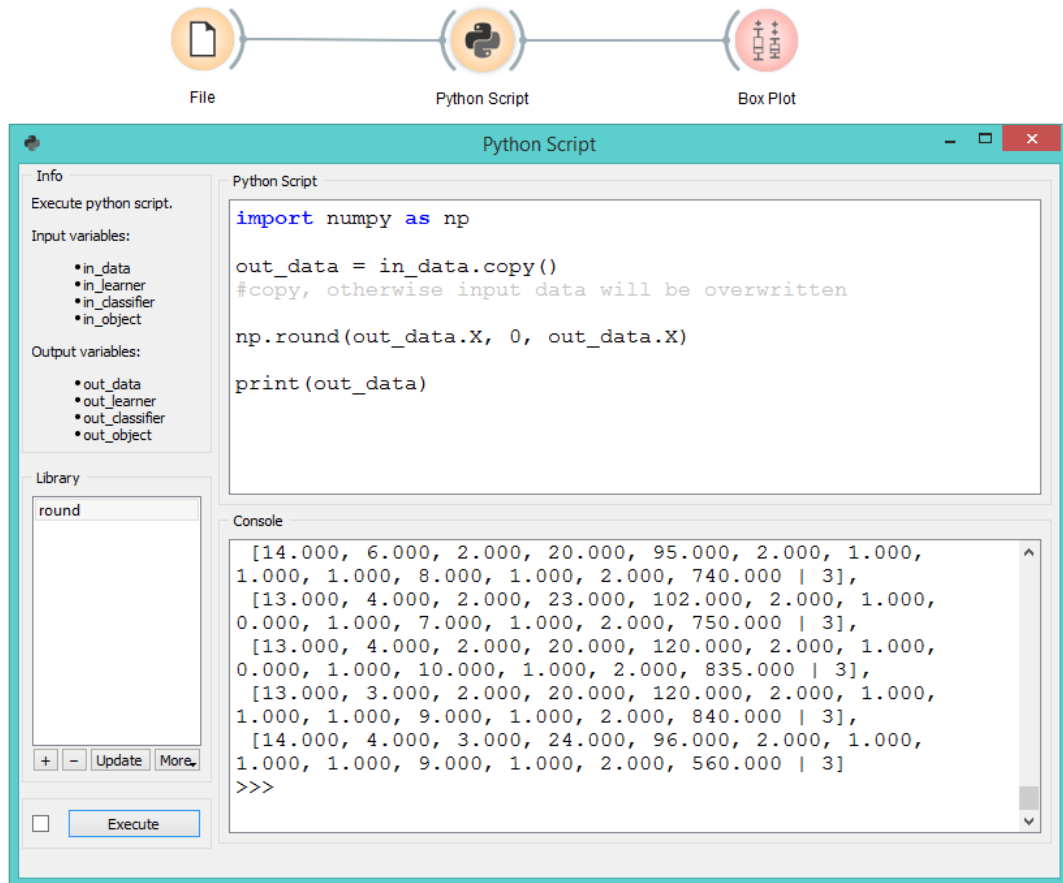
第二个示例演示如何用几行代码循环所有值。这次我们使用 wine.tab 并且循环所有数字。

```
import numpy as np
```

```
out_data = in_data.copy()
```

```
#copy, otherwise input data will be overwritten
```

```
np.round(out_data.X, 0, out_data.X)
```



1.16-3 示例图片

第三个例子为数据引入了一些高斯噪声。我们再次创建一个输入数据的副本，然后用双循环遍历所有的值，并添加随机噪声。

```
import random

from Orange.data import Domain, Table

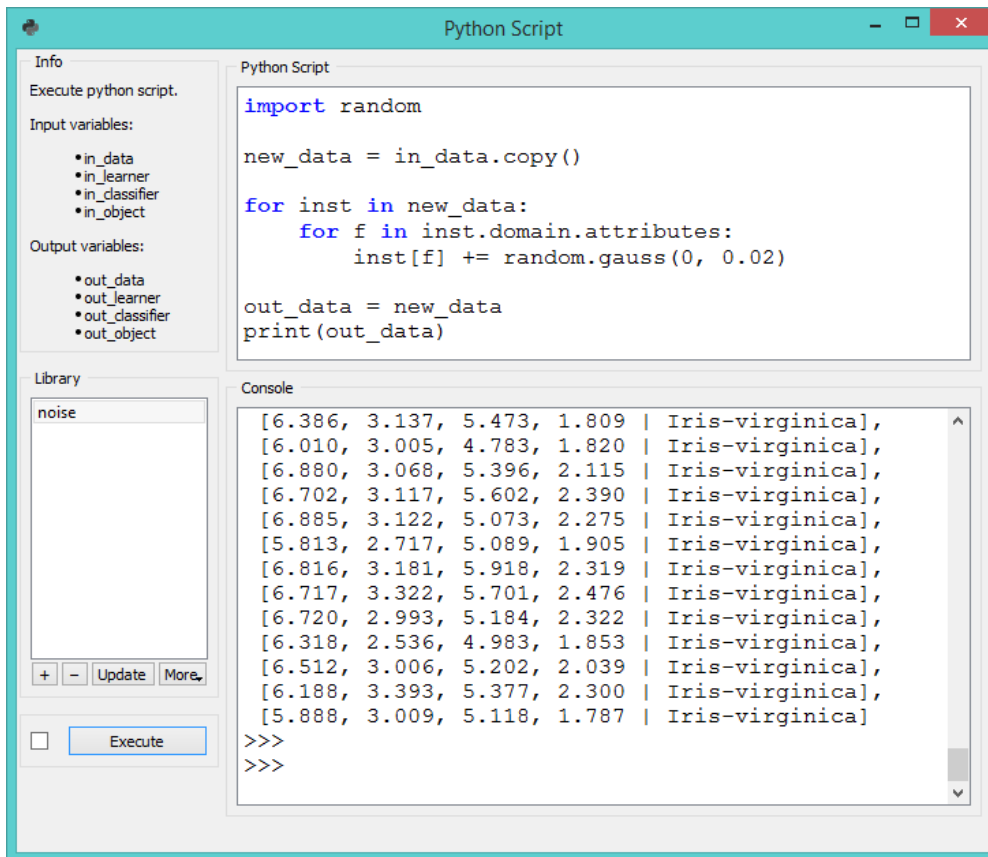
new_data = in_data.copy()

for inst in new_data:

    for f in inst.domain.attributes:

        inst[f] += random.gauss(0, 0.02)
```

```
out_data = new_data
```



1.16-4 示例图片

最后一个例子使用 Mining3-Text 加载项。Python Script 对于文本挖掘中的自定义预处理非常有用，从字符串中提取新功能，或者使用高级 nltk 或 gensim 函数。下面，我们简单地通过用空格分割 deerwester.tab 来表示我们的输入数据。

```
print('Running Preprocessing ...')

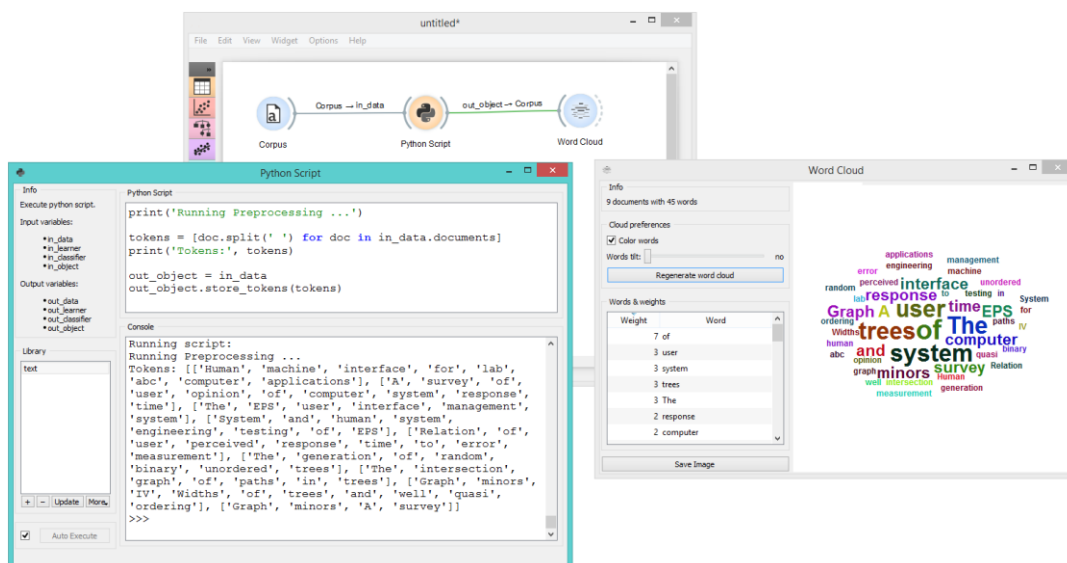
tokens = [doc.split(' ') for doc in in_data.documents]

print('Tokens:', tokens)

out_object = in_data
```

out_object.store_tokens(tokens)

您可以添加许多其他预处理步骤来进一步调整输出。 Python Script 的输出可以与任何接受脚本生成的输出类型的组件一起使用。在这个例子中，连接是绿色的，这意味着 Word Cloud 组件有正确类型的输入。



1.16-5 示例图片

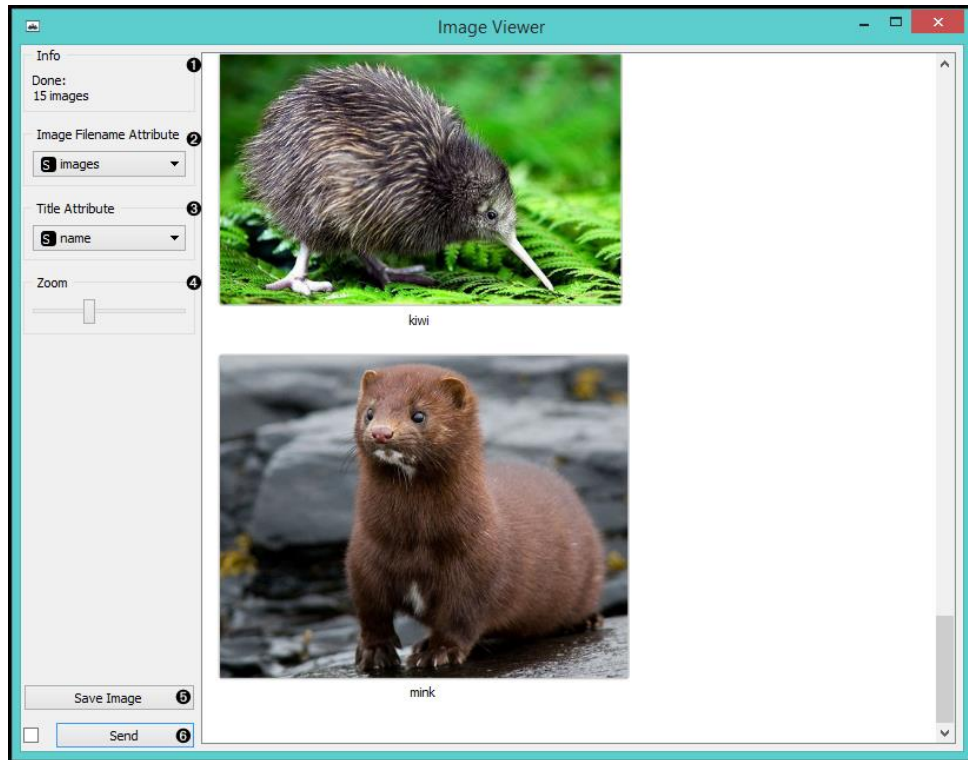
1.17 图片浏览



显示数据集随附的图片

1.17.1 描述

Image Viewer 组件可以显示来自本地或互联网上的数据集的图像。它可以用于图像比较，同时寻找所选数据实例之间的相似性或差异（例如手写的细菌生长或位图表示）。

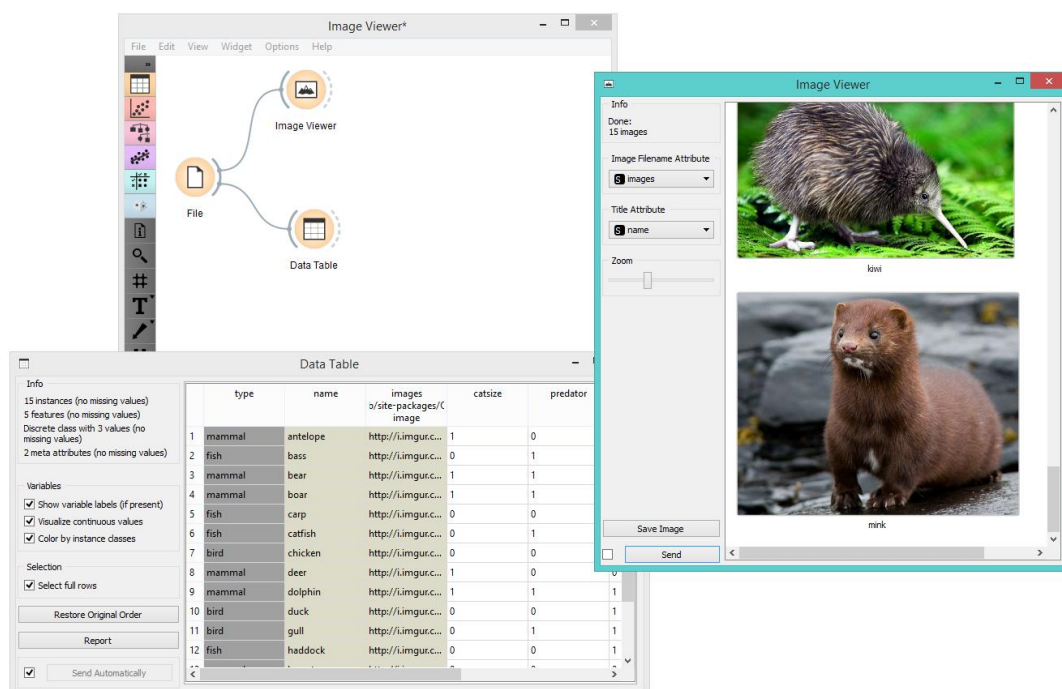


1.17-1 Image Viewer 窗口

1. 数据集的信息。
2. 选择带有图像数据（链接）的列。
3. 选择带有图像标题的列。
4. 放大或缩小。
5. 将图片保存在文件中。
6. 立即提交所有更改时选中 Send automatically。否则需要按 Apply 才能应用任何新的设置。

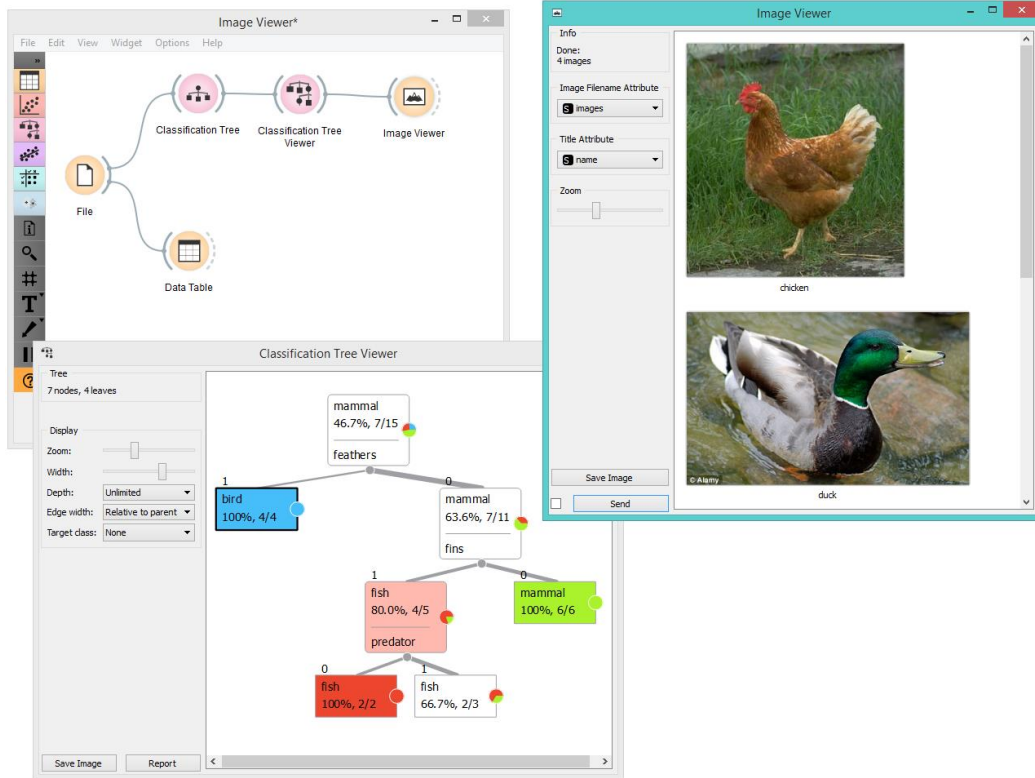
1.17.2 示例

使用这个组件的一个非常简单的方法是将 File 组件与 Image Viewer 连接，并查看数据集附带的所有图像。



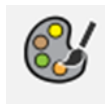
1.17-2 示例图片

或者，你可以只显示选定的实例，如下面的示例所示。



1.17-3 示例图片

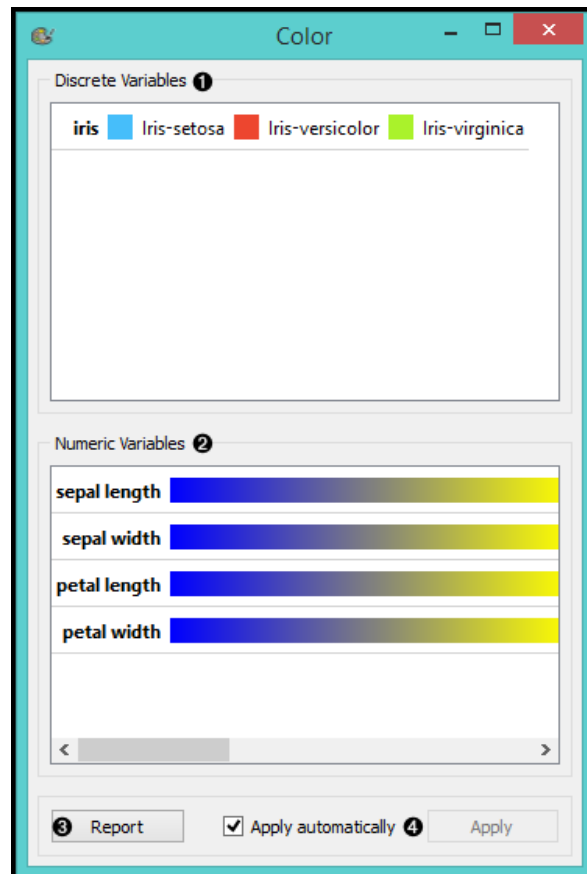
1.18 颜色



设置变量的颜色图例

1.18.1 描述

Color 组件使您可以根据自己的喜好在可视化设置中设置颜色图例。此组件为您提供强调结果的组件，并提供各种颜色选项来显示数据。它可以与大多数可视化组件组合。

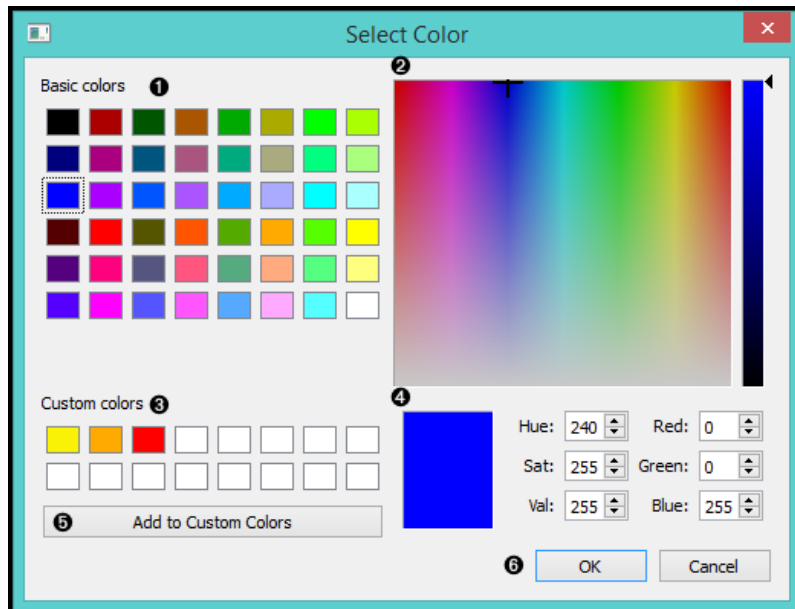


1.18-1 Color 窗口

1. 离散变量列表。您可以通过双击它并打开“调色板”或“选择颜色”窗口来设置每个变量的颜色。该组件还支持文本编辑。通过点击变量，您可以更改其名称。
2. 连续变量列表。您可以通过双击它们自定义颜色渐变。该组件还支持文本编辑。通过点击变量，您可以更改其名称。如果将鼠标悬停在渐变的右侧，则复制到所有显示。然后，您可以将自定义的颜色渐变应用于所有变量。

3. 制作报告。
4. 应用更改。如果自动应用 (Apply automatically) 勾选 , 则会自动进行更改。
或者 , 只需单击应用 (Apply) 。

1.18.2 离散变量

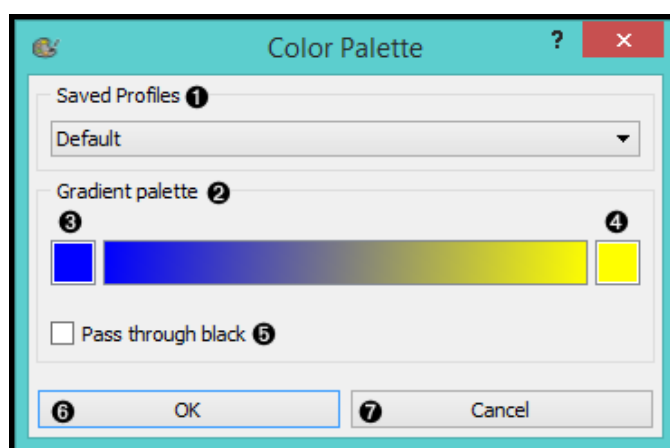


1.18-2 Select Color 窗口

1. 从基本颜色的调色板中选择所需的颜色。
2. 移动光标从调色板中选择一个自定义颜色。
3. 从您以前保存的颜色选择中选择一种自定义颜色。
4. 通过以下方式指定自定义颜色：

- 输入颜色的红色，绿色和蓝色分量为 0（最暗）和 255（最亮）之间的值
 - 输入颜色的色相，饱和度和发光成分，范围为 0 到 255。
5. 将创建的颜色添加到您的自定义颜色。
 6. 单击 OK 来保存您的选择或 Cancel 退出调色板。

1.18.3 数值变量



1.18-3 Color Palette 窗口

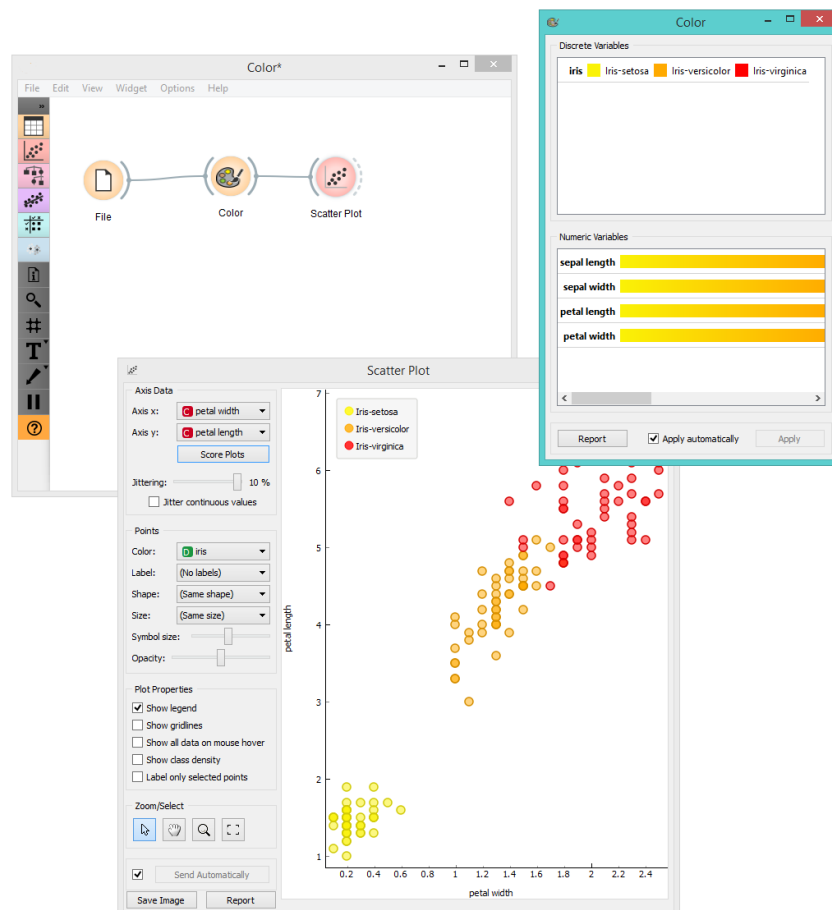
1. 从保存的配置文件中选择渐变。默认配置文件已设置。
2. 渐变调色板。
3. 选择渐变的左侧。双击颜色将打开“选择颜色”窗口。
4. 选择渐变的右侧。双击颜色将打开“选择颜色”窗口。
5. 通过黑色。

6. 点击 OK 来保存您的选择。
7. 点击 Cancel 来退出调色板。

1.18.4 示例

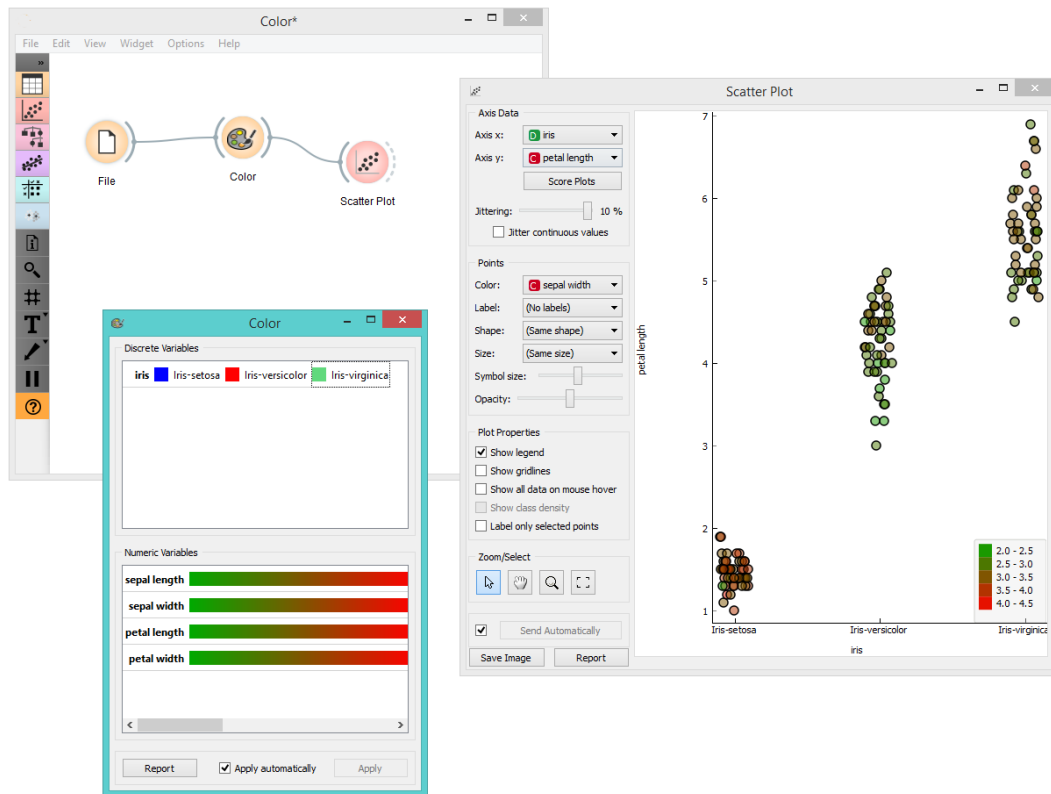
我们选择使用 Iris 数据集。我们打开了调色板，并为三种类型的 Irises 选择了三种新颜色。

然后我们打开了散点图 (Scatter Plot) 组件，并查看了对散点图所做的更改。



1.18-4 示例图片

在我们第二个例子中，我们希望演示 Color 组件对于具有连续变量的数据集的使用。我们在 x 轴上放置不同类型的鸢尾花，在 y 轴上放置 petal length。我们创建了一个新的颜色渐变，并将其命名为 greed (绿色+红色)。为了表明 sepal length 不是区分不同类型的鸢尾花的决定因素，我们选择根据 sepal width 对点进行着色。



1.18-5 示例图片

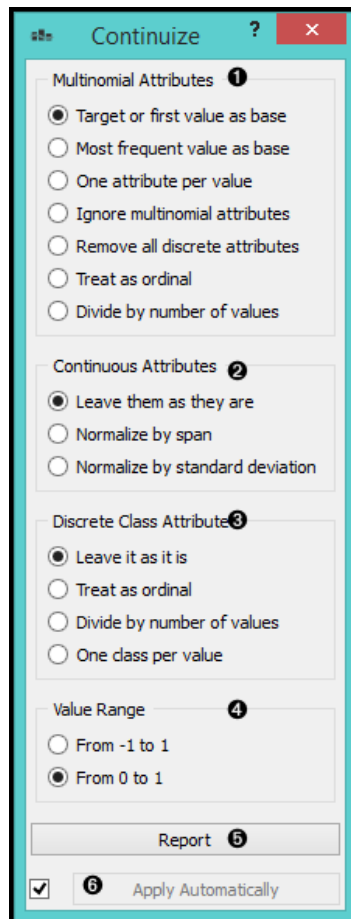
1.19 连续化



将离散属性转换为连续虚拟变量。

1.19.1 描述

Continuize 组件接收输入上的数据集并输出使用用户指定的方法将离散属性（包括二进制属性）替换为连续属性的数据。



1.19-1 Continuize 窗口

1. Continuization methods 定义多值离散属性的处理。比方说，我们拥有一个值为 low、middle、high（按这个顺序列出）的离散属性状态。那么用于其变换的选项如下。

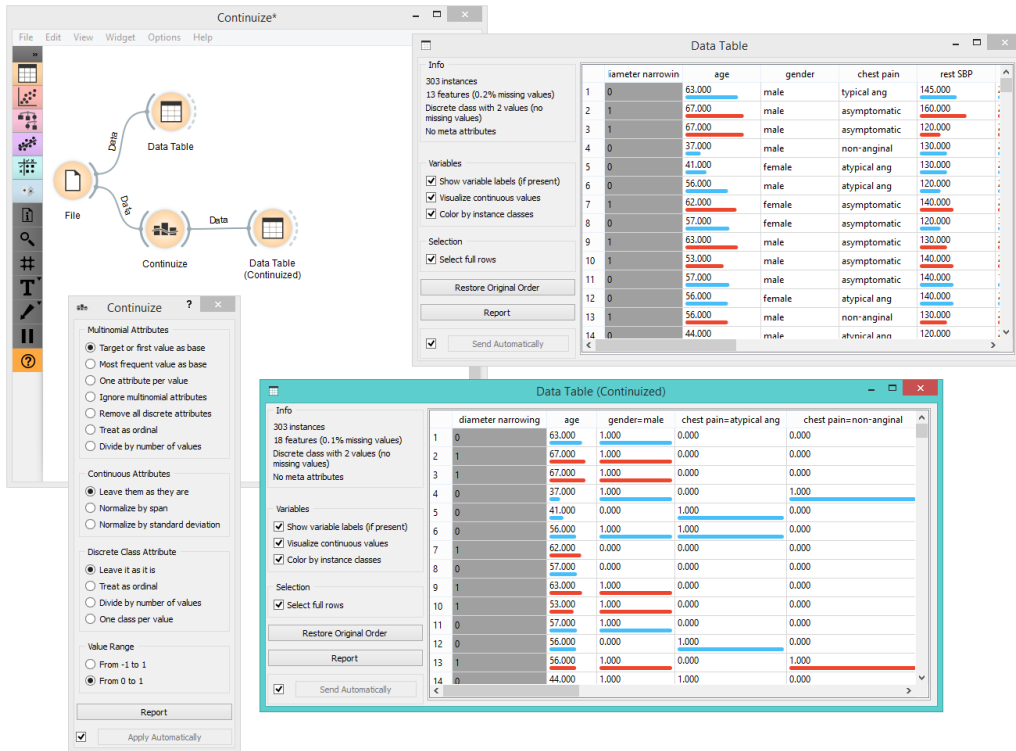
- Target or First value as base 该属性将变换成两个连续的属性，status=middle，值为 0 或 1，表示原始属性在特定示例上是否具有 middle 值，同样也适用于 status=high。因此，三值属性变换成两个连续属性，对应于该属性第一个值之外的所有属性。
- Most frequent value as base：与上述内容类似，只是会分析数据并且将最常用的值用作基本值。因此，如果大多数示例都拥有值 middle，那么两个新购机的连续属性将为 status=low 和 status=high。
- One attribute per value：该选项将从一个三值离散属性中构建三个连续属性。
- Ignore multinominal attributes：从数据中删除多项 (multinomial) 属性。
- Treat as ordinal：将属性转换为一个具有值 0、1 和 2 的连续属性。
- Divide by number of values：与上面的内容相同，只是会将值归一化到 0-1 范围之内。因此我们的示例将给出值 0、0.5 和 1。

2. 连续属性的处理。您通常会首选 Leave them as they are 选项。另一个选项是 Normalize by span，该选项将指示数据中的最低值并且除以跨度，因此所有值都将适合 [0, 1]。最后，Normalize by standard deviation 会减去平均值并除以方差。

3. 定义类属性的处理。除了保持原样之外，还有可用于多项 (multinomial)属性的选项，只是这些选项会将该属性拆分为多个属性，显然这可能无法得到支持，因为您无法拥有多个类属性。
4. 借助 value range，您可以定义新属性的值。在上面的文本中我们假定范围 from 0 to 1。您可以将它更改为 from -1 to 1。
5. 制作报告。
6. 如果设置 Send automatically，则会在发生任何更改时提交数据集。否则，您必须在每次更改之后按 Send data。

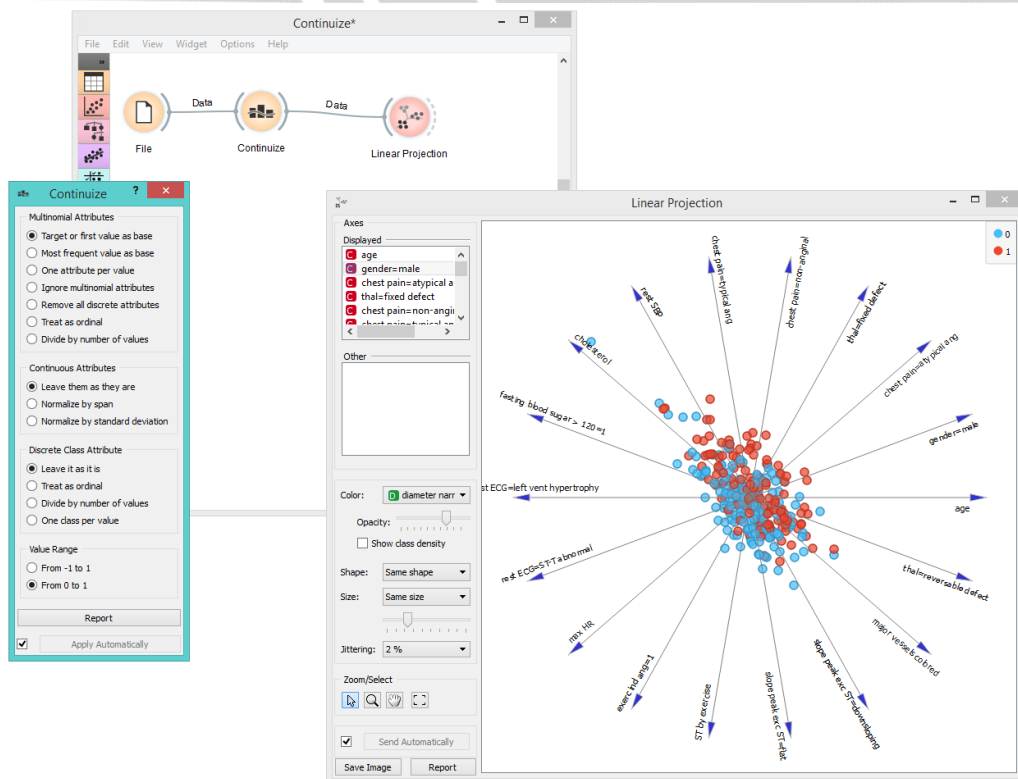
1.19.2 示例

首先，让我们观察一下 Continuize 组件的输出是什么。我们把原始数据 (Heart disease 数据集) 添加到 Data Table 组件里，然后观察。然后我们连续化属性的值，并把它添加到另一个 Data Table 里。



1.19-2 示例图片

在第二个例子中，我们展示了这个组件的典型用法。为了恰当的绘制数据的线性投影，离散的属性需要转变成连续的属性，这也是我们在绘制投影前需要先把数据通过 Continuize 组件。属性 chest pain 最初有 4 个值，被转化成 3 个连续属性，类似的情况发生在性别上，它被转化成单一属性。



1.19-3 示例图片

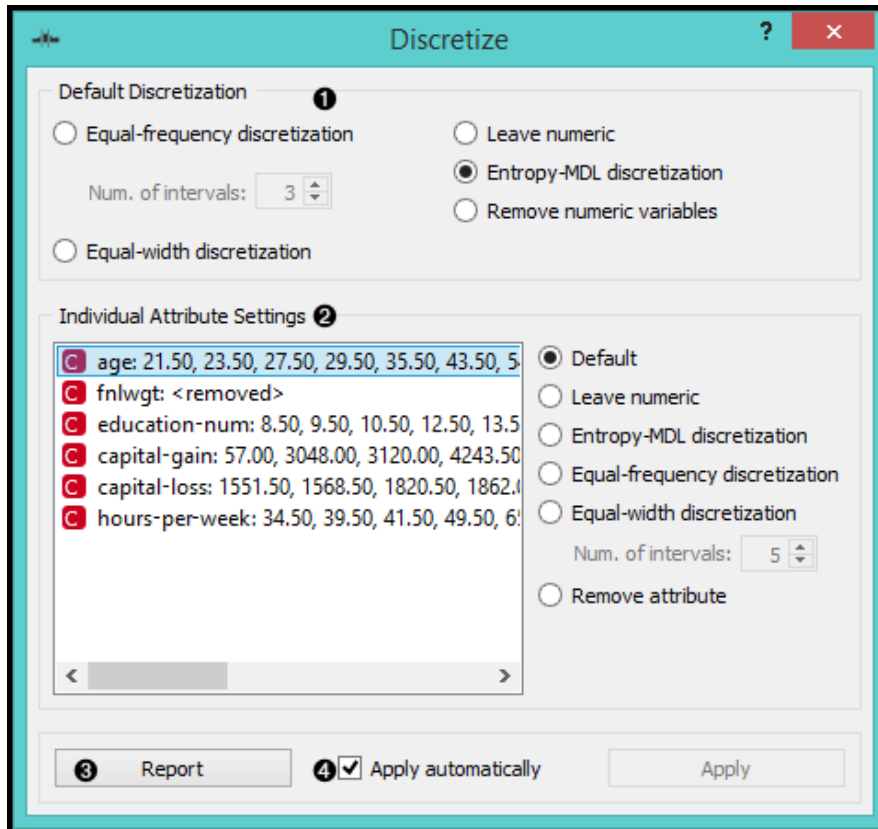
1.20 离散化



离散化输入数据集中的连续属性

1.20.1 描述

离散化 (Discretize) 组件用选取的方法离散连续的属性。



1.20-1 Discretize 窗口

1. 这个组件的基本构成十分简单。它可以选择三种不同的离散方法。

Entropy-MDL, 由 Fayyad 和 Irani 发明的一种自上而下的递归分割的属性离散化方法。它是将属性一步步进行最大化的信息增益切割, 直到增益小于切割的最小描述长度。这种离散化方法可能会形成任意大小的间隔, 包括单一的间隔, 而这种情况下, 属性是无用的。

Equal-frequency 将属性分割成指定数量的间隔数, 使他们各自包含大致相同的实例。

Equal-width 均匀的分割最小和最大观测值之间的范围。并且可以手动设置间隔数。

这个组件也可以设置为舍弃连续属性或者删除连续属性。

2. 我们使用 Individual Attribute Settings 来分别处理属性。它们显示每个属性的特定离散化并允许更改。首先，左上角列表显示每个属性的分界点。在简介中，我们使用 Entropy-MDL 离散法来自动确定最佳区间数目。我们可以看到它将年龄离散为 7 段 :21.50, 23.50, 27.50, 35.50, 43.50, 54.50, 61.50. 同时，资本收益(capital-gain) 被分割成有一些间断的许多间隔。最后的重量 (fnlwgt) 被留在一个单一的间隔，并且去除掉。

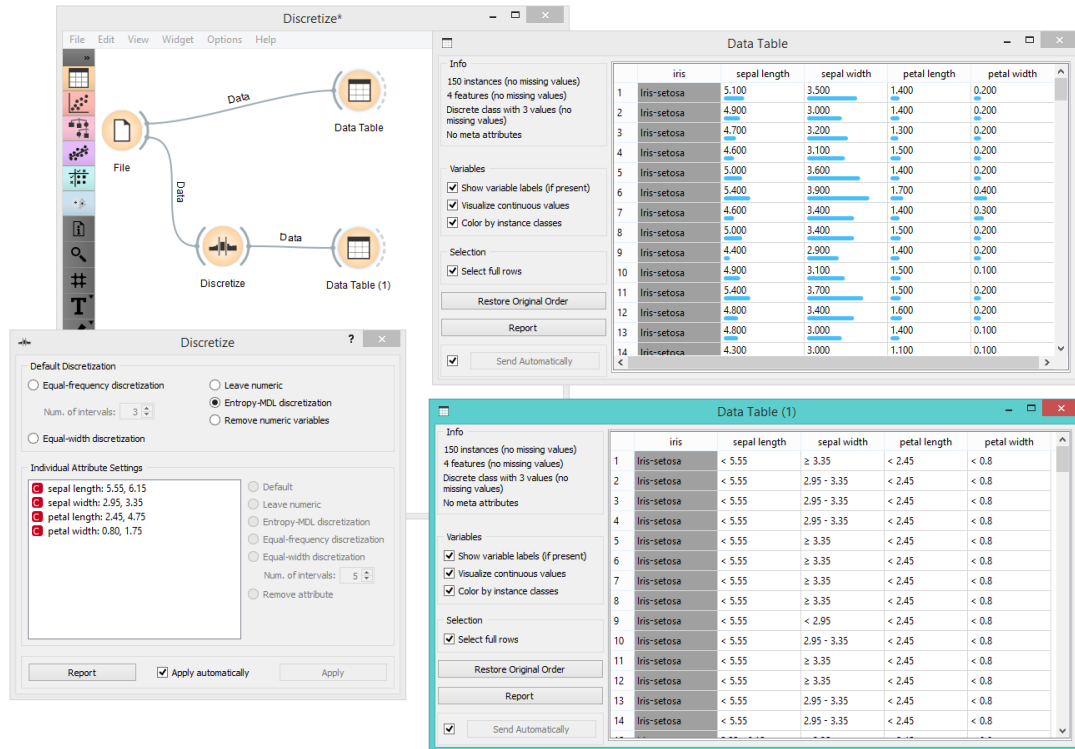
在右边，我们可以为每个属性选择一个特定的离散化方法。属性 fnlwgt 被 MDL 离散法去除，为了防止它被去除，我们选择这个属性，并选择另外的离散法，例如，Equal-frequency 离散法。我们也可以选择舍弃连续属性。

3. 制作报告。

4. 在 Apply automatically 上打钩来自动上传更改。或者点击 Apply。

1.20.2 示例

在下面的示例中，我们显示了具有连续属性（与原始数据文件中一样）和离散化属性的 Iris 数据集。



1.20-2 示例图片

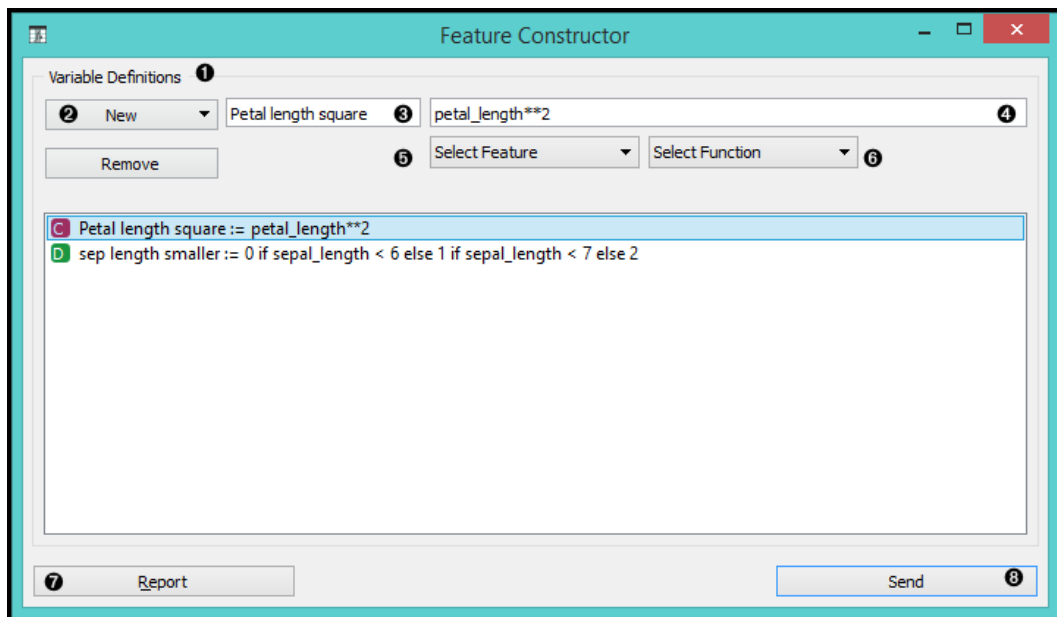
1.21 特性构造函数



为数据集添加新功能

1.21.1 描述

Feature Constructor 组件允许您手动将特征（列）添加到数据集中。新特征可以是现有的一个或多个（加法，减法等）的组的计算。您可以选择它将是什么类型的特点（离散，连续或字符串）及其参数（名称，值，表达式）。对于连续变量，只需要在 Python 中构造一个表达式。

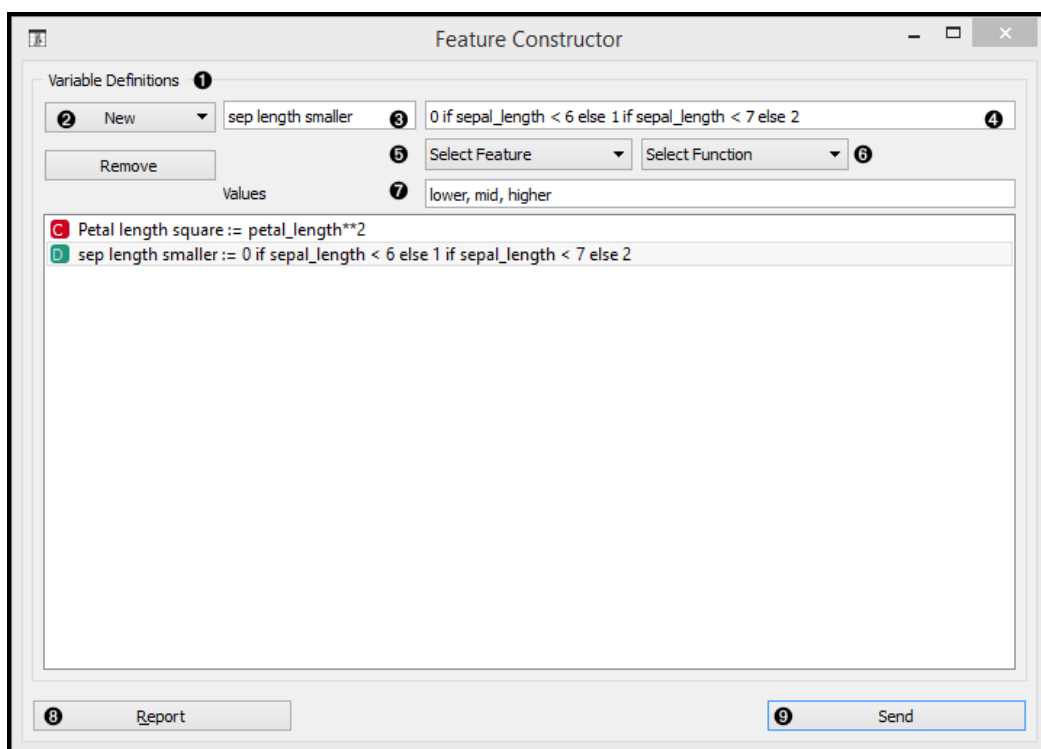


1.21-1 Feature Constructor 窗口（连续变量）

1. 构造变量列表。
2. 添加或删除变量。
3. 新功能名称。
4. 在 Python 中表达。
5. 选择一个特征。
6. 选择一个函数。
7. 制作报告。

8. 按发送 (Send) 更改。

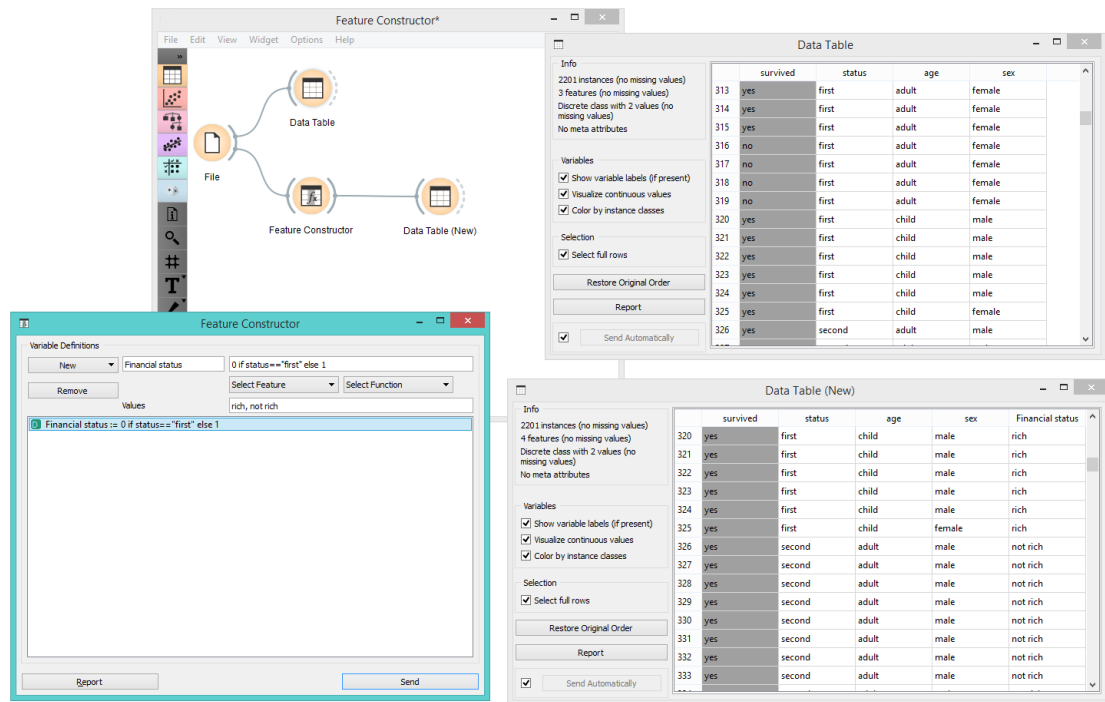
然而，对于离散变量，还有更多的工作。首先添加或删除新功能所需的值。然后选择基本值和表达式。在下面的例子中，我们构造了一个表达式，如果'低于'并定义了三个条件;如果原始值低于 6，则程序归为 0（我们称之为低），如果低于 7 则为 1（中），其余的值则为 2（高）。请注意，我们对功能名称使用下划线（例如 `petal_length`）。



1.21-2 Feature Constructor 窗口 (离散变量)

1.21.2 示例

通过 Feature Constructor，您可以轻松地将现有功能调整或组合成新功能。下面，我们向 Titanic 数据集添加了一个新的离散特征。我们创建了一个名为“财务状况”的新属性，如果该人属于第一类（状态=第一）并且对其他人不富有，则将该值设置为“富”。我们可以使用 Data Table 组件查看新的数据集。



1.21-3 示例图片

1.22 清除域



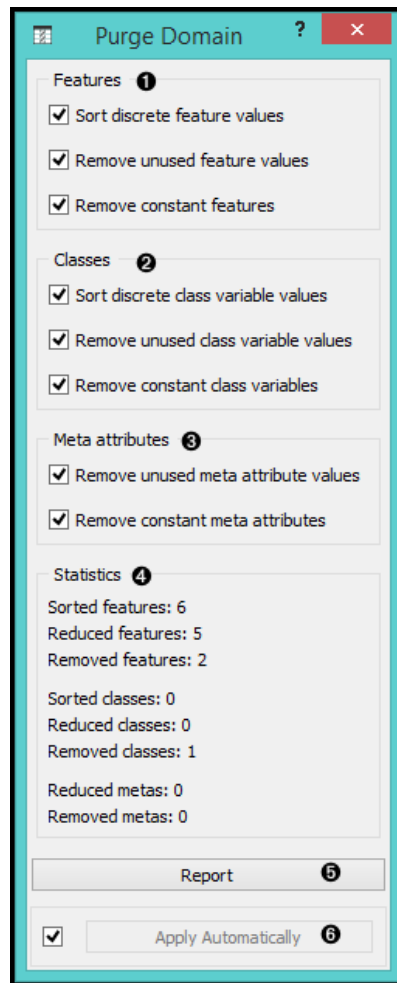
删除未使用的属性值和无用的属性，对剩余的值进行排序

1.22.1 描述

属性的定义有时包含数据中未出现的值。即使原始数据中并未发生这种情况，但过滤数据、选择示例子集以及类似操作都会删除其属性中包含某些特殊值的所有示例。此类值会搞乱数据显示，尤其是各种可视化，因此应当删除。

清除某个属性之后，该属性可能变为单值，或者在极少数情况下，根本没有值（如果未针对所有示例定义此属性的值）。在这种情况下，可以删除该属性。

另一个问题是属性值的顺序：如果数据是从未预先声明值的格式的文件中读取的，则按“出现的顺序 (in order of appearance)”对它们进行排序。有时，我们更希望它们按照字母顺序排序。



1.22-1 Purge Domain 窗口

- 1.清除属性。
- 2.清除类。
- 3.清除元属性。
- 4.过滤过程的信息。
- 5.制作一份清除域的报告。
- 6.如果这项选中，这个组件会在每次组件设置变化时输出数据。

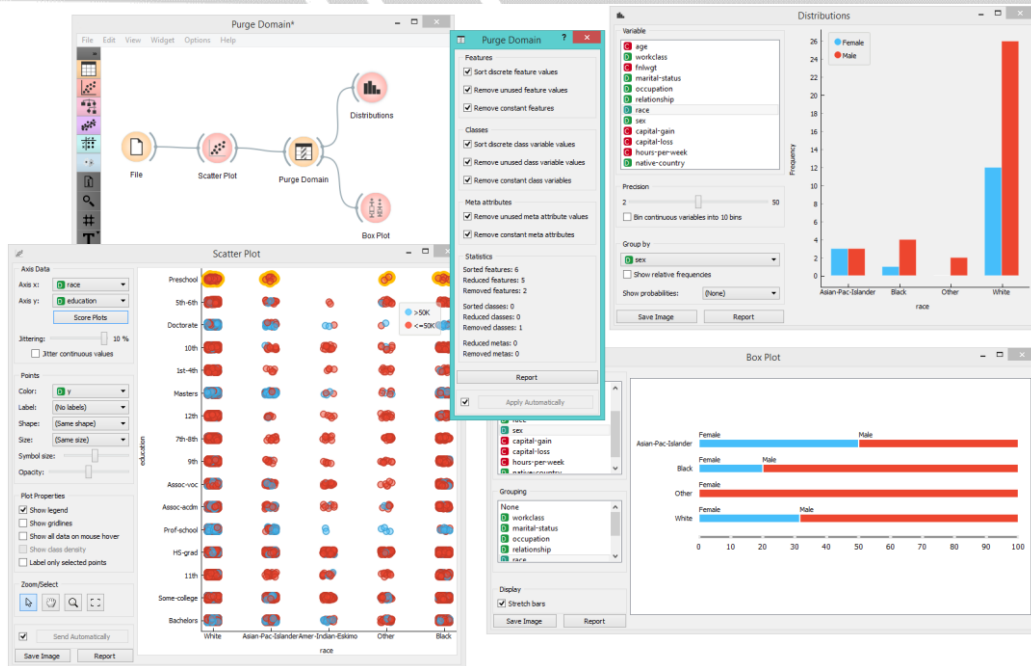
对于每个属性，我们都可以决定是否要对值进行排序。接下来，我们可以允许这个组件删除少于两个值的属性，或者删除少于两个类的类属性。最后，我们可以指示这个组件检查哪些属性的值实际出现在数据中并删除未使用的值。如果不允许删除属性，那么这个组件则无法删除值；因为（可能）拥有没有值的属性没有意义。

新的简化属性得到一个前缀“R”，用于区分这些属性与原始属性。可以从旧属性计算新属性的值，反过来则不可以。这意味着如果您从新属性构建分类器，您可以使用该分类器对原始属性所述的示例进行分类。但反过来则不可以：不能从旧属性构建分类器，也不能在简化的属性所述的示例上使用该分类器。幸运的是，后者不常发生。在典型的设置中，用户会浏览数据、可视化数据、过滤数据、清除数据...，然后在原始数据上测试最后的模型。

1.22.2 示例

清除域 (Purge Domain) 通常在数据过滤之后出现，例如，在选择可视化的示例子集之后。

在上面的方案中，我们使用了 `adult.tab` 数据集：我们对它进行可视化并选择只包含 4/5 原始类的一部分数据。为了清除空的类，我们在进入 `Box Plot` 组件之前通过 `Purge Domain` 放置数据。`Box Plot` 只显示实际出现的四个类。为了查看数据清除的效果，我们取消选中 `Remove unused class values` 并在 `Box Plot` 上观察该操作的效果。



1.22-2 示例图片

1.23 保存



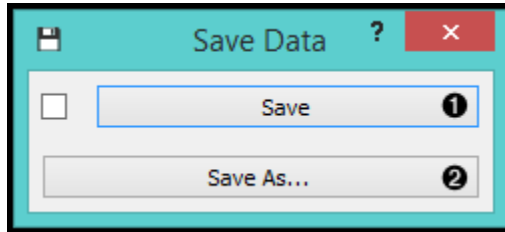
将数据保存到文件

1.23.1 描述

保存 (Save) 组件考虑在输入通道上提供的数据集并将其保存到具有指定名称的数据文件。

它可以保存到制表符分隔和逗号分隔的文件。

按照设计，这个组件并不是每次在输入上收到新信号时都保存数据，因为这样会不断（大多数情况下是无意的）覆盖文件，而是只有在设置了新文件名称或者用户按下 Save 按钮之后才保存数据。

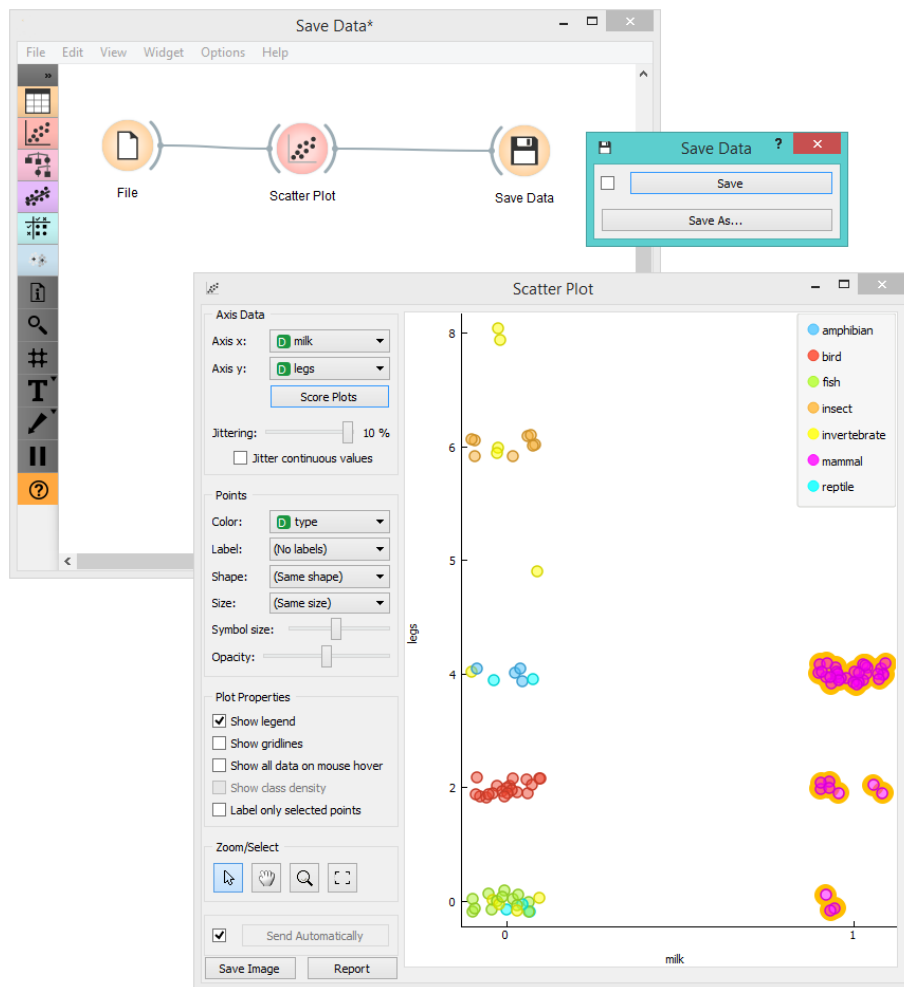


1.23-1 Save Data 窗口

1. 将数据保存到所选择的数据文件。
2. 指定要保存到的新数据文件。

1.23.2 示例

在下面的工作流程中，我们是用 Zoo 数据集。我们将数据加载到 Scatter Plot 组件中，在这个组件中我们选择数据实例的一个子集并将它们推向 Save 组件以将它们存储在某个数据文件中。



1.23-2 示例图片

2 可视化

 分类树图	 属性统计	 分布	 散点图
 筛法图	 马赛克显示	 线性投影	 热图
 维恩图	 轮廓图	 毕达哥拉斯树	 毕达哥拉斯森林
 CN2 规则查看器	 散点图		

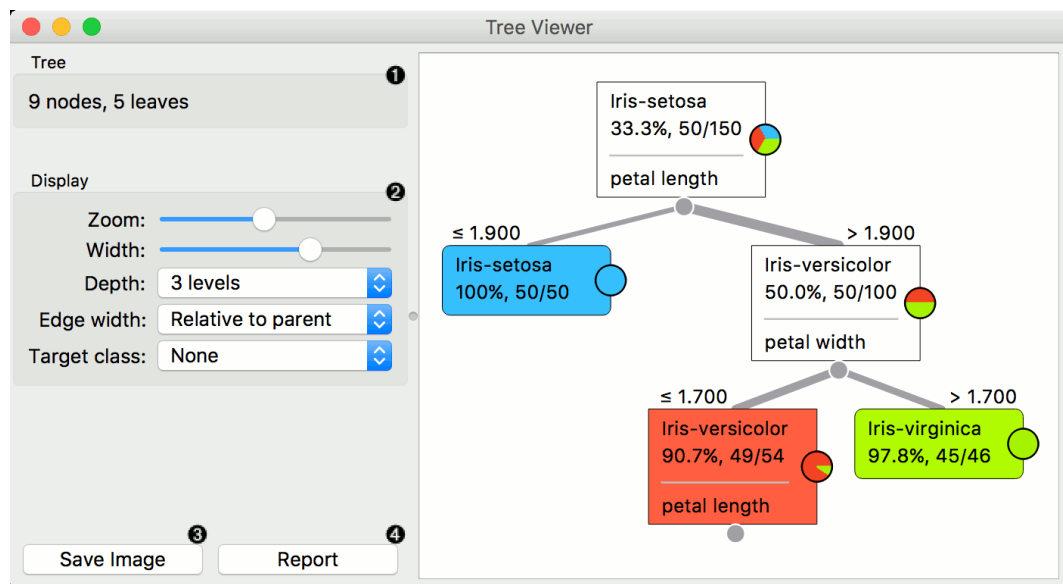
2.1 分类树图



分类和回归树的可视化

2.1.1 描述

这是一个多功能的组件，具有 2-D 可视化的分类和回归树。用户可以选择节点，指示小部件输出与节点相关联的数据，从而实现探索性数据分析。



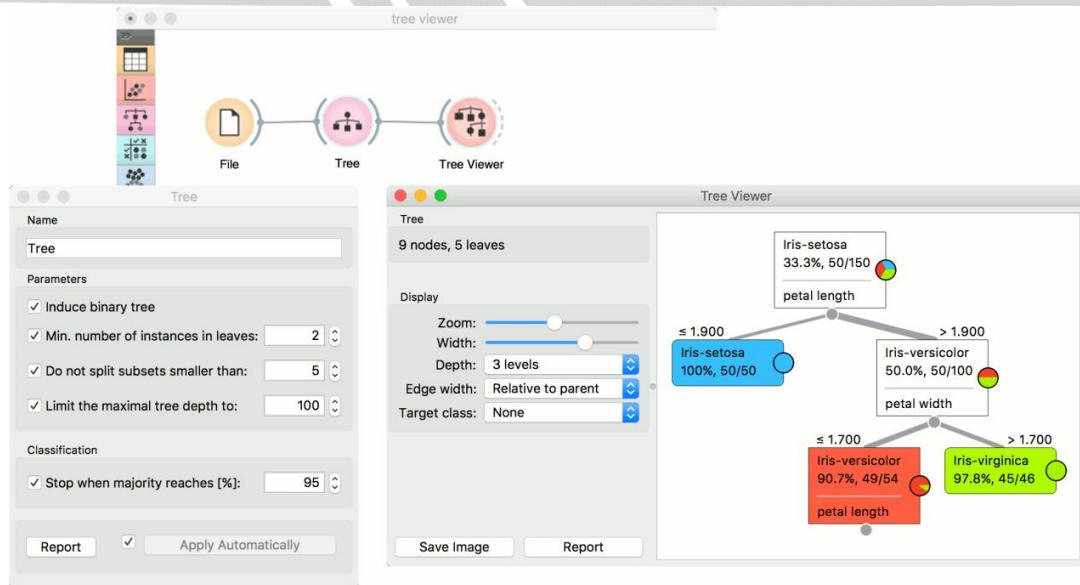
2.1-1 Tree Viewer 窗口

1. 输入信息。
2. 显示选项：
 - 放大或缩小。
 - 选择树宽。当它们悬停在其上时，节点显示信息气泡。
 - 选择树的深度。
 - 选择边宽。树形图中节点之间的边缘是根据选定的边缘宽度绘制的。
 - 如果选择固定，所有边缘将具有相等的宽度。

- 当选择 lative to root 时，边界的宽度将对应于相应节点上的实例的比例，与训练数据中的所有实例相对应。在这种选择下，当朝着树底移动时，边缘会越来越薄。
 - Relative to parent 使边缘宽度与节点中实例的实例的比例相对应。
 - 定义目标类，您可以根据数据中的类来更改。
3. 按保存图像 (Save image) 建的树形图保存为您的计算机作为.svg 或.png 文件。
 4. 制作报告。

2.1.2 示例

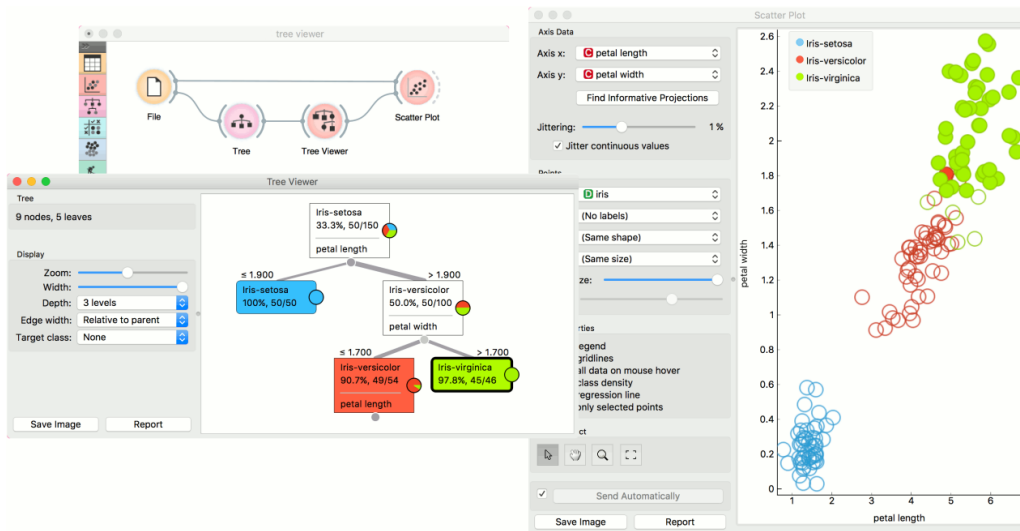
下面是一个简单的分类模式，我们已经阅读了数据，构建了决策树，并在 Tree Viewer 中查看了它。如果观察者和树都是打开的，tree viewer 算法的任何变化将立即影响可视化。因此，您可以使用此组合来探索感应算法的参数如何影响树的结构。



2.1-2 示例图片

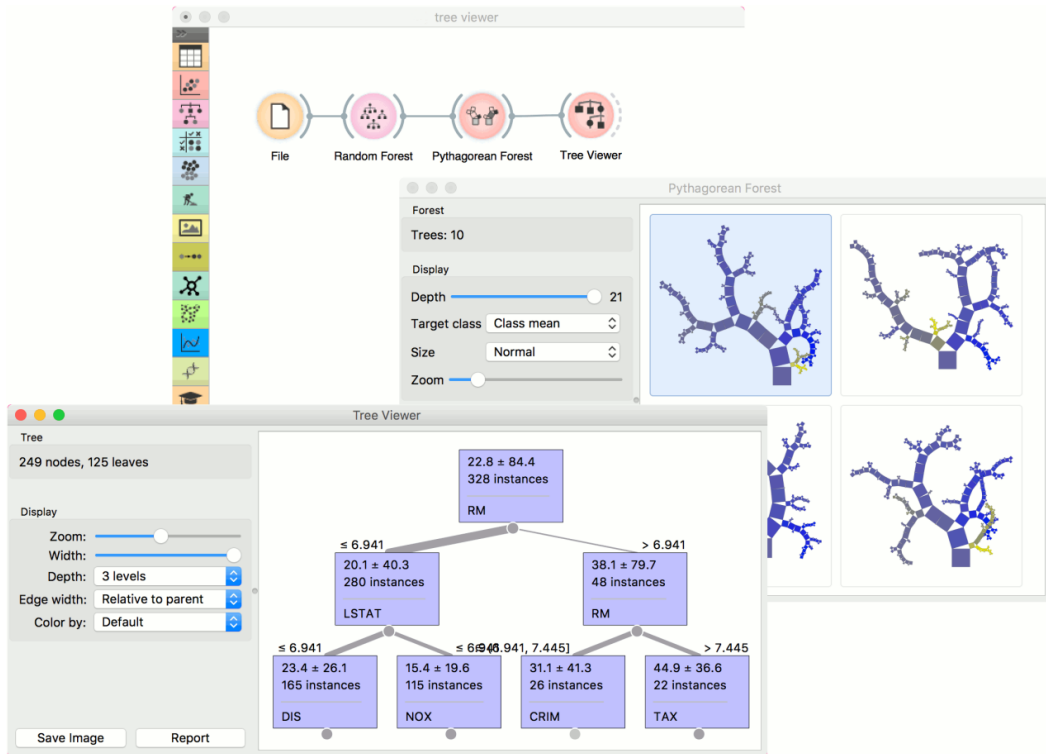
单击任何节点将输出相关的数据实例。这在下面的模式中进行了探索，显示了数据表和 Scatter plot 中的子集。确保树数据作为数据子集传递；这可以通过首先将 Scatter plot 连接到 File 组件，然后将其连接到 Tree Viewer 组件。所选数据将以粗体显示。

Tree Viewer 还可以导出标记的数据。将 Data Table 连接到 Tree Viewer，并将组件之间的链接设置为 Data 而不是 Selected Data。这将会将整个数据发送到 Data Table，并附加一个标记所选数据实例的元列（对于选定为是，剩余为否）。



2.1-3 示例图片

最后，Tree Viewer 也可用于可视化回归树。使用 housing.tab 数据集将 Random Forest 连接到 File 组件。然后将 Pythagorean Forest 连接到 Random Forest。在 Pythagorean Forest 中，选择一个您希望进一步分析并将其传递给 Tree Viewer 的回归树。为了可视化较大的树，尤其是回归，Pythagorean Tree 可能是一个更好的选择。



2.1-4 示例图片

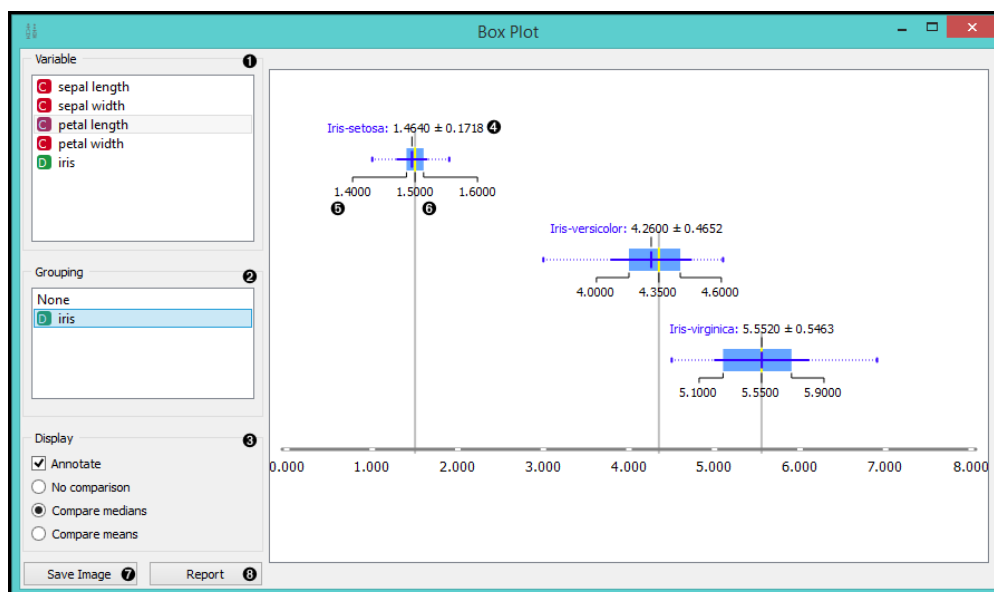
2.2 属性统计



显示属性值的基本分布

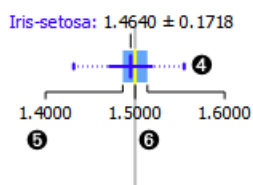
2.2.1 描述

属性统计 (Box Plot) 显示属性值的分布。最好使用这个组件查看任何新的数据，以便快速发现任何异常，例如重复值（即灰色的值）、离群点以及其他类似的异常情况。



2.2-1 Box Plot 窗口

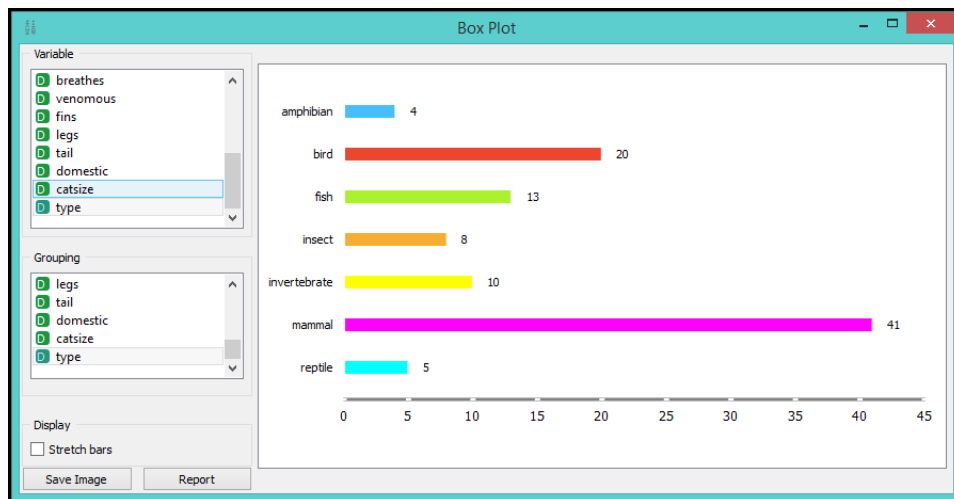
1. 选择要绘制的变量。
2. 选择 Grouping 来观察 box plots 的分类显示。
3. 当实例按类分组时，可以更改显示模式。注释框将显示结束值，平均值和中位数，而比较中位数和比较平均值自然会比较类组之间的选定值。



这个组件对于连续属性显示：

4. 平均值（深蓝色垂直线）。
5. 平均值的标准偏差的边界值。蓝色突出显示的区域是平均值的整个标准偏差。
6. 中位数（黄色垂直线）。细蓝线表示第一（25%）和第三（75%）分位数之间的区域，而细虚线表示整个值的整个范围（从所选参数的数据集中的最低值到最高值）。
7. 保存图片。
8. 制作报告。

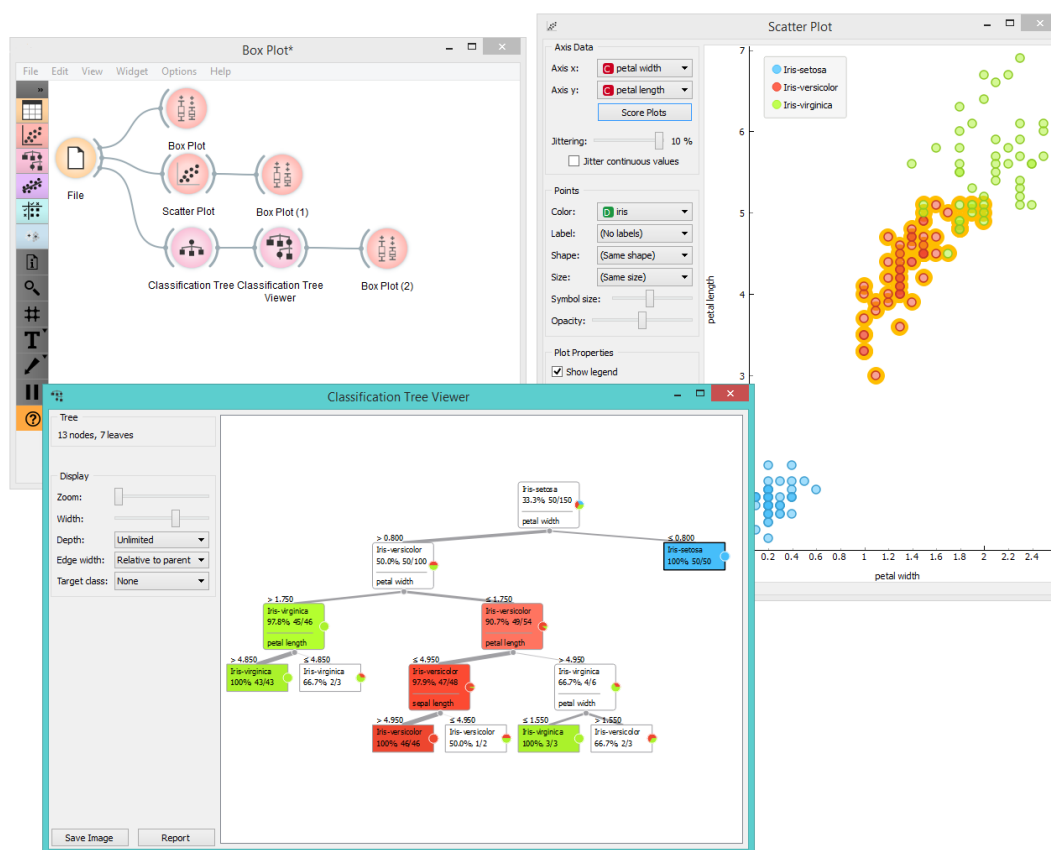
对于离散属性，条形表示具有每个特殊属性值的示例数量。该图显示 Zoo 数据集中不同动物类型的数量：4 个哺乳动物、13 条鱼等等。



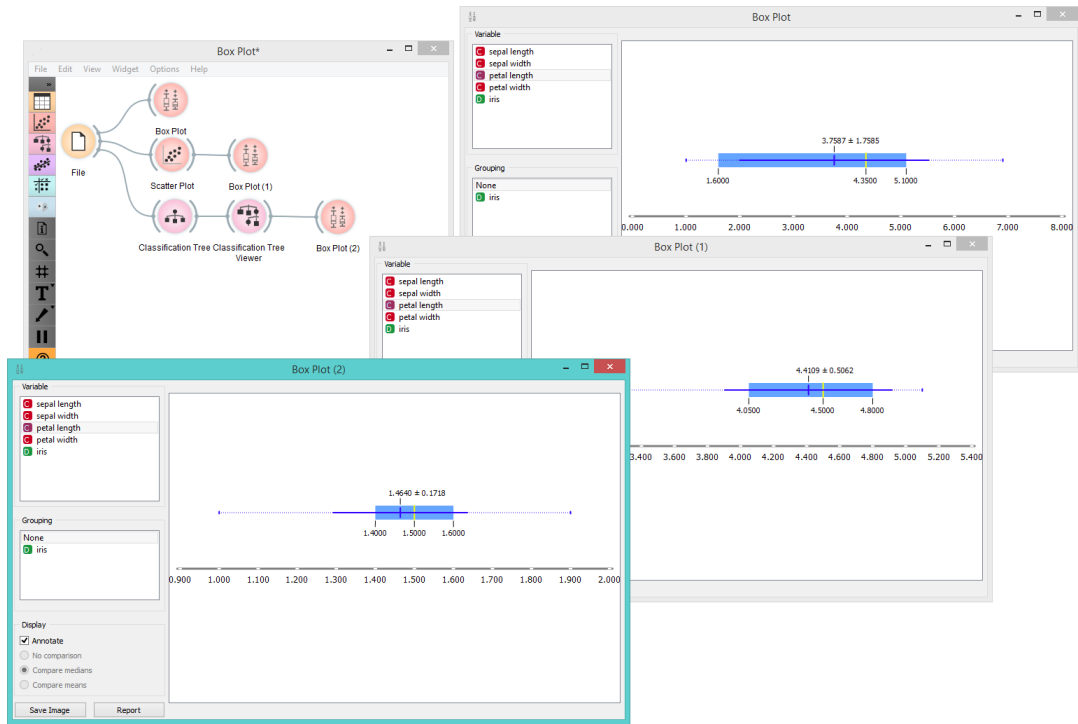
2.2-2 Box Plot 窗口（离散属性）

2.2.2 示例

属性统计 (Box Plot) 通常紧接着在 File 组件之后使用,用于观察数据集的统计属性。它还可用于查找特定数据集的属性,例如,在另一个组件(如散点图)中手动定义的一组示例,或属于某些簇或某个分类树节点的示例,如下面的方案所示。



2.2-3 示例图片



2.2-4 示例图片

2.3 分布



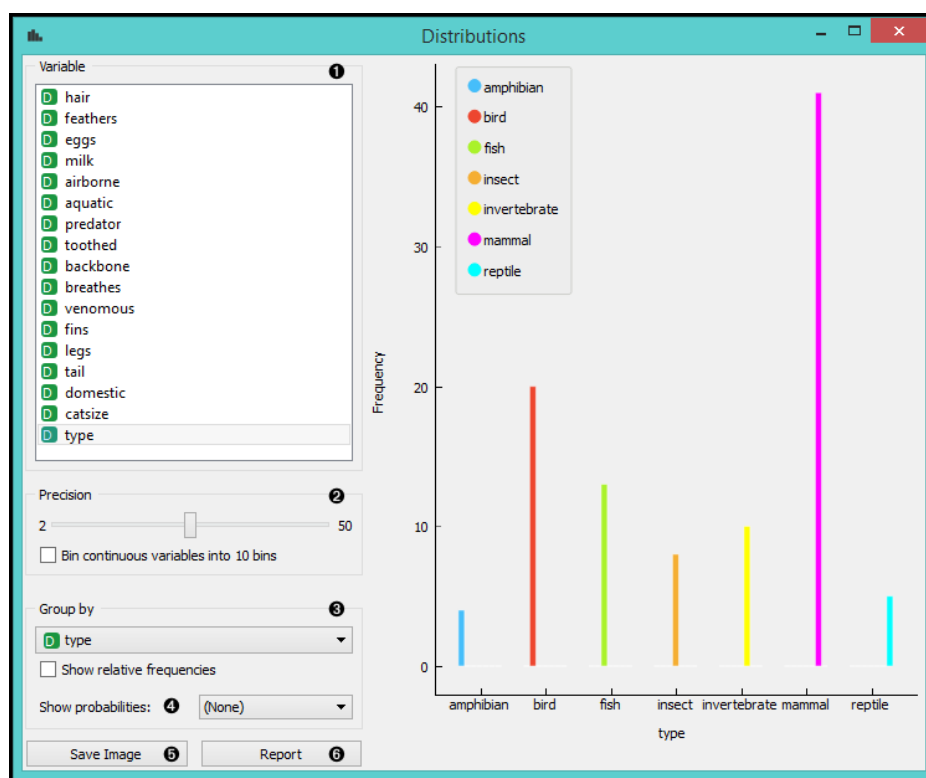
显示单个属性的值分布

2.3.1 描述

分布 (Distributions) 显示离散或连续属性的值分布。如果数据包含类，则会在该类上调整分布。

对于离散属性，这个组件显示的图形会显示每个属性值在数据中出现的次数(即其中包含的数据实例数)。如果数据包含类变量，则还会显示每个属性值的类分布(如上面的图片所示)。

为了创建这个图，我们使用了 Zoo 数据集。

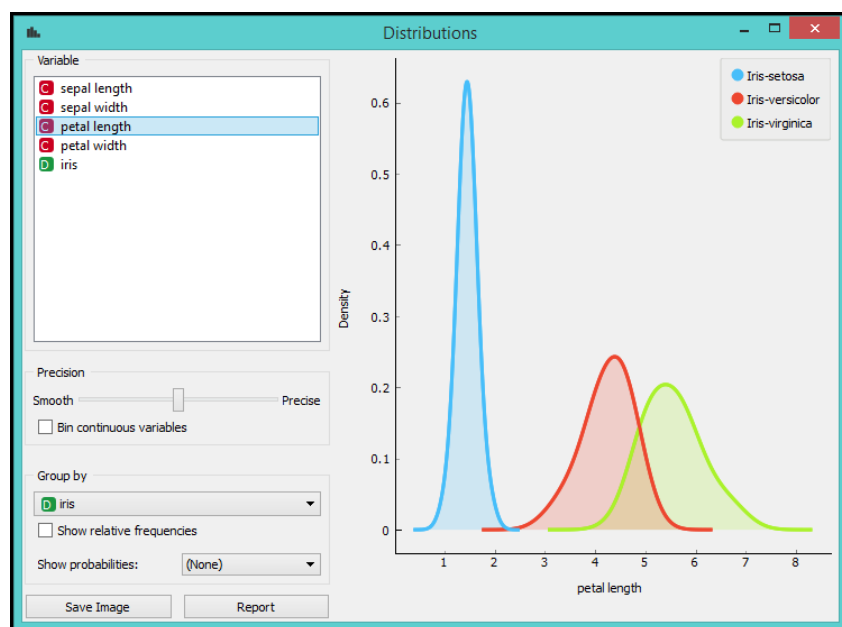


2.3-1 Distributions 窗口 (离散属性)

1. 分布显示的变量列表。

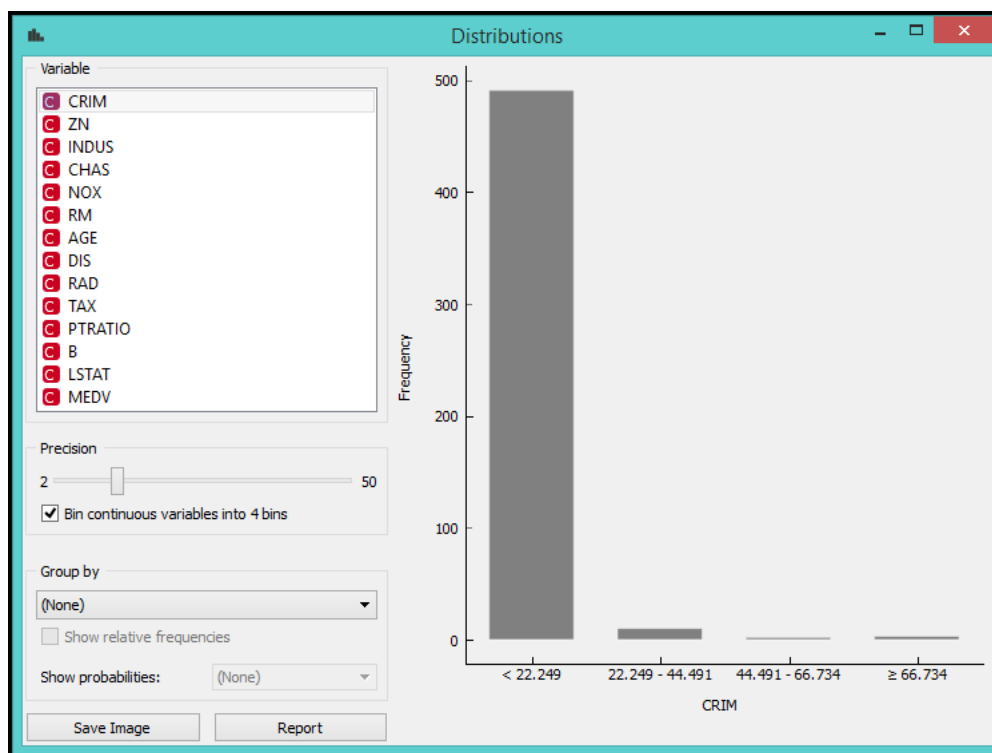
2. 如果 Bin continuous variables 被打钩，这个组件会通过指定连续变量的间隔来使他们离散化。间隔的数量是由精确尺度决定的。另外，可以设置连续变量的分布曲线的平滑度。
3. 这个组件可以被要求仅针对某些类（分组）的实例显示值分布。 Show relative frequencies 将按数据集的百分比缩放数据。
4. 显示概率。
5. Save image 可以将图片以.svg 或.png 格式来保存在电脑里。
6. 制作报告。

对于连续属性，属性值会被展示成函数图。通过高斯核密度估计获得连续属性的类概率，而使用精度栏（平滑或精确）设置曲线的外观。为了达成这个目的，我们使用 Iris 数据集。



2.3-2 Distributions 窗口（连续属性）

在无类域中，条形显示为灰色。这里我们选中 Bin continuous variables into 10 bins，它将变量分配到 10 个间隔，并将这些间隔的平均值显示为直方图（见上文）。我们使用 Housing 数据集。



2.3-3 Distributions 窗口（无类域）

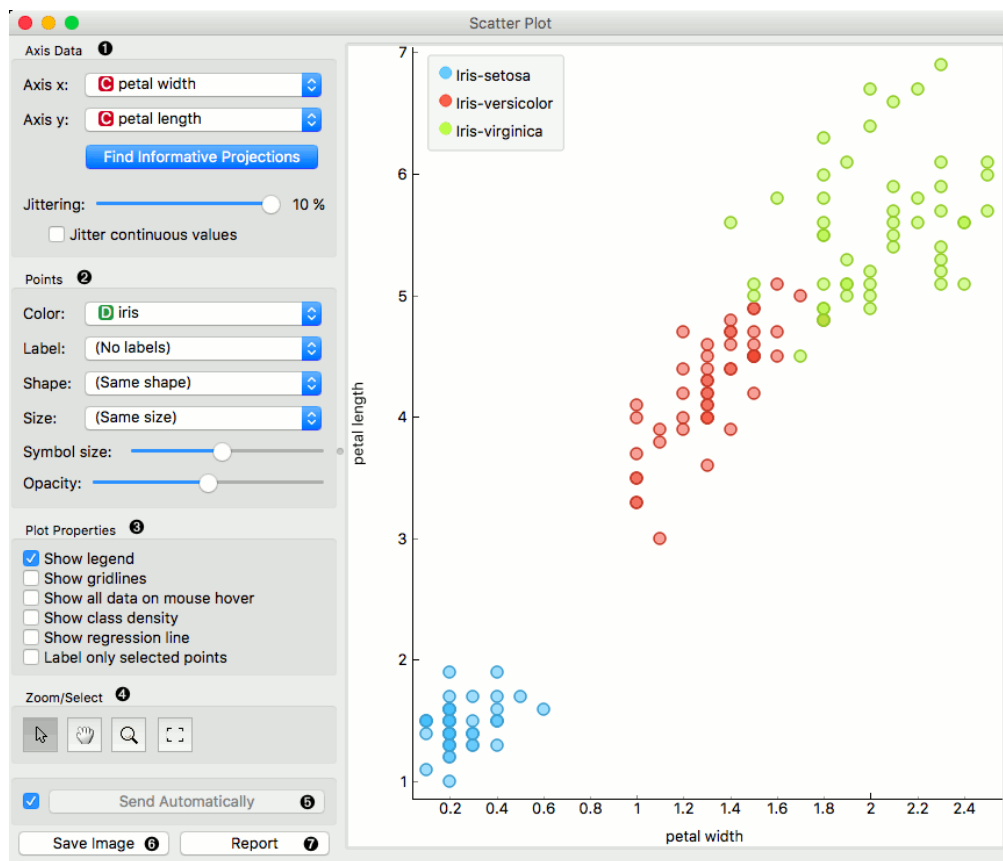
2.4 散点图



具有探索性分析和智能数据可视化增强的标准散点图可视化

2.4.1 描述

散点图 (Scatter plot) 组件为连续和离散值的属性提供了一个标准的二维散点图可视化。数据显示为点的集合，具有 X-axis attribute 值的每个点确定它在水平轴上的位置，具有 Y-axis attribute 值的每个点确定它在垂直轴上的位置。图形的各种属性（如颜色、大小和形状）通过这个组件的 Main 窗格中的相应设置类控制，而其他属性（如图例、轴标题、最大点大小以及抖动）则在 Settings 窗格中设置。下面的图片显示了 Iris 数据集的散点图，点的大小与 sepal width 属性的值成正比，颜色与类属性的颜色相匹配。



2.4-1 Scatter Plot 窗口

1. 选择 x 轴和 y 轴的属性。使用 Rank Projections 优化你的投影。此功能通过平均分类精度对属性对进行分值，并返回最高评分对，同时进行可视化更新。设置 jittering 以防止点重叠。如果 Jitter continuous values 被选中，连续的实例将被分散。

2. 设置显示点的颜色（你将为离散值和连续的灰度点设置颜色）。设置标签，形状和大小以区分点。为所有数据点设置符号大小和不透明度。设置所需的颜色刻度。

3. 调整图表属性：

Show legend 显示右侧的图例。 点击并拖动图例将其移动。

Show gridlines 显示图形后面的网格。

Show all data on mouse hover 如果光标位于点上，则启用信息气泡。

Show class density 根据类型来给图形上色（见下面的屏幕截图）。

Show regression line 为连续的属性绘制回归线。

Label only selected points 可以选择单个数据实例并进行标记。

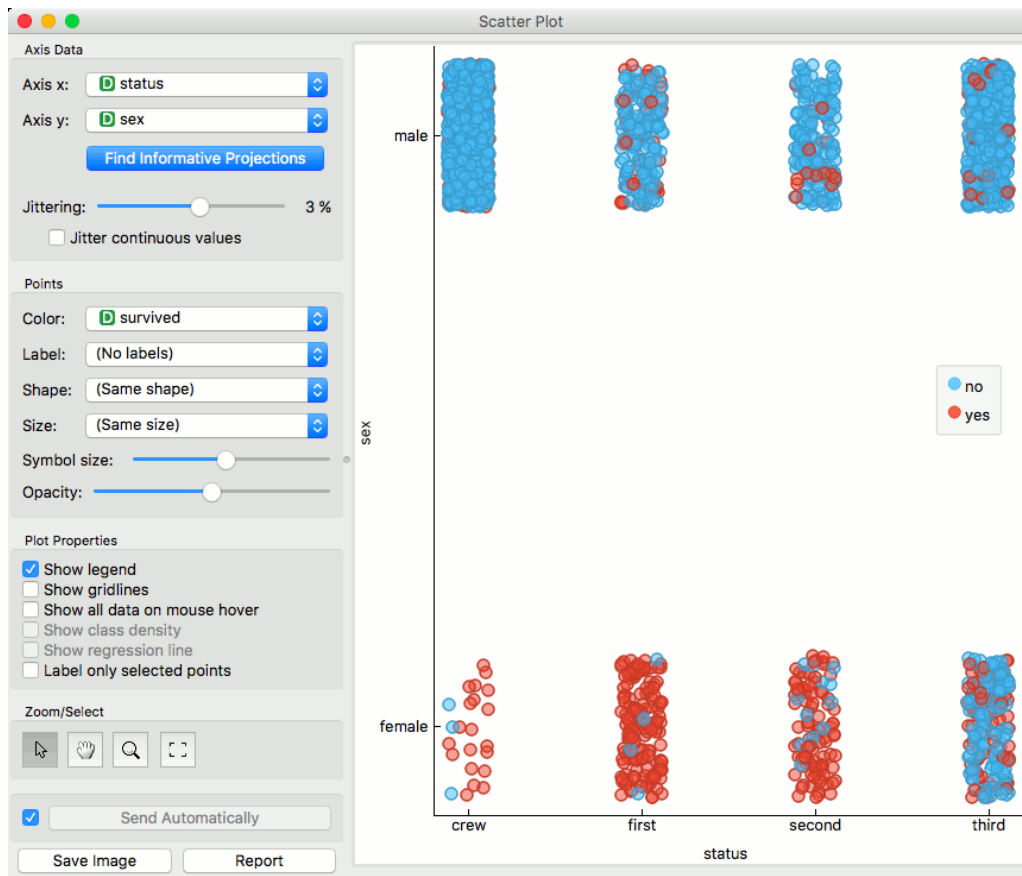
4. 选择，平移，缩放，适当的缩放以适应浏览图形的选项。数据实例的手动选择用作非角/自由选择组件。双击移动投影。滚动来放大或缩小。

5. 如果选中 Send automatically，更改会自动上传，否则，点击 Send。

6. Save Image 将以 svg 或 png 格式在你的电脑上储存图片。

7. 制作一报告。

如果是离散属性，则应使用抖动 (Jittering options) 来避免这两个轴上相同值的点的重叠，并且绘制特定区域中的点的密度图，以便更好地对应于具有该特定组合值的数据的密度。作为此类图的一个示例，下面显示了 Titanic 数据的散点图，它报告了乘客的性别以及旅行类；如果没有抖动，散点图将只显示八个不同点。



2.4-2 Scatter Plot 窗口 (离散属性)

下面是 the Show class density 和 Show regression line boxes 被选中时的图形。



2.4-3 Scatter Plot 窗口 (显示分类密度)

2.4.2 智能数据可视化

如果数据集有很多属性，则无法手动扫描所有属性对来查找有趣的散点图。Mining 通过组件中的 Find Informative Projections 选项实现智能数据的可视化。优化的任务是查找具有不同类标签的实例被很好地分离的散点图投影。

使用这种方法，在组件中打开 Find Informative Projections 选项，然后在子窗口中选择 Start 选项。该功能将通过平均分类精度得分返回属性对的列表。

下面是一个例子来说明排名的效用。第一个散点图投影在图中被设置为默认的 sepal width 到 sepal length(为了简便 ,我们使用 Iris 数据集)。在运行 Find Informative Projections 优化后 ,散点图转化成更好的投影。

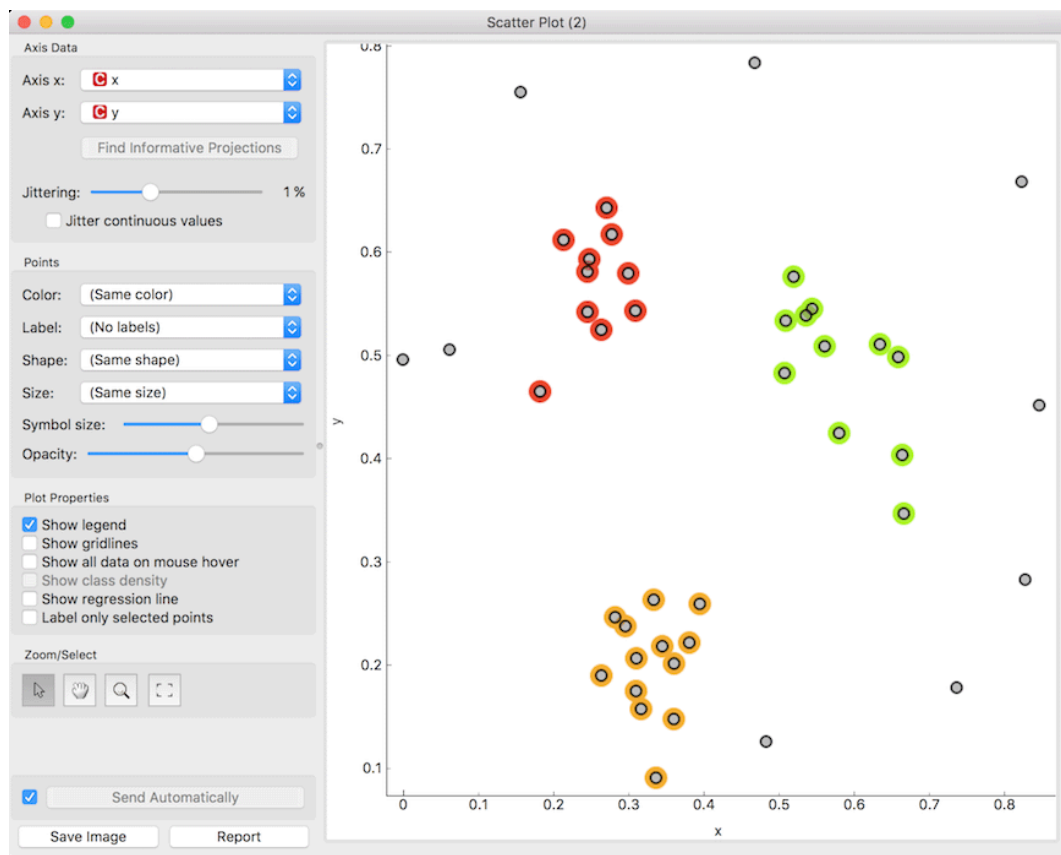


2.4-4 示例图片

2.4.3 选择

选择可以用于数据中的手动定义的子集。在选择数据实例时使用 Shift 修饰符将其放入一个新组中。Shift + Ctrl (或者 Shift + Cmd 在 macOS 系统中)可以将数据实例添加到最后一组。

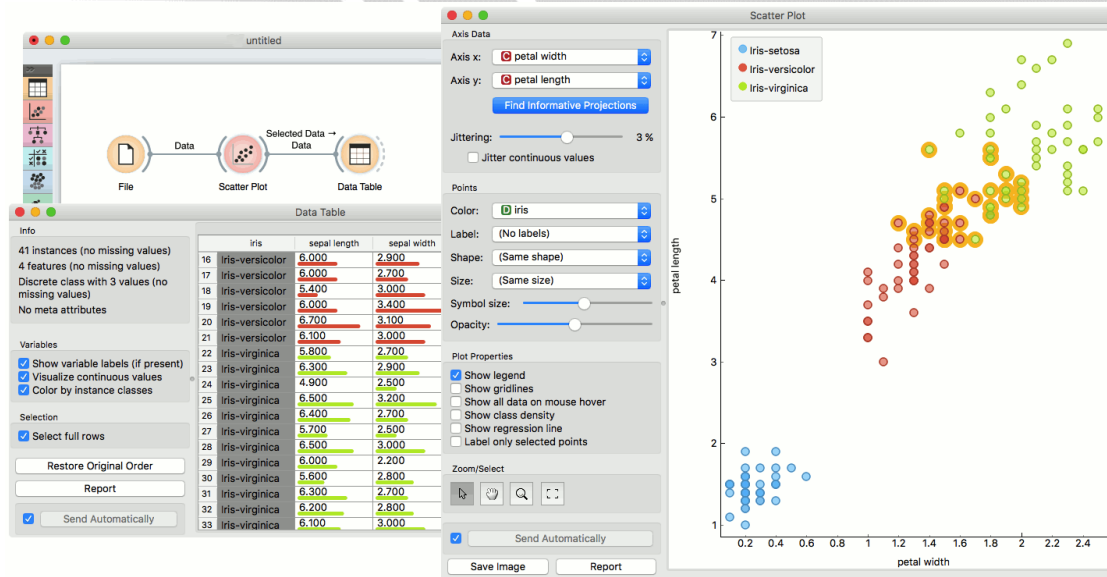
信号数据输出一个包含组索引的附加列的数据表。



2.4-5 示例图片

2.4.4 探索性数据分析

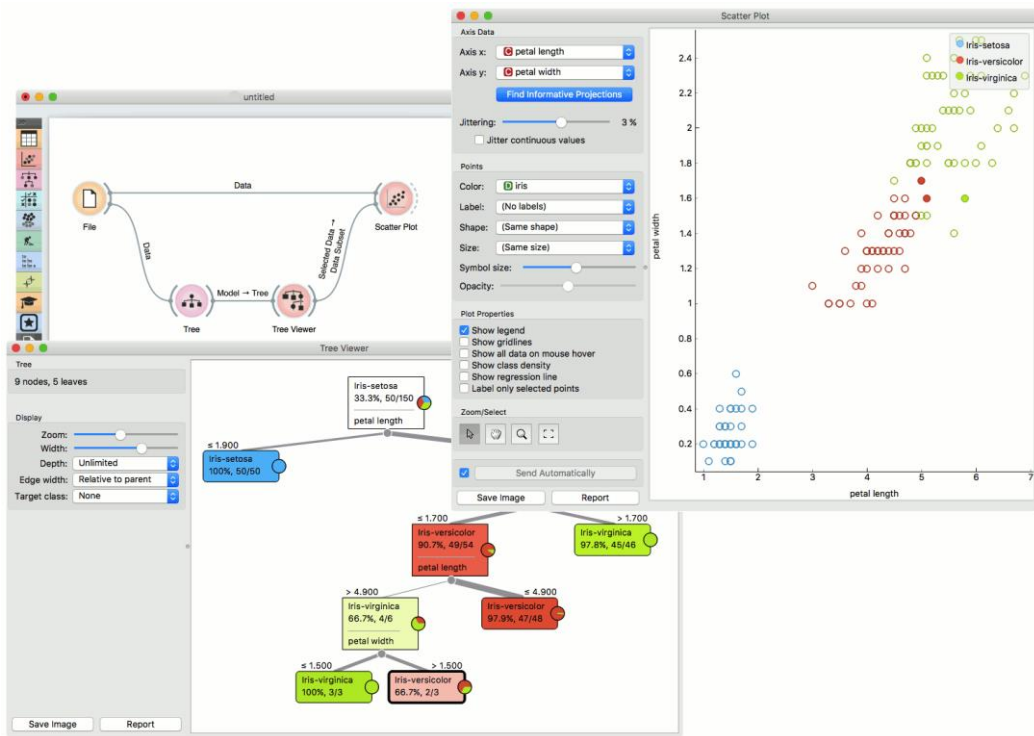
和 Mining 的其他组件一样，Scatter plot 支持图形的放大和缩小以及实例的手动选择。这些功能在组件的左下角可用。默认的组件是 Select，它选择矩形区域内的数据实例。Pan 可以移动窗格内的散点图。使用 Zoom，你可以使用鼠标滚动来放大和缩小，而 Reset zoom 会将可视化重置为最佳大小。在下面的简单实例中，我们从矩形区域中选择数据实例并把它们发送到 Data Table 组件中。注意由于一些数据实例的重叠（他们有两个属性使用相同的值），它不会显示所有的 52 个数据。



2.4-6 示例图片

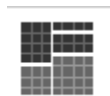
2.4.5 示例

散点图可以很好地与输出所选数据实例列表的其他组件组合。例如，下面所示的分类树和散点图的组合，成为了一个很好地显示与所选分类树节点相关的数据实例的探索性组件（单击分类树的任何节点会将所选择的数据实例集发送到散点图，从而更新可视化并用填充的符号标记所选择的实例）。



2.4-7 示例图片

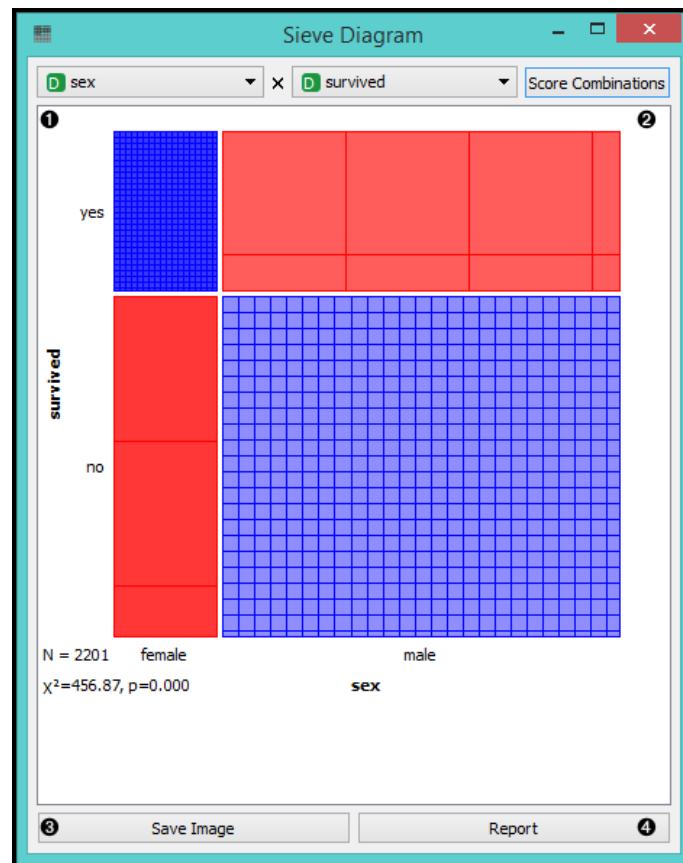
2.5 筛法图



为一对属性绘制一个筛法图

2.5.1 描述

筛法图是用于使双向列联表中的频率可视化并将其与独立性假设下的预期频率进行比较的图形方法。筛法图是由 Riedwyl 和 Schüpbach 在 1983 年的技术报告中提出的，后来被称为拼花图。在该显示中，每个矩形的面积与期望的频率成正比，并且观察到的频率由每个矩形中的正方形数量表示。观察到的频率和预期频率之间的差异（与标准的 Pearson 残差成正比）显示为阴影的浓度，使用颜色来指示偏离独立性的偏差是正（蓝色）还是负（红色）。

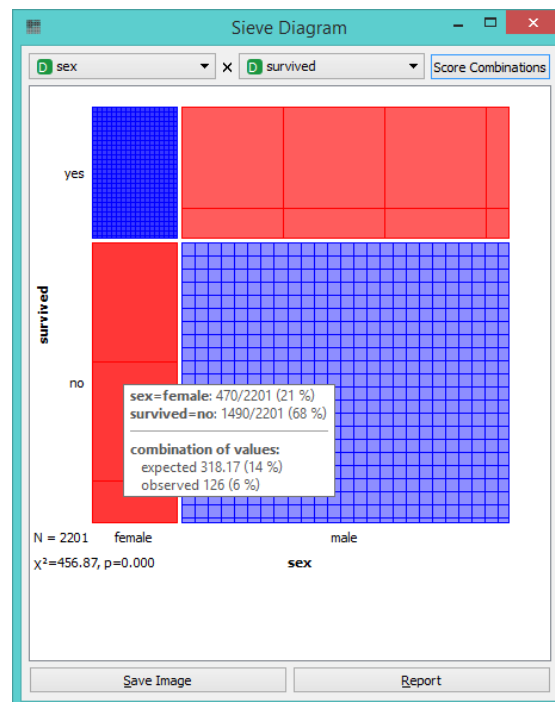


2.5-1 Sieve Diagram 窗口

1. 选择要在筛选图中显示的属性。

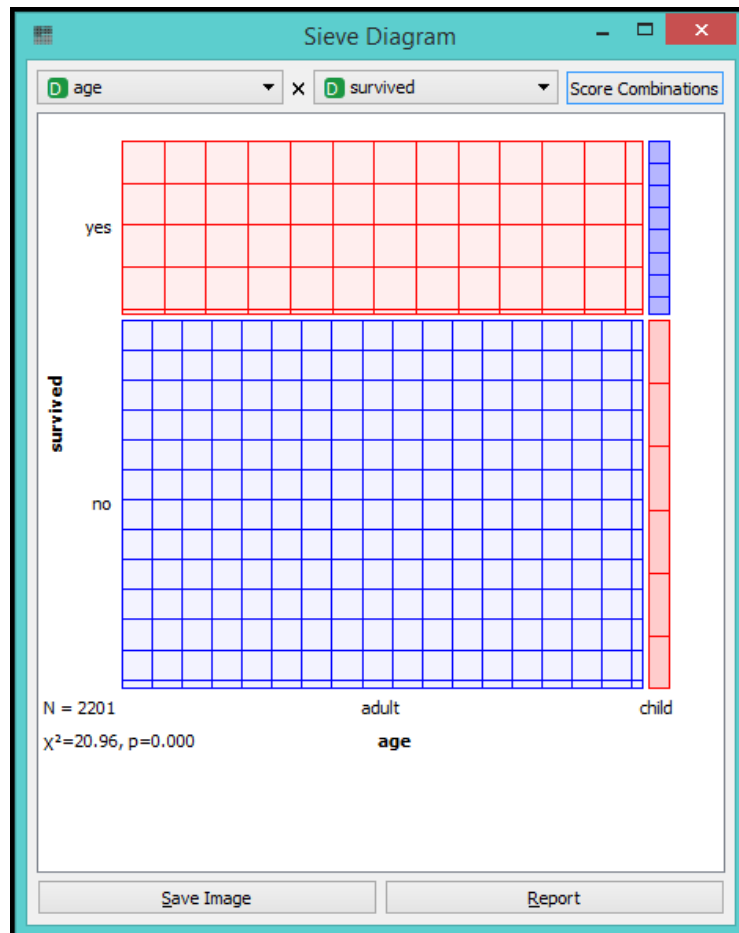
2. 分数组合使您能够找到最佳可能的属性组合。
3. Save Image 将以 svg 或 png 格式在你的电脑上储存图片。
4. 制作报告。

下面的图片显示了 Titanic 数据集以及属性 sex 和 survived 的筛法图 (后者实际上是此数据集中的一个类属性)。该图显示两个变量高度相关,因为在所有四个象限中观察到的和期望的频率之间存在显著差异。例如,在图中突出显示的,女性乘客没有在事故中幸存下来的几率远远低于预期 (0.06 比 0.15)。



2.5-2 示例图片

具有有趣关联的属性对具有强阴影，例如上述图片中所示的图。为了进行对比，下面显示了最无趣的属性对（age 与 survival）的筛法图。

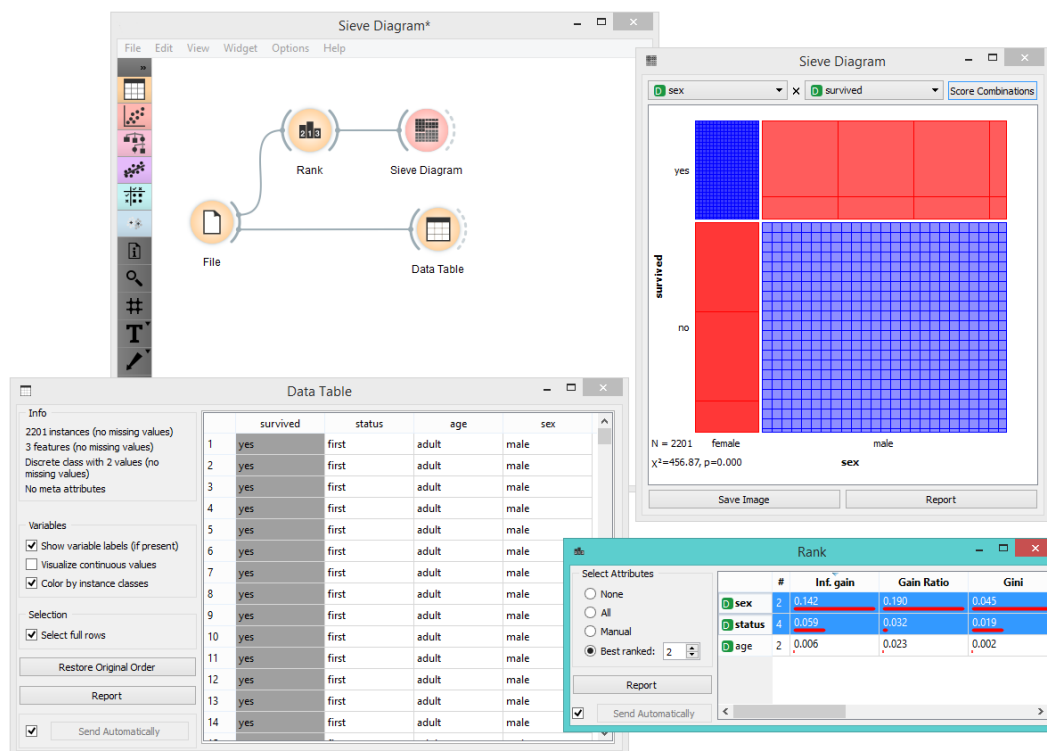


2.5-3 示例图片

2.5.2 示例

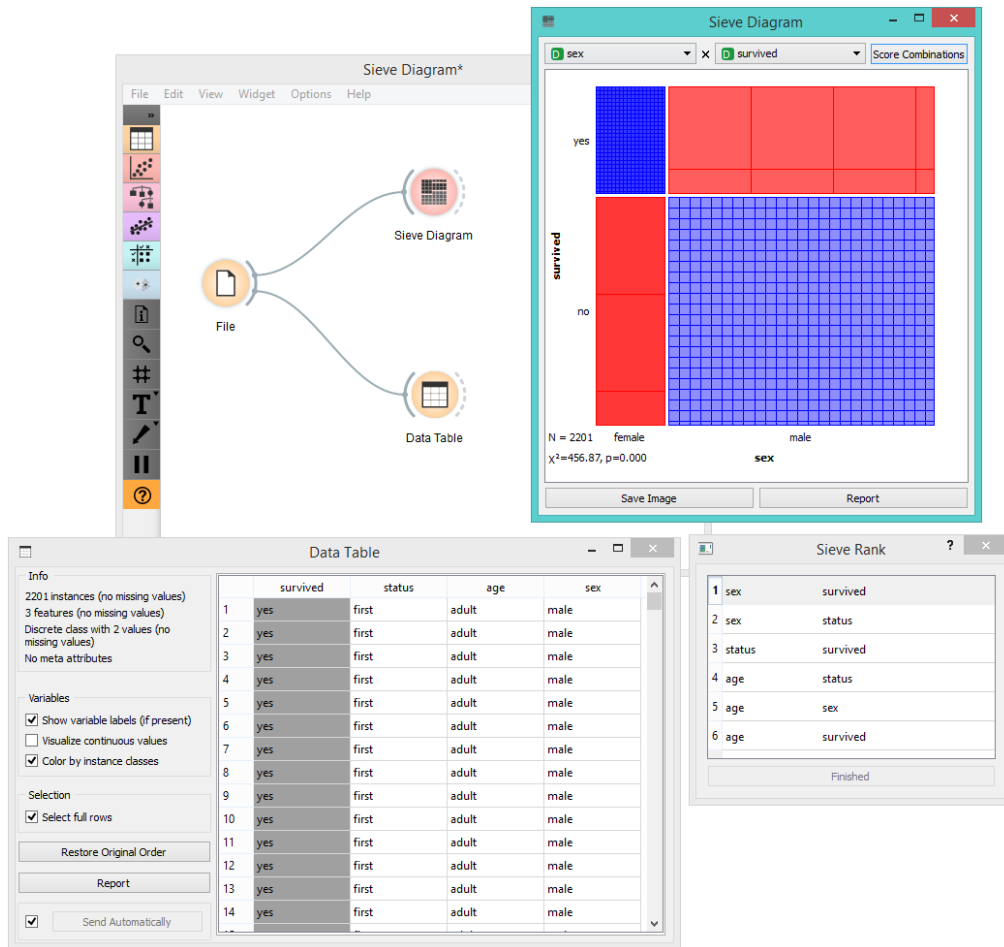
下面，我们看到一个使用 Titanic 数据集的简单模式，我们使用 Rank 组件来选择最佳属性（信息增益最大，增益比或 gini 指数最高的属性），并将它们提供给 Sieve Diagram。这

显示了两个最佳属性的筛选图，在我们的例子中是性别和状态。我们看到泰坦尼克号的生存率对于一流的女性来说非常高，女性船员的生存率非常低。



2.5-4 示例图片

Sieve Diagram 还具有“分数组合”选项，这使得属性的排名更加容易。



2.5-5 示例图片

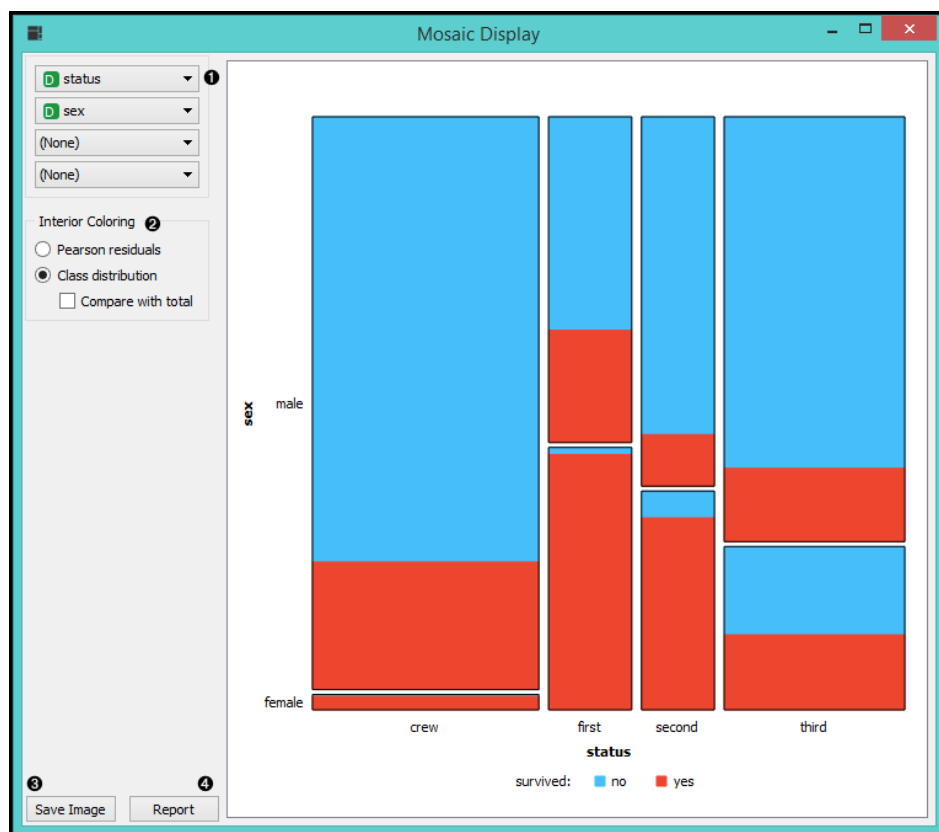
2.6 马赛克图



显示多路(n-way)表马赛克图

2.6.1 描述

马赛克图是一种采用多路列联表（即每个单元格对应于 n 个属性的不同值组合的表）可视化数量的图形方法。该方法由 Hartigan 和 Kleiner 提出并由 Friendly 进行了扩展。马赛克图中的每个单元格都对应于列联表中的一个单元格。如果数据包含类属性，马赛克图将显示类分布。



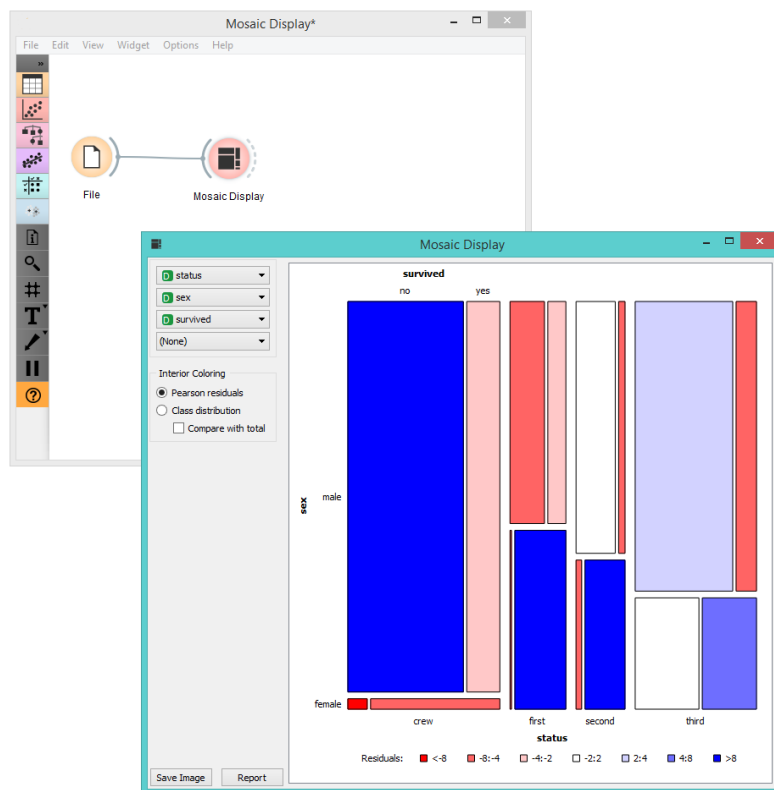
2.6-1 Mosaic Display 窗口

1. 选择您想要查看的变量。
2. 选择内部着色。您可以根据分类对内部进行着色，也可以使用 Pearson residual，这是观测值和拟合值之间的差值除以观测值的标准偏差的估计值。如果 Compare to total 被选中，则比较所有实例。

3. Save Image 将以 svg 或 png 格式在你的电脑上储存图片。
4. 制作报告。

2.6.2 示例

我们加载了 titanic 数据集，并将其连接到 Mosaic Display 组件。我们决定把重点放在三个变量，即地位，性别和生存。我们根据 Pearson residuals 对内部进行着色，以证明观察值和拟合值之间的差异。



2.6-2 示例图片

我们可以看到，男性和女性的存活率明显偏离拟合值。

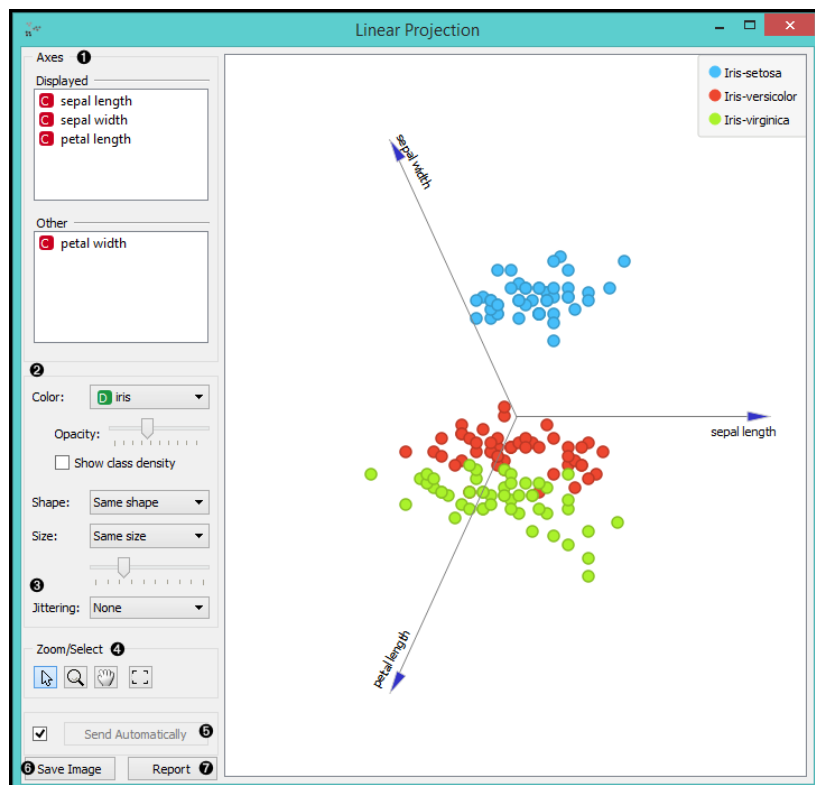
2.7 线性投影



具有探索性数据分析和智能数据可视化增强的各种线性投影方法

2.7.1 描述

此组件显示类标记数据的线性投影。首先考虑下面所示的 Iris 数据集的投影。请注意, sepal width 和 sepal length 已经将 Iris setosa 与其他两种分离, 而 petal length 是将 Iris versicolor 与 Iris virginica 分离的属性。



2.7-1 Linear Projection 窗口

1. 显示的投影中的轴和其他可用的轴。
2. 设置显示点的颜色 (您将获得离散值的彩色点和连续的灰度点)。 设置不透明度 , 形状和大小以区分实例。
3. 设置 jittering 以防止点重叠 (特别是对于离散属性)。
4. 选择 , 平移 , 缩放 , 适当的缩放以适应浏览图形的选项。数据实例的手动选择用作非角/自由选择组件。双击移动投影。滚动来放大或缩小。
5. 如果选中 Send automatically , 更改会自动上传 , 否则 , 点击 Send。
6. Save Image 将以 svg 或 png 格式在你的电脑上储存图片。
7. 制作报告。

2.7.2 示例

线性投影组件与其他可视化组件一样工作。下面我们将它连接到 File 组件 , 以查看投影在二维平面上的集合。然后我们选择了进一步分析的数据 , 并将其连接到 Data Table 组件 , 以查看所选子集的详细信息。



2.7-2 示例图片

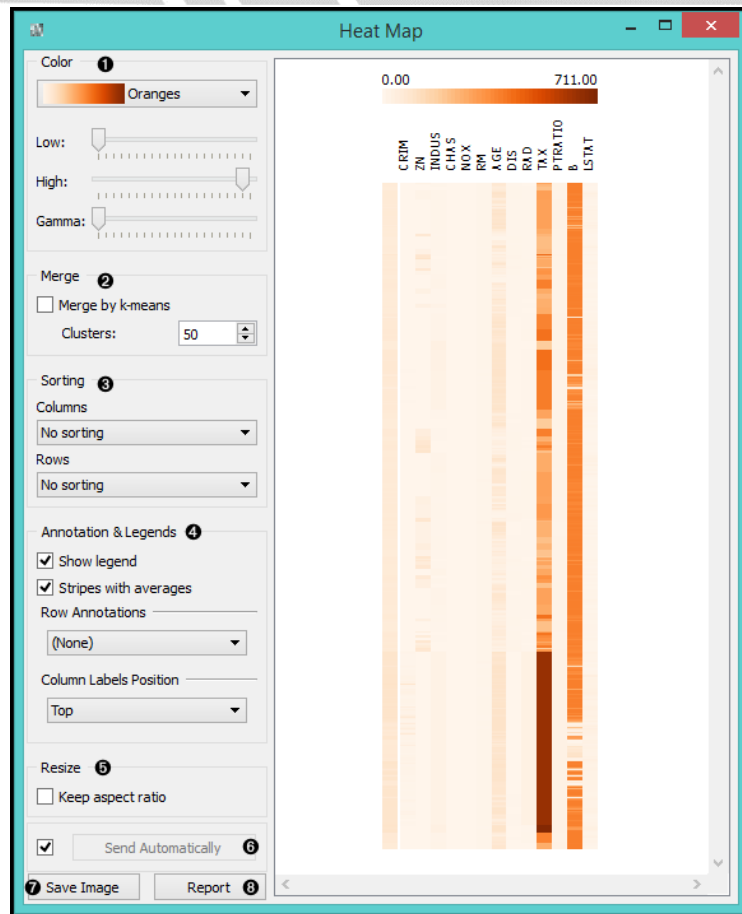
2.8 热图



为一对属性绘制热图

2.8.1 描述

热图是一种用于通过双向矩阵中的类可视化属性值的图形方法。它只适用于包含连续变量的数据集。值由颜色表示：某个值越高，表示的颜色越暗。通过组合 x 和 y 轴上的类和属性，我们看到属性值在哪里是最强的，哪里最弱，从而使我们能够找到每个类的典型特征（离散）或值范围（连续）。



2.8-1 Heat Map 窗口

1. 配色方案图例。低和高是调色板的阈值（对于具有低值的属性为低，对于具有高值的属性，为低）。
2. 合并数据。
3. 排序列和行：- 无排序（列出数据集中的属性）- 聚类（通过相似性聚类数据）- 具有有序树叶的聚类（最大化相邻元素的相似性之和）。
4. 在注释和图例（Annotation & Legend）设置图形中显示的内容。- 如果显示图例（Show legend）被勾选，地图上方将显示一个颜色图表。- 如果带有平均值的条纹（Stripes with averages）被勾选，则具有属性平均值的新行将显示在左

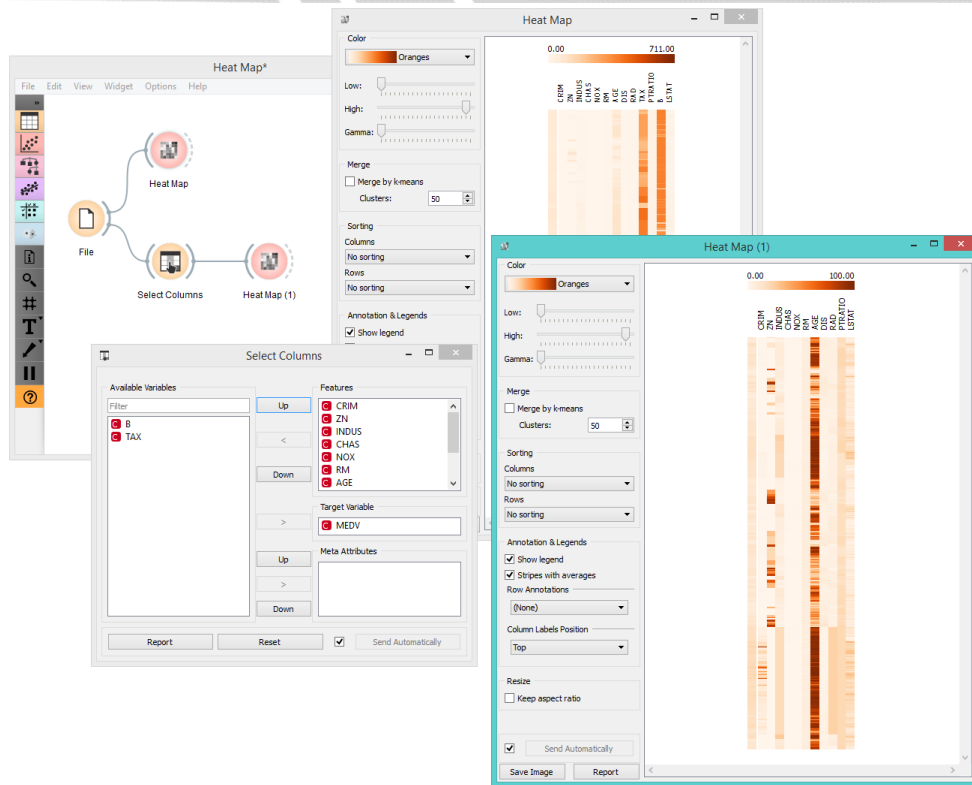
侧。 - Row Annotations 为每个实例添加注释。 - 列标签位置 (Column Label Positions) 将列标签放置在选定的位置 (无 , 顶部 , 底部 , 顶部和底部) 。

5. 如果勾选 “保持宽高比” (Keep aspect ratio) , 则每个值将以正方形显示 (与地图成比例) 。
6. 如果勾选 “自动发送” (Send Automatically) , 则自动进行更改。或者 , 单击发送。
7. 保存图像以.svg 或.png 格式将图像保存到计算机。
8. 制作报告。

2.8.2 示例

下面的 Heat Map 显示了 Housing 数据集的属性值。上述数据集涉及波士顿郊区的房屋价值。我们在地图上看到的第一件事是 “B” 和 “Tax” 属性 , 它们是深橙色中唯一的两个。

“B” 属性提供有关城市黑人比例的信息 , “Tax” 属性告诉我们每 10,000 美元的全值房产税。为了获得更清晰的热图 , 我们然后使用 Select Columns 组件 , 并从数据集中删除两个属性。然后我们再次将数据提供给 Heat map。新的投影提供了额外的信息。通过删除 “B” 和 “Tax” , 我们可以看到其他决定因素 , 即 “Age” 和 “ZN” 。“Age” 属性提供了有关 1940 年之前建成的自住单位比例的信息 , “ZN” 属性告诉我们每个城镇的非零售商业用地的比例。



2.8-2 示例图片

Heat Map 组件是发现数据中相关功能的组件。通过删除一些更显著的功能，我们发现了隐藏在后台的新信息。

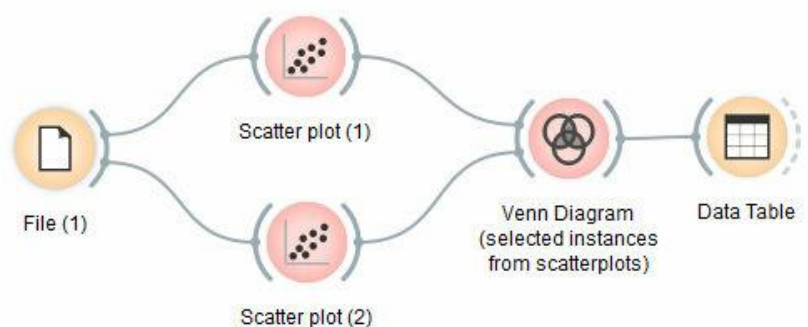
2.9 维恩图



绘制两个或多个数据子集的维恩图

2.9.1 描述

Venn Diagram 组件显示数据集之间的逻辑关系。该投影显示由不同颜色的圆圈表示的两个或多个数据集。交点是属于多个数据集的子集。要进一步分析或可视化子集，请单击交叉点。



2.9-1 Venn Diagram 窗口

1. 输入数据的信息。
2. 选择用于比较数据的标识符。
3. 如果要删除重复项，请勾选输出重复项 (Output duplicates) 。
4. 如果自动提交 (Auto commit is on) 开启，更改将自动传送到其他组件。

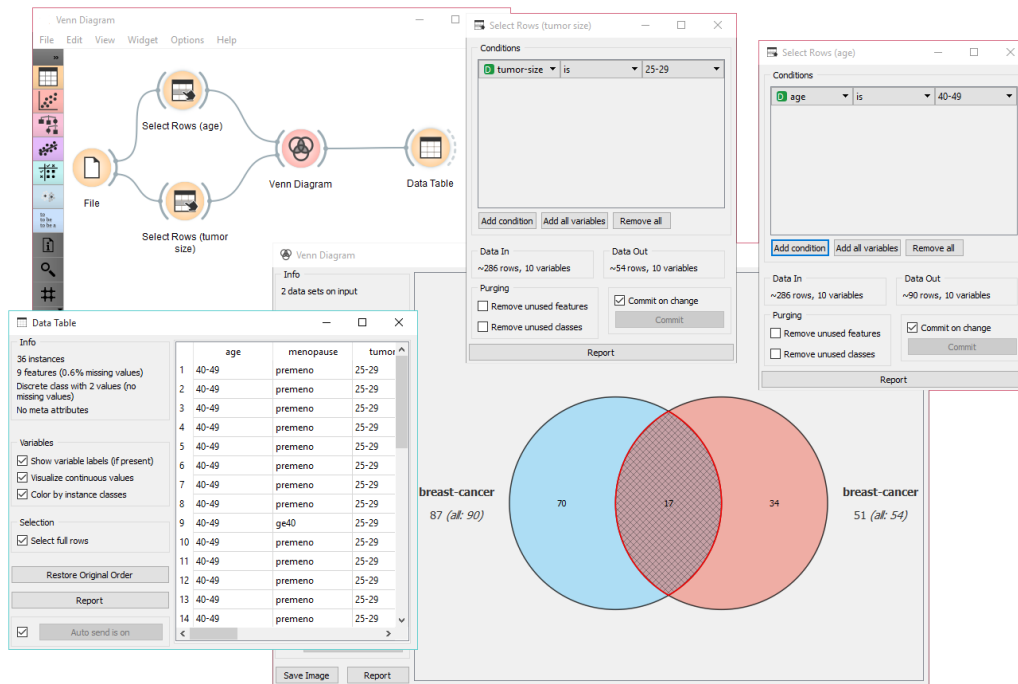
或者，单击提交 (Commit) 。

5. 保存图像将创建的图像以.svg 或.png 格式保存到计算机。

6. 制作报告。

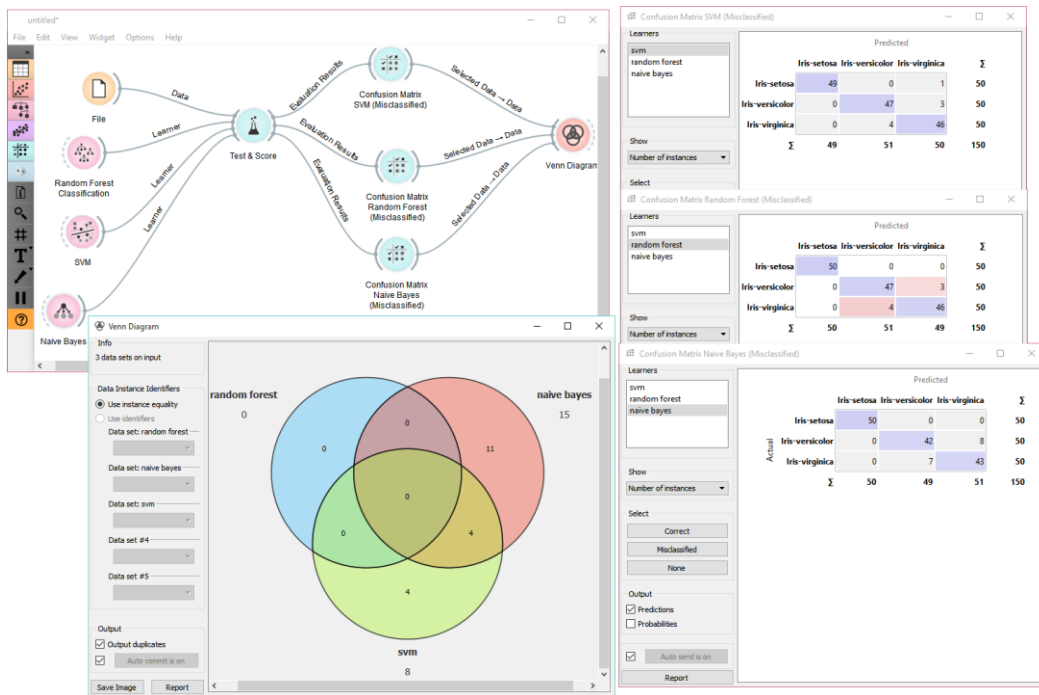
2.9.2 示例

使用 Venn Diagram 最简单的方法是选择数据子集，并在可视化中找到匹配的实例。我们使用 breast-cancer 数据集通过 Select Rows 组件选择两个子集 - 第一个子集是年龄在 40 和 49 之间的乳腺癌患者，第二个是肿瘤大小在 20 和 29 之间的患者。Venn Diagram 有助于我们发现对应于两个标准的实例，可以在两个圆圈的交集中找到。



2.9-2 示例图片

Venn Diagram 组件也可用于探索不同的预测模型。在下面的例子中，我们根据他们的错误分类实例分析了 3 种预测方法，即朴素贝叶斯，SVM 和随机森林。通过在三个 Confusion Matrix 组件中选择错误分类并将其发送到 Venn diagram，我们可以看到所使用的每种方法都可可视化的所有错误分类实例。然后我们打开 Venn Diagram 并选择所有三种方法（在例子 2 中）识别的错误分类的实例。这表示为所有三个圆圈的交集。单击交叉点以查看 Scatterplot 组件中标记的两个实例。尝试选择不同的图表部分来查看散点图可视化的变化。



2.9-3 示例图片

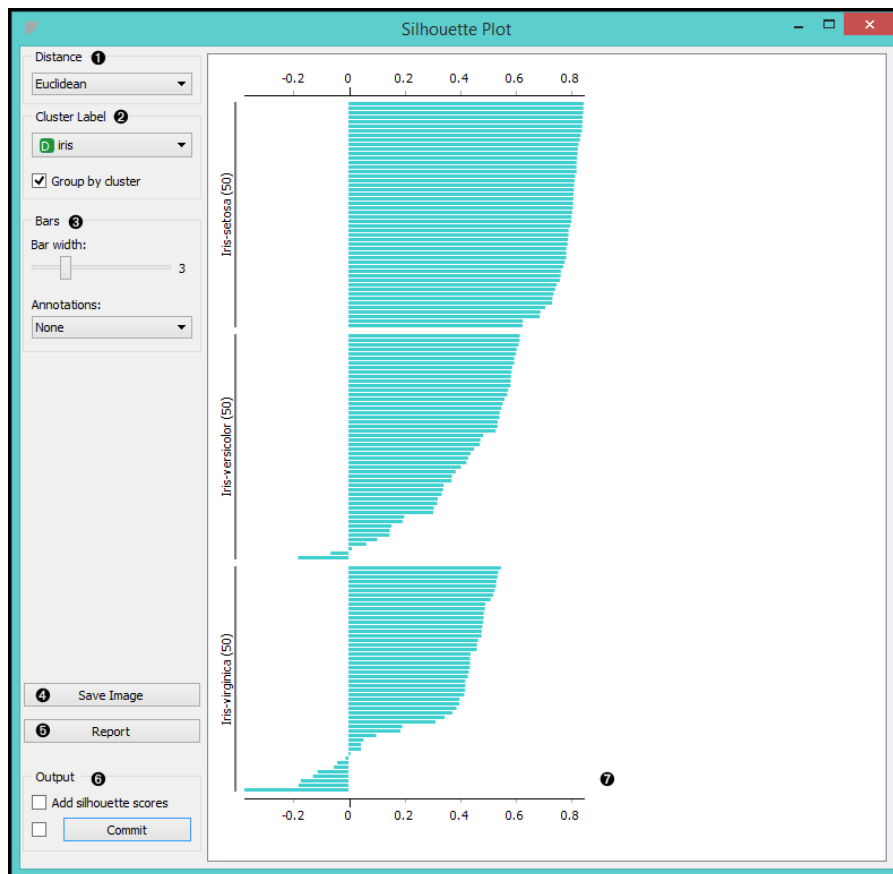
2.10 轮廓图



在数据集群内的一致性的图形表示

2.10.1 描述

Silhouette Plot 组件提供了数据集簇中的一致性的图形表示，并为用户提供了视觉评估集群质量的方法。剪影分数是与其他集群相比较，对象与其自己的集群相似度的度量，并且在创建轮廓图时至关重要。剪影分数接近 1 表示数据实例接近集群的中心，并且接近 0 的剪影分数的实例位于两个集群之间的边界上。



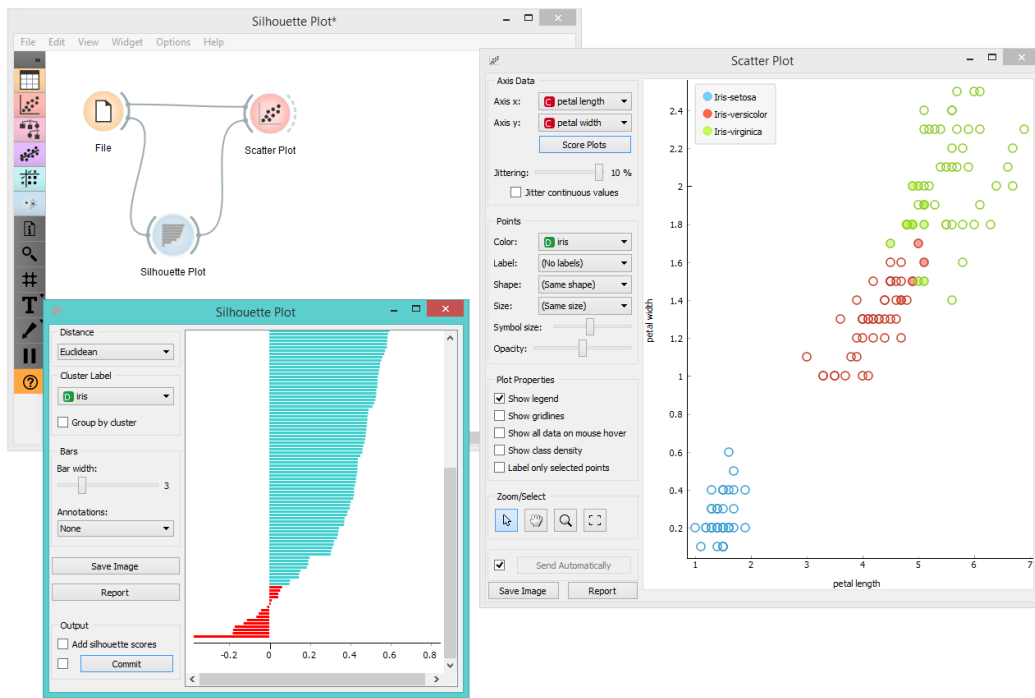
2.10-1 Silhouette Plot 窗口

1. 选择距离度量。您可以选择：
 - 欧几里德（“直线”，两点之间的距离）
 - 曼哈顿（所有属性的绝对差异之和）
2. 选择群集标签。您可以决定是否按集群分组实例。
3. 显示选项：

- 选择栏宽。
 - 注释：注释剪影图。
4. 保存图像以.png 或.svg 格式将创建的剪影图保存到计算机。
 5. 制作报告。
 6. 输出:
 - 添加剪影分数（好的群集有更高的 silhouette 分数）。
 - 通过单击 Commit，更改将分配给组件的输出。或者，勾选左侧的框，更改将自动通知。
 7. 创建的剪影图。

2.10.2 示例

在下面的图片中，我们决定在 iris 数据集上使用 Silhouette Plot。我们选择具有低剪影分数的数据图，并将其作为子集传递给 Scatter Plot 组件。这种可视化只能确认 Silhouette Plot 组件的准确性，因为您可以清楚地看到该子集位于两个群集之间的边界。



2.10-2 示例图片

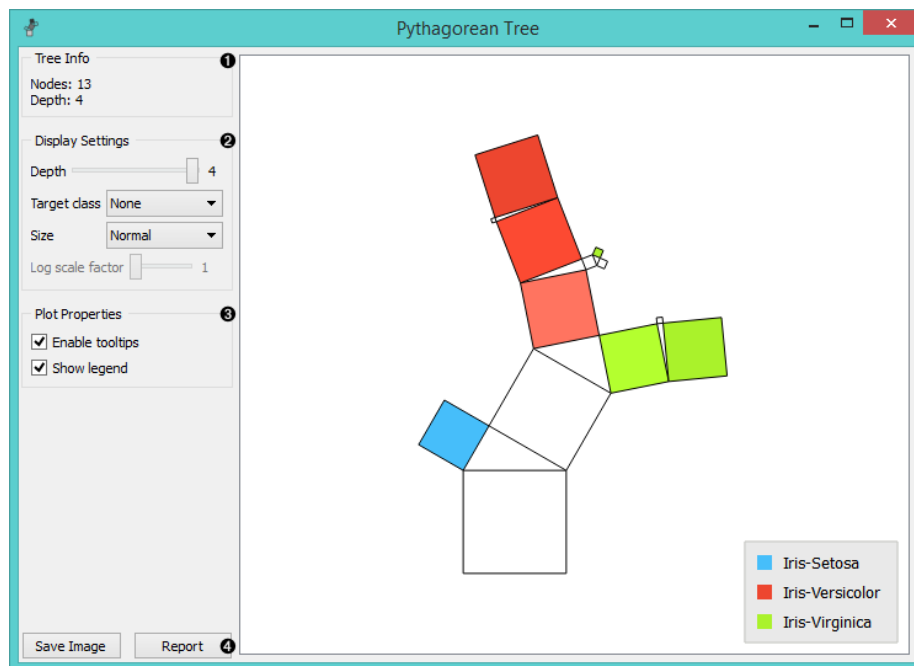
2.11 毕达哥拉斯树



毕达哥拉斯树可视化分类或回归树

2.11.1 描述

Pythagorean Trees 是平面分形，可用于描述一般树状分层结构，如 Fabian Beck 和 co-authors 的文章所述。在我们的例子中，它们用于可视化和探索树模型，如 Tree。



2.11-1 Pythagorean Tree 窗口

1. 输入树模型的信息。
2. 可视化参数:
 - 深度：设置显示树的深度。
 - 目标类（对于分类树）：树的节点的颜色强度将对应于目标类的概率。如果选择“无”，则节点的颜色将表示最可能的类。
 - 节点颜色（用于回归树）：节点颜色可以对应于节点中训练数据实例的类值的平均值或标准偏差。

- 大小：定义一个方法来计算代表节点的平方的大小。Normal 将保持节点大小与节点中训练数据子集的大小相对应。Square root 和 Logarithmic 是节点大小的相应变换。
- 对数比例因子 (Log scale factoris) 仅在选择对数变换时启用。您可以设置 1 到 10 之间的对数因子。

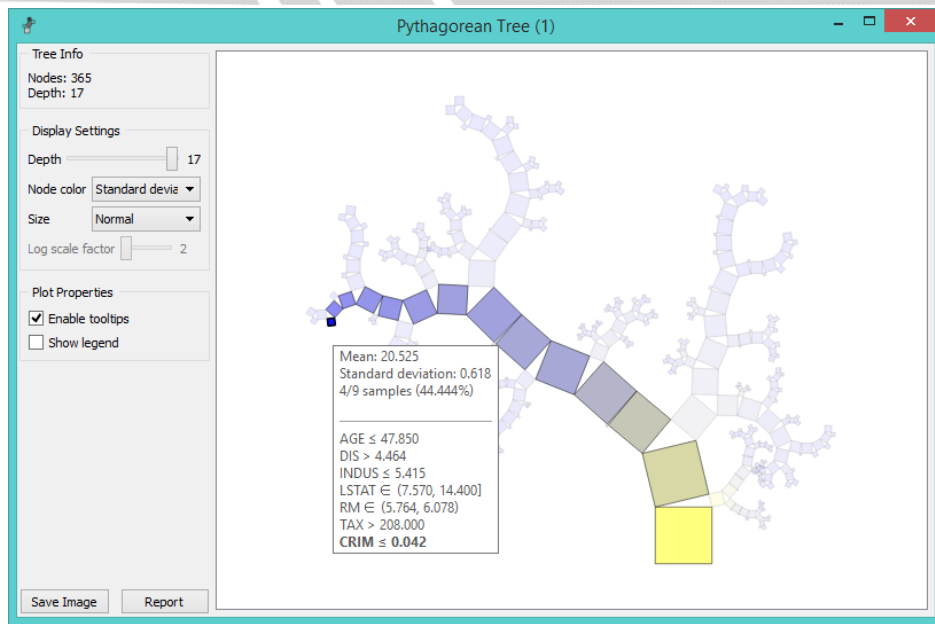
3. 绘图属性：

- 启用组件提示：悬停时显示节点信息。
- 显示图例：显示情节的颜色图例。

4. 报告：

- 保存图像：将可视化文件保存到 SVG 或 PNG 文件。
- 报告：将可视化添加到报表。

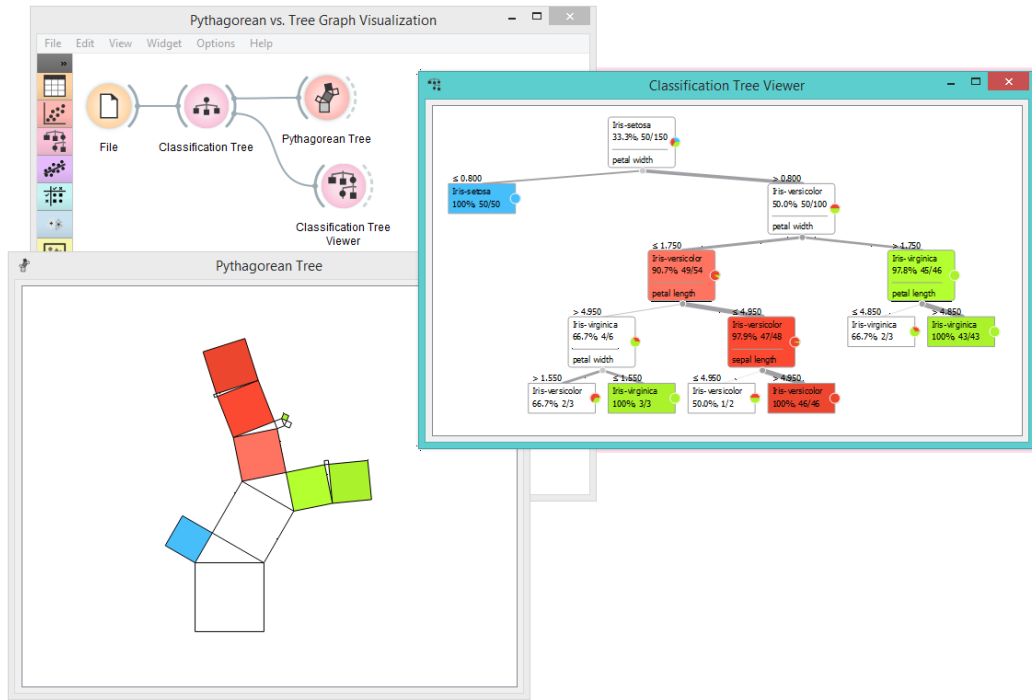
Pythagorean Tree 可以显示分类和回归树。以下是回归树的一个例子。两者之间的唯一区别是回归树不能按类进行着色，而是可以按类平均值或标准偏差进行着色。



2.11-2 示例图片

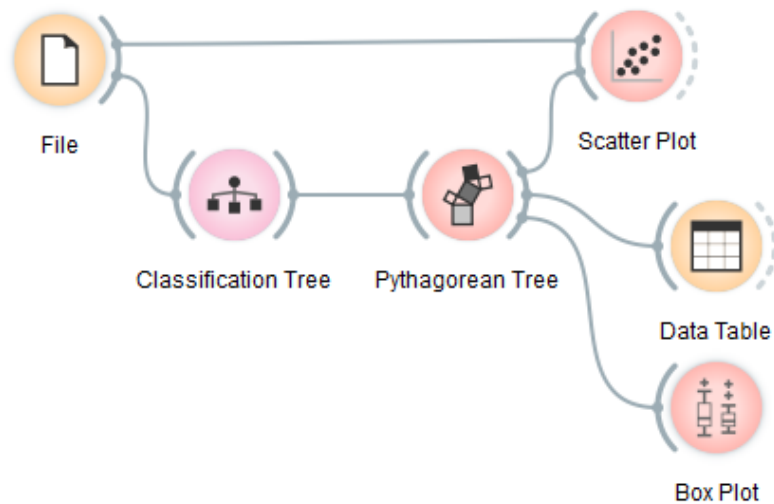
2.11.2 示例

下面的截图的工作流程演示了 Tree Viewer 和 Pythagorean Tree 的区别。它们都可以显示树，但毕达哥拉斯可视化占用较少的空间，更加紧凑，即使是一个小的鸢尾花数据集。对于两个可视化组件，我们通过点击控件和可视化区域之间的分割器来隐藏左侧的控制区域。

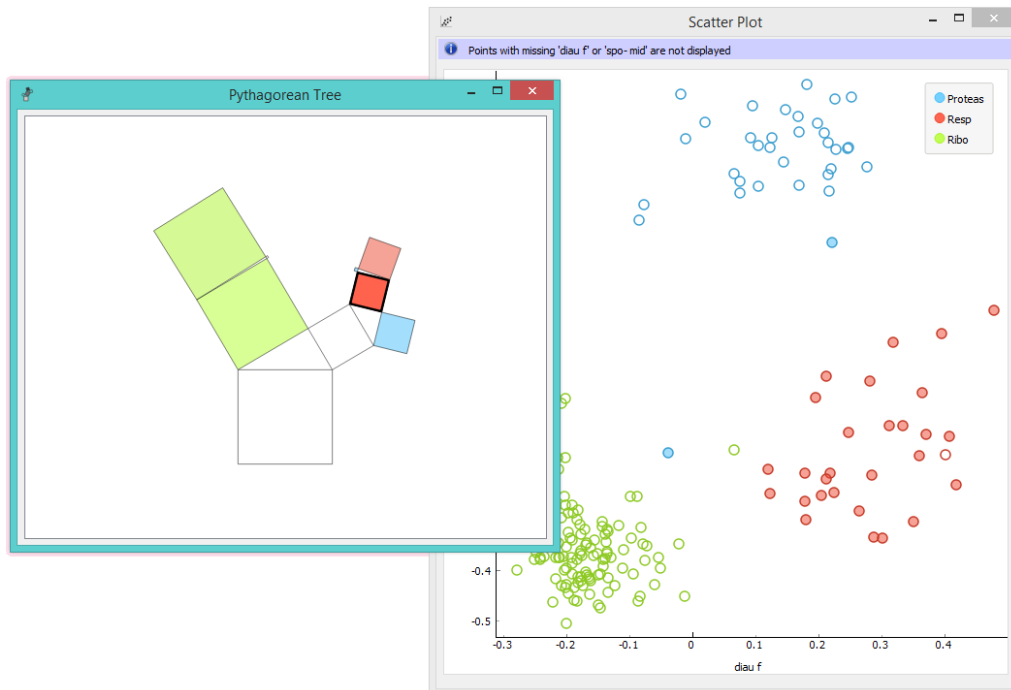


2.11-3 示例图片

Pythagorean Tree 是交互式的：点击任何一个节点（方块）选择与该节点相关联的训练数据实例。以下工作流程将探讨这些功能。



所选择的数据实例在 Scatter Plot 中显示为子集，发送到 Data Table 并在 Box Plot 中进行检查。在本例中我们使用了 brown-selected 数据集。树和散点图如下所示；树中的选定节点具有黑色轮廓。



2.11-4 示例图片

2.12 毕达哥拉斯森林



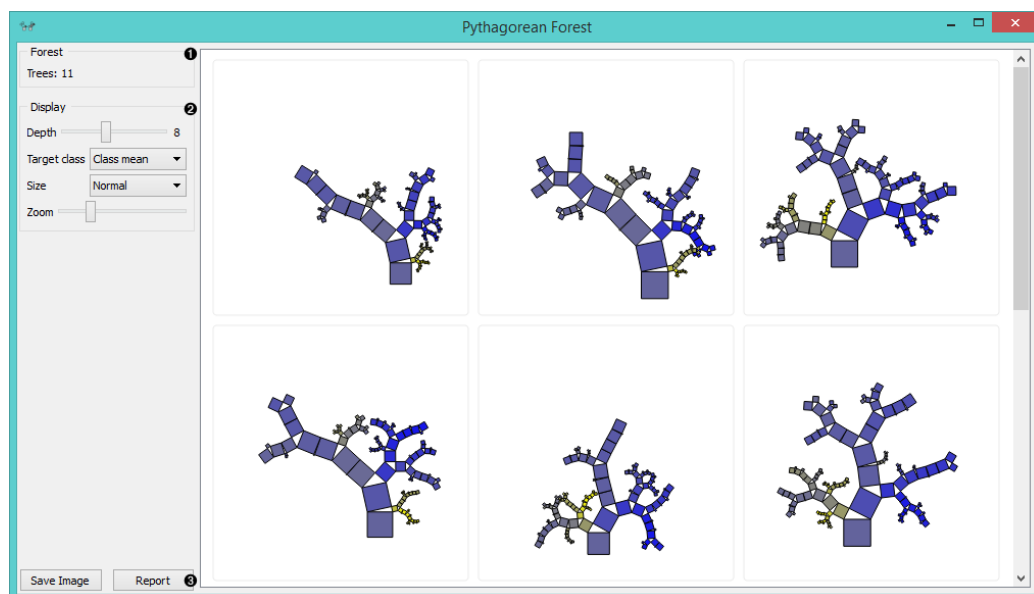
毕达哥拉斯森林可视化随机森林

2.12.1 描述

Pythagorean Forest 显示来自 Random Forest 组件的所有学习决策树模型。它显示为毕达哥拉斯树，每个可视化都属于一个随机构建的树。在可视化中，您可以选择一棵树并在 Pythagorean Tree 组件中显示。最好的树是最短和最强烈色彩的树枝。这意味着很少的属性很好地分割了分支。

组件显示分类和回归结果。分类需要数据集中的离散目标变量，而回归需要连续的目标变量。

不过，他们都应该在输入上输入树。



2.12-1 Pythagorean Forest 窗口

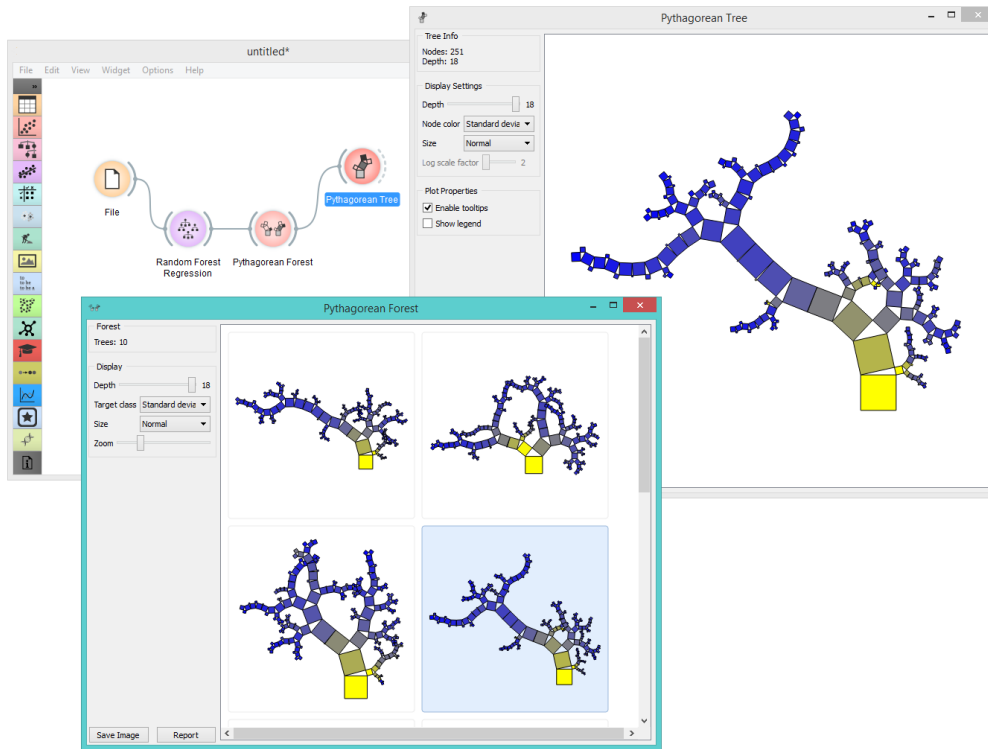
1. 输入随机森林模型的信息。
2. 显示参数：
 - 深度：设置树长成长的深度。

- 目标类：设置用于着色树的目标类。如果选择 None ，树将为白色。如果输入是分类树，则可以通过其相应的类来对节点进行着色。如果输入是回归树，则选项是 Class mean ，它将通过类平均值和标准偏差对树节点进行颜色，然后按照节点的标准偏差值进行颜色校正。
 - 大小：设置节点的大小。Normal 会使节点保持节点中子集的大小。Square root 和 Logarithmic 是节点大小的相应变换。
 - 缩放：允许您查看树可视化的大小。
3. 保存图像：将可视化文件作为.svg 或.png 文件保存到计算机。报告：制作报告。

2.12.2 示例

Pythagorean Forest 擅长于立即可可视化几个树图。在下面的例子中，我们使用了 housing 数据集，并绘制了我们用 Random Forest 绘制的 10 棵树。在 Random Forest 中更改参数时，Pythagorean Forest 中的可视化也将发生变化。

然后，我们在可视化中选择了一棵树，并使用 Pythagorean Tree 组件进一步检查。



2.12-2 示例图片

2.13 CN2 规则查看器

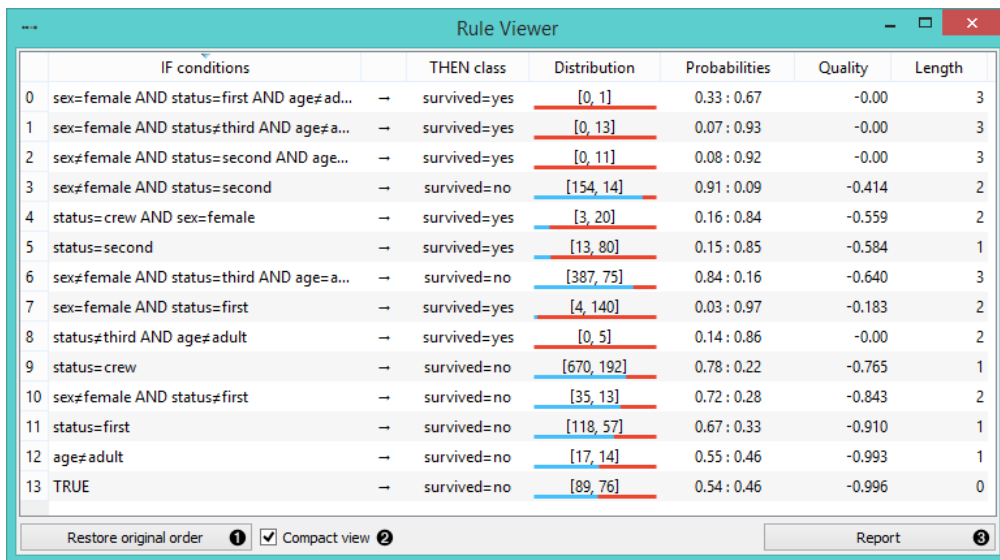


CN2 规则查看器

如果数据连接，一旦主动选择（至少选择一个规则），就会发出过滤的数据。输出是所有选定规则涵盖的数据实例。

2.13.1 描述

显示 CN2 分类规则的组件。如果数据也被连接，在规则选择时，可以分析哪些实例遵守条件。



	IF conditions	THEN class	Distribution	Probabilities	Quality	Length
0	sex=female AND status=first AND age≠ad...	→ survived=yes	[0, 1]	0.33 : 0.67	-0.00	3
1	sex=female AND status≠third AND age≠a...	→ survived=yes	[0, 13]	0.07 : 0.93	-0.00	3
2	sex≠female AND status=second AND age...	→ survived=yes	[0, 11]	0.08 : 0.92	-0.00	3
3	sex≠female AND status=second	→ survived=no	[154, 14]	0.91 : 0.09	-0.414	2
4	status=crew AND sex=female	→ survived=yes	[3, 20]	0.16 : 0.84	-0.559	2
5	status=second	→ survived=yes	[13, 80]	0.15 : 0.85	-0.584	1
6	sex=female AND status=third AND age=a...	→ survived=no	[387, 75]	0.84 : 0.16	-0.640	3
7	sex=female AND status=first	→ survived=yes	[4, 140]	0.03 : 0.97	-0.183	2
8	status≠third AND age≠adult	→ survived=yes	[0, 5]	0.14 : 0.86	-0.00	2
9	status=crew	→ survived=no	[670, 192]	0.78 : 0.22	-0.765	1
10	sex≠female AND status≠first	→ survived=no	[35, 13]	0.72 : 0.28	-0.843	2
11	status=first	→ survived=no	[118, 57]	0.67 : 0.33	-0.910	1
12	age≠adult	→ survived=no	[17, 14]	0.55 : 0.46	-0.993	1
13	TRUE	→ survived=no	[89, 76]	0.54 : 0.46	-0.996	0

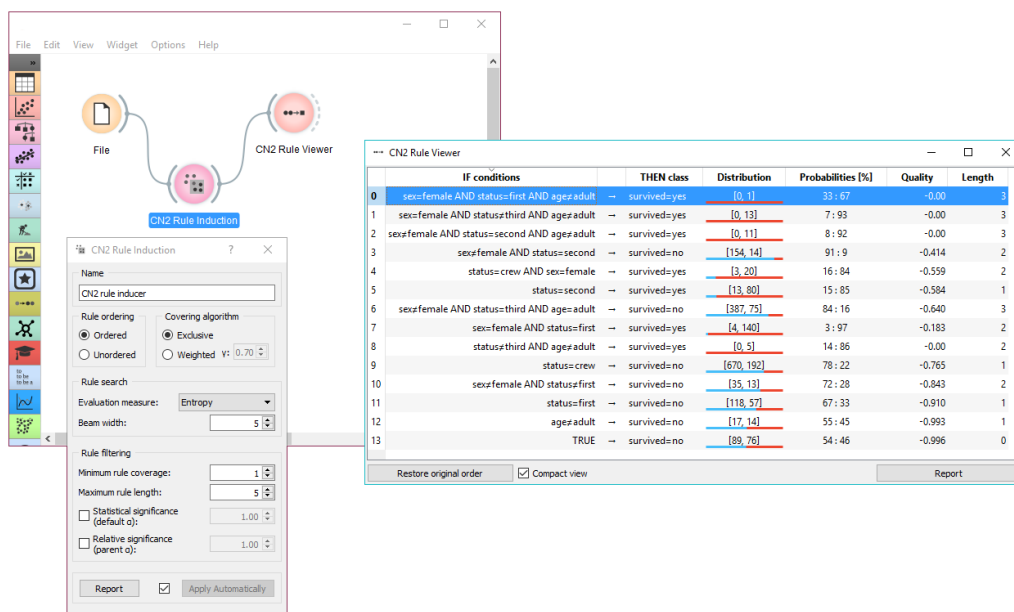
2.13-1 Rule Viewer 窗口

1. 诱导规则的原始顺序可以恢复。
2. 当规则很多且复杂时，视图可以显示包装。为此，实施了紧凑的视野，允许平坦的演示和清洁的规则检查。
3. 点击 Report ，详细说明规则归纳算法及其参数，数据域和诱导规则。

此外，选择后，可以通过按默认的系统快捷方式 (ctrl + C , cmd + C) 将规则复制到剪贴板。

2.13.2 示例

在下面的模式中，提供了该组件最常见的用法。首先，数据的读取和 CN2 规则分类器训练。我们使用的是 Titanic 的数据集的规则建设。然后使用 Rule Viewer 查看规则。为了探索不同的 CN2 算法，并且了解调整参数如何影响学习过程，在设置 CN2 学习算法（演示文稿将被及时更新）的同时，Rule Viewer 应保持开放状态。



2.13-2 示例图片

选择规则输出过滤数据实例。这些可以在 Data Table 中查看。

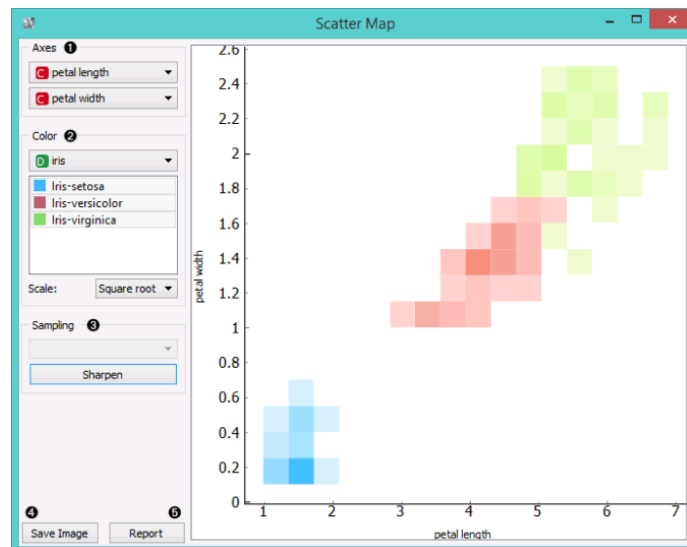
2.14 散点图



绘制一对连续属性的散点图

2.14.1 描述

散点图是用于通过颜色可视化双向矩阵中的频率的图形方法。某个值的出现越高，表示的颜色越暗。通过在 x 和 y 轴上组合两个值，我们可以看到属性组合最强，哪里最弱，从而使用户能够找到强相关性或代表性实例。



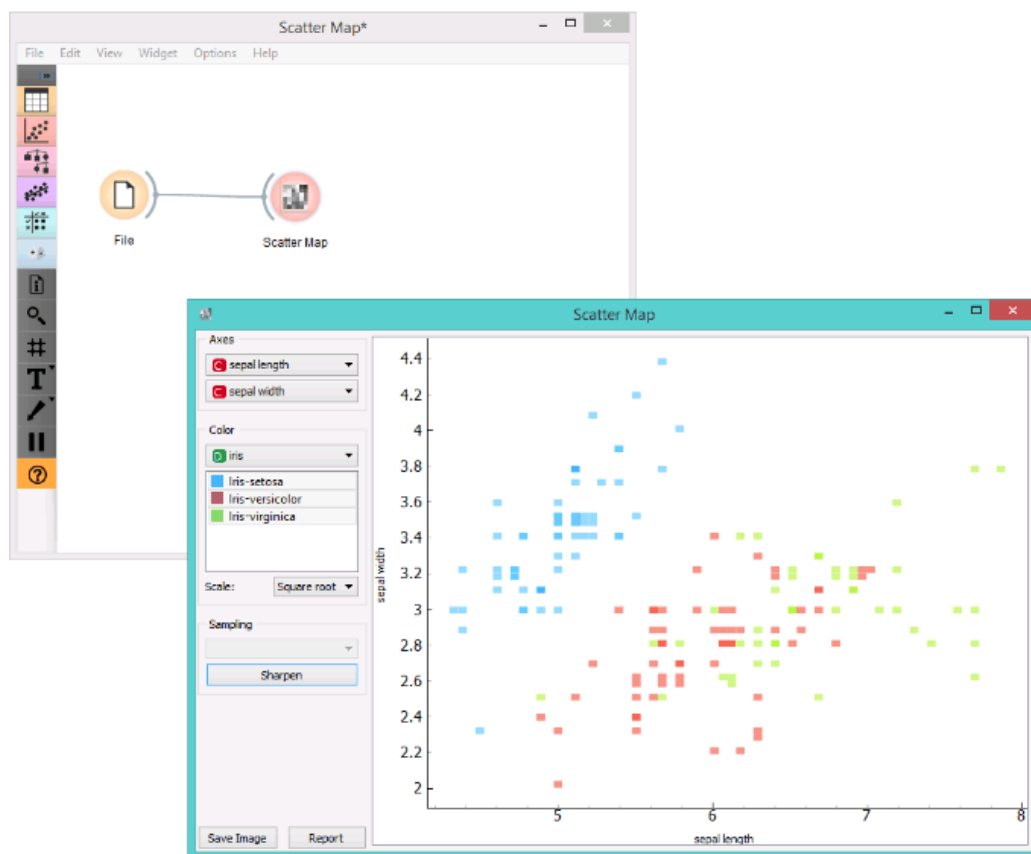
2.14-1 Scatter Map 窗口

1. 选择要绘制的 x 和 y 属性。
2. 通过属性来给图形上色。您还可以通过点击它们在可视化中选择要查看哪些属性实例。在底部，您可以选择色标强度（线性，平方根或对数）。

3. 只有当该组件连接到 SQL Table 组件时，才会启用采样。您可以为大数据设置采样时间，以加快分析速度。锐化适用于所有数据类型，它将调整（锐化）绘图中的正方形。
4. 保存图像将创建的图像以.svg 或.png 格式保存到计算机。
5. 制作报告。

2.14.2 示例

下面，您可以看到 Scatter Map 组件的示例工作流。请注意，该组件仅适用于连续数据，因此您需要先连续地显示要显示的数据属性。下面，Scatter Map 将显示 Iris 数据集的两个属性，即 petal width 和 petal length。在这里，我们可以看到每个类型的宽度和长度值的分布。您可以看到，通过 petal width 和 petal length，品种 Iris setosa 与其他两个品种明显分离，并且这些属性的最典型值对于 petal width 为 0.2 左右，petal length 为 1.4 至 1.7。这表明 petal width 和 petal length 是将 Iris setosa 与其他两个品种区分开来的良好属性。



2.14-2 示例图片

3 分类

 多数学习法	 CN2 规则学习法	 k 最近邻学习法	 分类树学习法
 随机森林	 SVM 学习法	 逻辑回归学习法	 朴素贝叶斯学习法
 Adaboost 学习法	 保存分类器	 加载分类器	

3.1 多数学习法



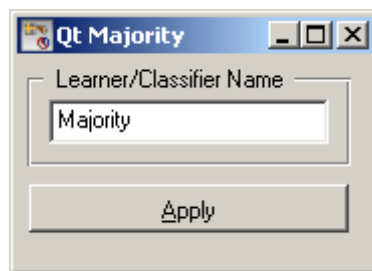
返回多数类而不考虑示例的属性的学习法

3.1.1 描述

这个组件为产生始终预测多数类的分类器的学习法提供图形界面。当问及概率时，它会返回训练集中类的相对频率。当有两个或多个多数类时，该分类器会随机选择预测的类，但对于特殊的示例，它始终返回相同的类。

这个组件通常用于将其他学习算法与默认的分类精度进行比较。

与用于分类的所有其他组件一样，这个组件提供学习法和分类器，前者可以馈送给用于测试学习法的组件，而分类器本身不是非常有用。



唯一的选项是出现在它下面的名称，比方说 Test Learners。默认的名称为“Majority”。

当您更改它时，您需要单击 Apply。

3.1.2 示例

这个组件的典型用途是，将它连接到 Test Learners 以将其他学习法算法（比如这个方案中的 kNN）的分数与默认分数进行比较。

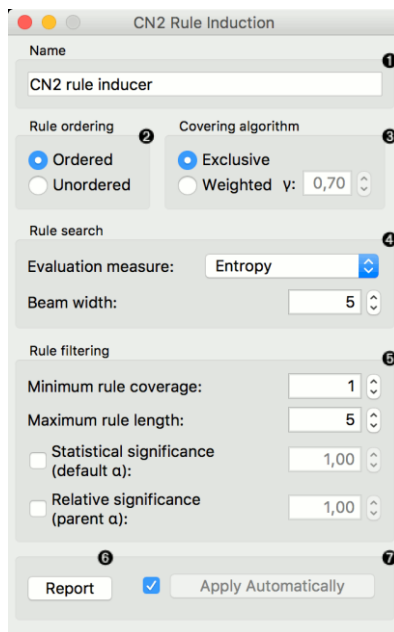
3.2 CN2 规则学习法



使用 CN2 算法从数据引导规则

3.2.1 描述

CN2 算法是一种分类技术，它可以有效地归纳出简单的、可理解的形式规则，即使在可能存在噪声的领域中也可以。CN2 规则归纳法只适用于分类。



3.2-1 CN2 Rule Induction 窗口

1. 命名。默认名称为 CN2 Rule Induction。

2. 规则排序：

有序：诱导有序规则（决策列表）。

无序：引发无序规则（规则集）。

3. 覆盖算法：

专属：覆盖学习实例后，将其从进一步考虑中删除。

加权：覆盖学习实例后，减轻其权重，反过来又会降低对算法进一步迭代的影响。

4. 规则搜索：

-评估措施：选择启发式来评估发现的假设：

熵（内容不可预测性的度量）

湾拉普拉斯精度

加权相对精度

-光束宽度：记住发现的最佳规则，并监视固定数量的备选方案（光束）。

5. 规则过滤：

最低规则覆盖率：找到的规则必须至少涵盖所覆盖范例的最低要求数量。无序规则

必须涵盖许多目标类的例子。

最大规则长度：找到的规则最多可以组合最大允许的选择器数量（条件）。

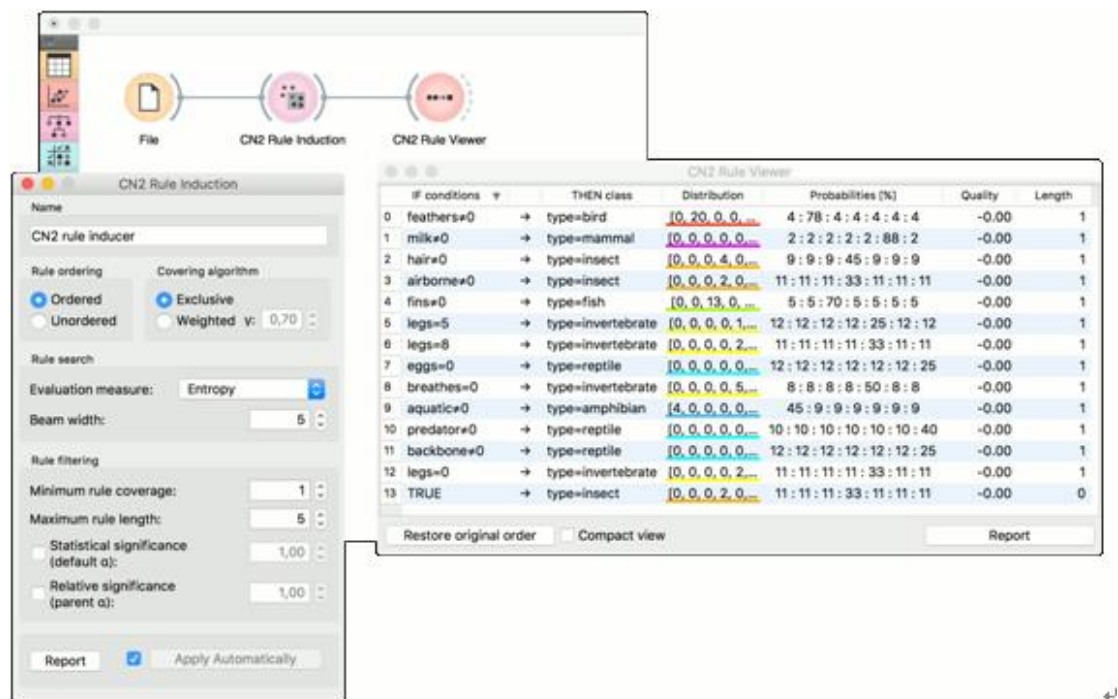
默认 alpha

父类 alpha

6. 勾选“自动应用”以自动传送其他组件的更改，或者，在配置后按“应用”。

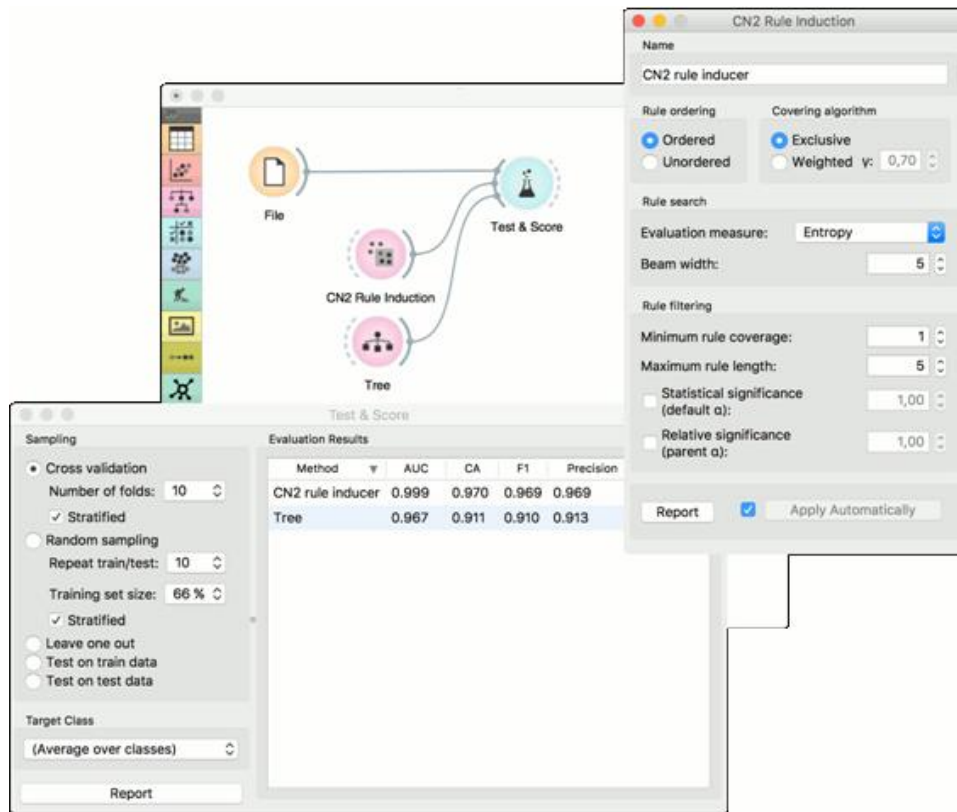
3.2.2 示例

对于下面的例子，我们将数据集传递给 CN2 Rule Induction。我们可以使用 CN2 规则查看器组件来查看模型。



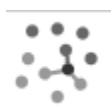
3.2-2 示例图片

第二个工作流测将测试和分数组件规则与树、CN2 规则法组件连接。



3.2-3 示例图片

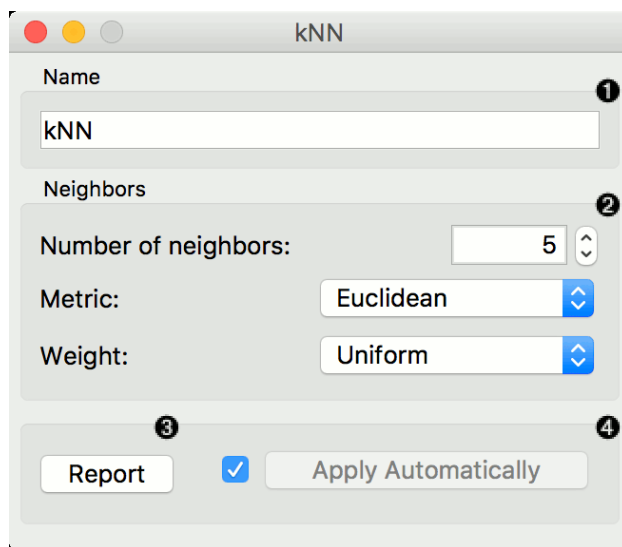
3.3 K 最近邻学习法



k 最近邻(kNN)学习法

3.3.1 描述

kNN 组件使用 kNN 算法，在特征空间中搜索 k 个最接近的训练样本，并使用它们的平均值作为预测。



3.3-1 kNN 窗口

1. 命名。默认名称为“kNN”。
2. 设置最近邻居的数量
 - a) 距离参数（度量）和权重作为模型标准。
 - b) 欧几里德（“直线”，两点之间的距离）
 - c) 曼哈顿（所有属性的绝对差异之和）
 - d) 最大（属性之间绝对差异最大）
 - e) 马氏距离（点与分布之间的距离）。

-重量

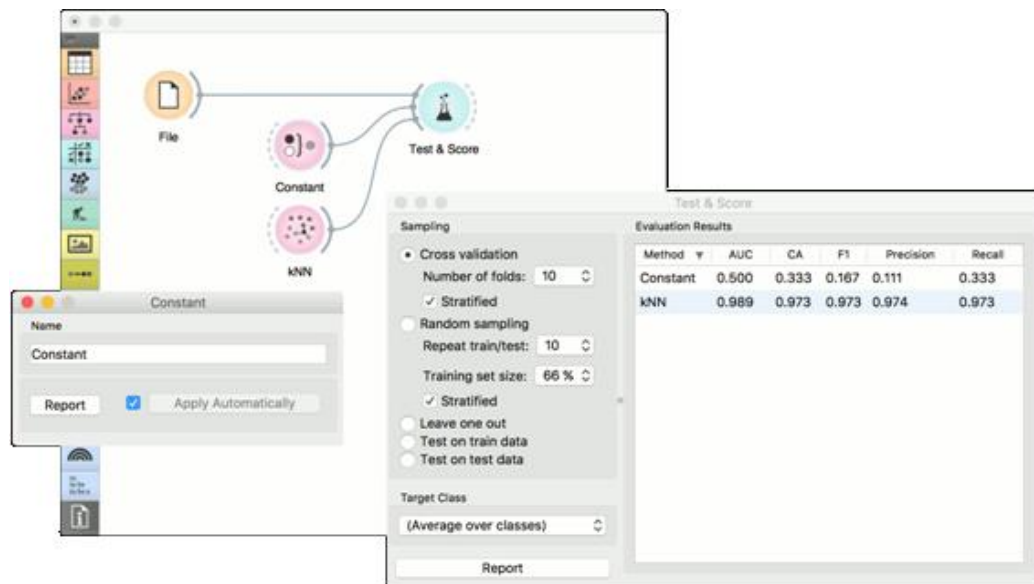
- f) o 一致：每个邻域中的所有点 均加权相等。
- g) o 距离：查询点的邻近的邻居比比其他邻居影响更大。

3. 生成报告。

4. 当您更改一个或多个设置时，您需要单击 Apply（应用），也可以通过单击应用按钮左侧的框自动应用更改。

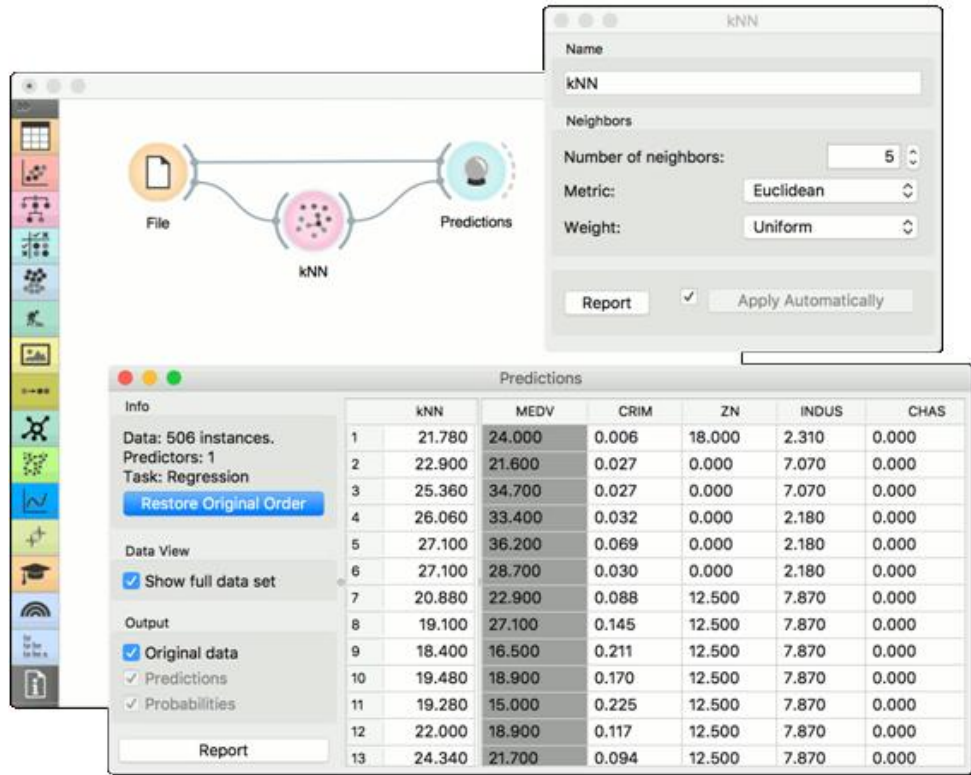
3.3.2 示例

第一个例子是 iris 数据集的分类任务。我们将 k-最近邻居的结果与常量组件进行比较。



3.3-2 示例图片

第二个例子是回归任务。我们将 kNN 预测模型输入到预测组件中并观察预测值。



3.3-3 示例图片

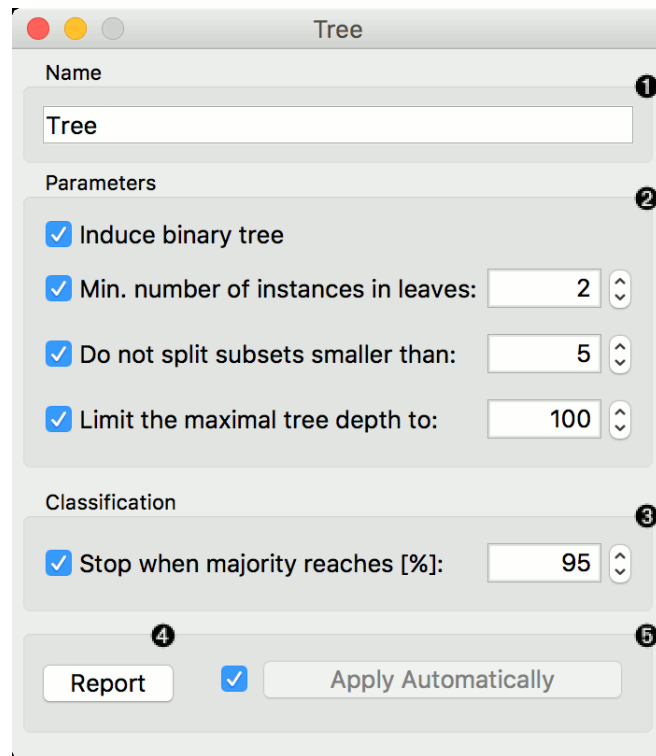
3.4 分类树学习法



分类树学习法

3.4.1 描述

树是一种简单的算法，它通过类纯度将数据分解成节点。它是随机森林的前身。Mining 中的树可以处理离散和连续的数据集。



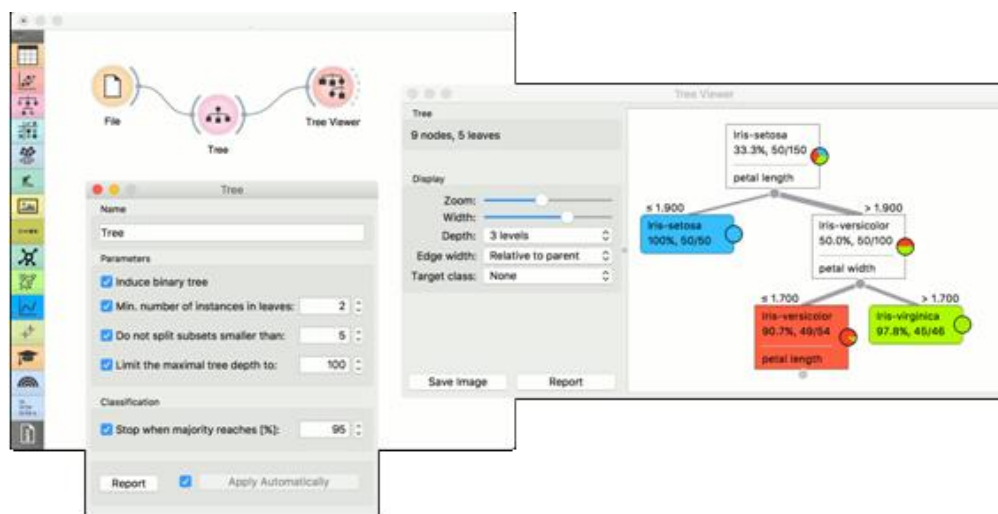
3.4-1 Tree 窗口

1. 命名。默认名称为“树”。
2. 树参数：
 - 诱导二叉树：构建一个二叉树（分为两个子节点）
 - 叶中的最小实例数：如果选中，算法将永远不会构建一个小于指定数量的训练样本的分割进入任何分支。
 - 不要拆分小于（ ）以下的子集：禁止用少于给定数量的实例分割节点的算法。

- 限制最大树深度：将分类树的深度限制为指定数量的节点级别。
3. 当多数达到[%]时停止：达到指定的多数阈值后，停止分割节点
 4. 制作报告。
 5. 更改设置后，您需要单击“应用”，或者，勾选左侧的框，更改将自动通知。

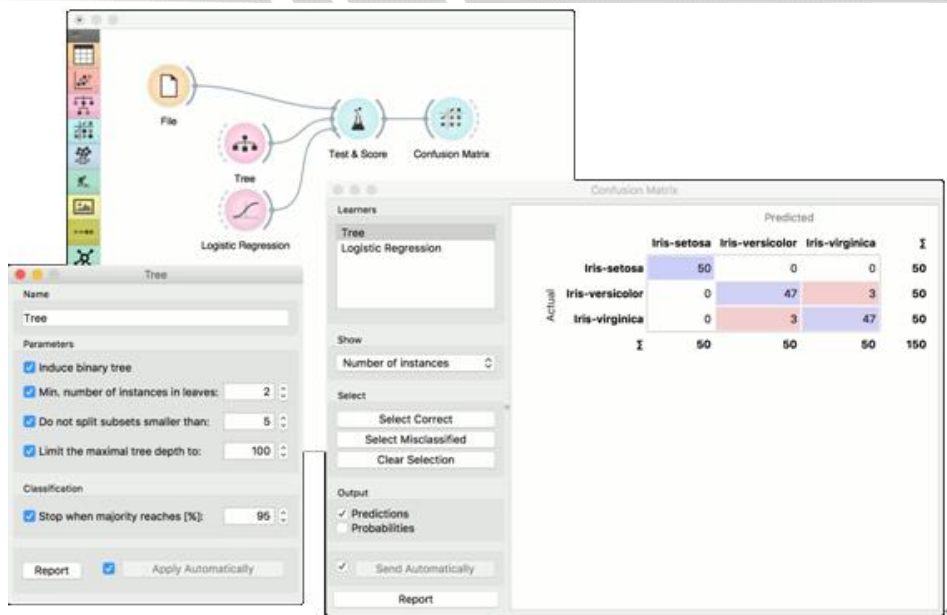
3.4.2 示例

这个组件有两个典型的用途。首先，您可能需要引用一个模型并检查它在 Tree Viewer 中的情况。



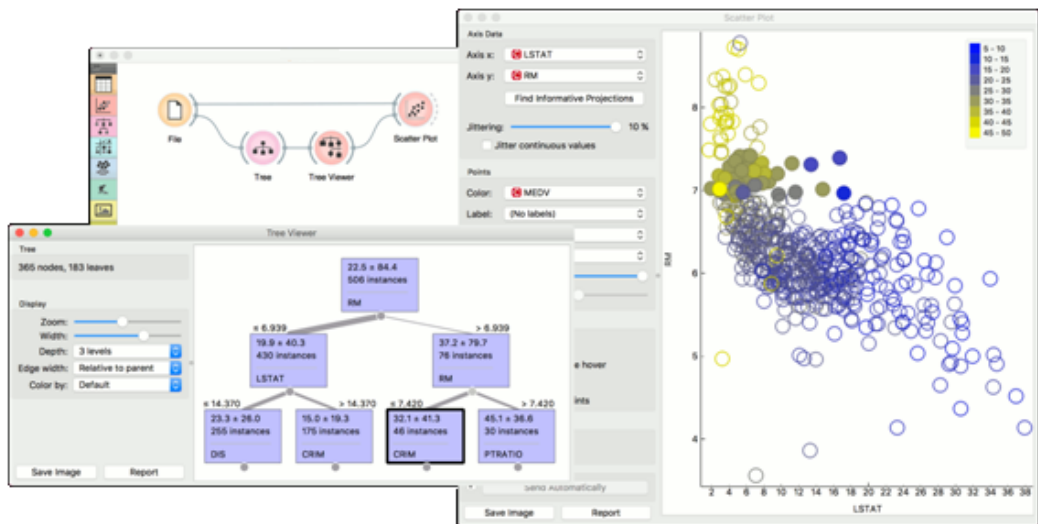
3.4-2 示例图片

第二个模式训练一个模型，并评估其与逻辑回归性能。



3.4-3 示例图片

我们在这两个例子中都使用了 iris 数据集。然而，Tree 也用于回归任务。使用 housing 数据集并将其传递给树。来自 Tree Viewer 的选择的树节点显示在散点图中，我们可以看
到所选示例具有相同的特征。



3.4-4 示例图片

3.5 随机森林

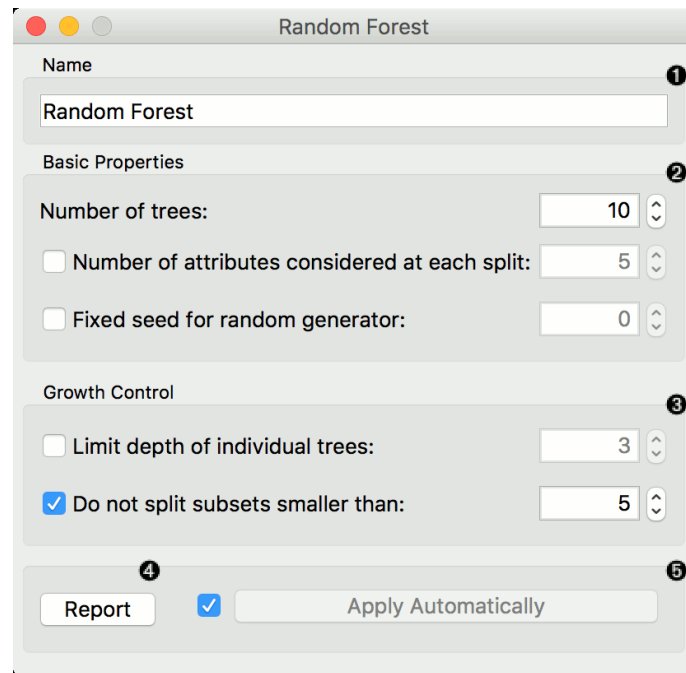


随机森林学习法

3.5.1 描述

随机森林是一种用于分类、回归等任务的集成学习方法。它首先是由 Tin Kam Ho 提出并由 Leo Breiman 和 Adele Cutler 进一步发展。

随机森林建立一组决策树。每个树从引导样本的训练数据开发的。在开发单个树时，绘制任意属性子集（即“随机”），从中选择分割的最佳属性。最后的模型是基于在森林中单独开发的树木的多数结果。

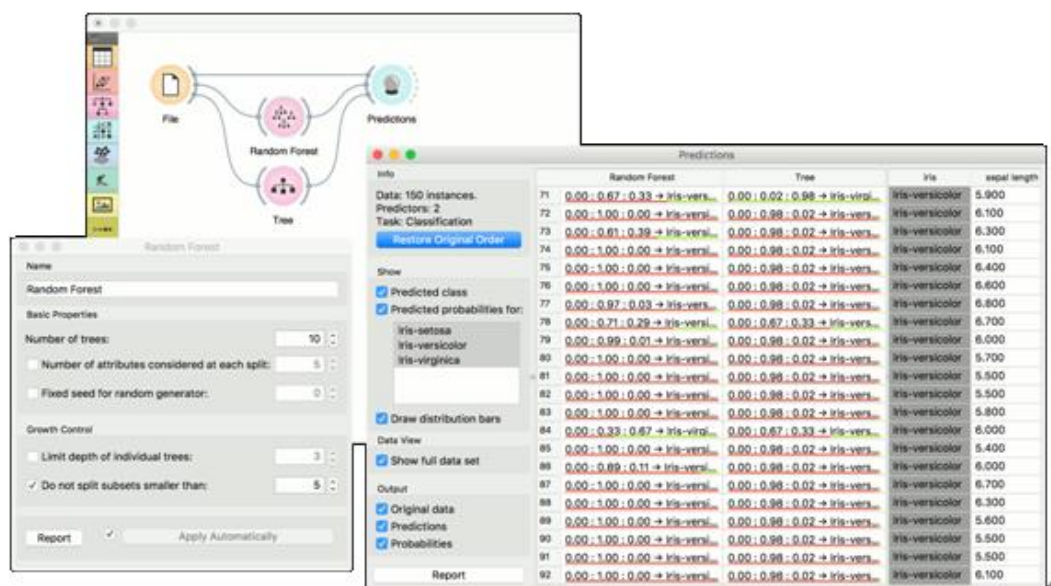


3.5-1 Random Forest 窗口

1. 命名。默认名称为“随机森林”。
2. 指定森林中将包括多少个决策树（森林中的树数），以及任意绘制多少属性以供每个节点考虑。如果未指定后者，则该数字等于数据中属性数的平方根。
3. 最初 Brieman 的建议是对决策树不做任何预先设置，但是由于预处理工作很好，速度更快，用户可以设置随机森林的深度（限制单棵树的深度）。另一个预处理是选择可以拆分的最小子集（不要拆分子集小于）。
4. 生成报告。
5. 单击应用将更改通知给其他组件。或者，勾选“应用”按钮左侧的框，更改将自动进行通信。

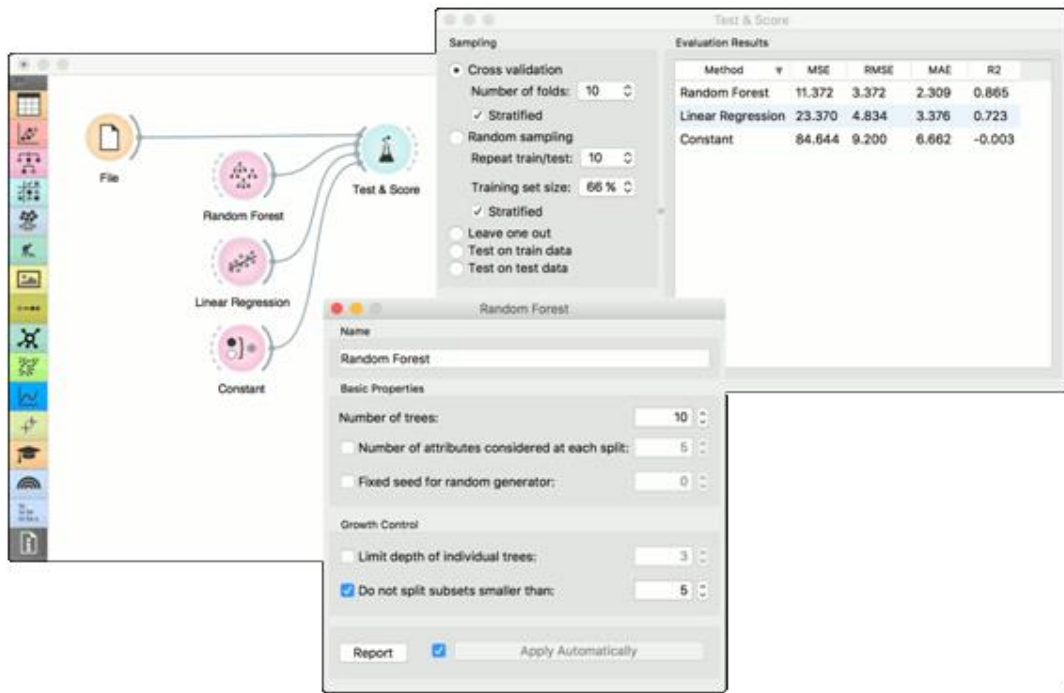
3.5.2 示例

对于分类任务，我们使用 iris 数据集。将其连接到预测组件。然后将文件连接到随机林和树，并将它们进一步连接到预测。最后，观察两个模型的预测。



3.5-2 示例图片

对于回归任务，我们将使用 housing 数据。在这里，我们将在测试和分数组件中比较不同的模型，即随机森林，线性回归和恒定组件。



3.5-3 示例图片

3.6 SVM 学习法

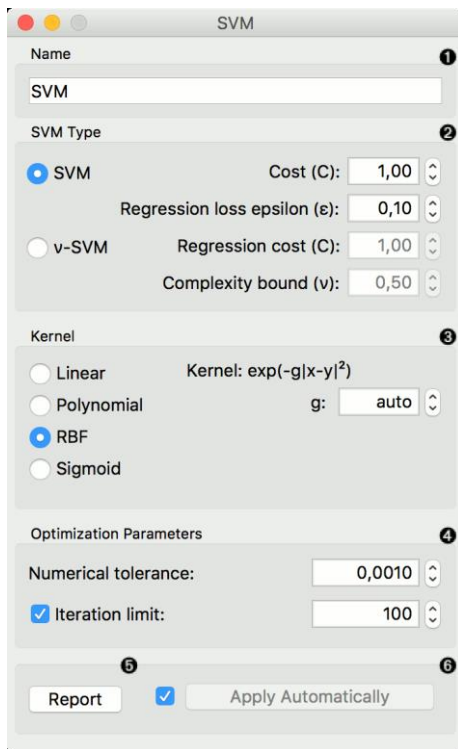


支持向量机将输入映射到更高维特征空间

3.6.1 描述

支持向量积 (SVM) 是一种流行的分类技术，它将在属性空间中构造一个分离的超平面，以最大化不同类的实例之间的边界。该技术往往会产生最高的预测性能结果。Mining 在 LIBSVM 程序包中嵌入了 SVM 的一个流行的实现，这个组件为它的功能提供了一个图形用户界面。

对于回归任务，SVM 使用 ϵ 不敏感的损失在高维特征空间中执行线性回归。其估计精度取决于 C ， ϵ 和核参数的良好设置。组件基于 SVM 回归输出类预测。该组件适用于分类和回归任务。



3.6-1 SVM 窗口

1. 命名。默认名称为“SVM”。
2. 具有测试错误设置的 SVM 类型。SVM 和 ν -SVM 基于误差函数的差异最小化。在右侧，您可以设置测试错误范围：

\emptyset SVM :

成本：损失的罚款项，适用于分类和回归任务。

ϵ : epsilon-SVR 模型的参数适用于回归任务。定义与真实值的距离，其中没有惩罚值与预测值相关联。

$\emptyset\nu$ -SVM :

成本：损失的罚款项，仅适用于回归任务

ν : ν -SVR 模型的参数，适用于分类和回归任务。训练误差分数的上限和支持向量分数的下限。

3. 核是将属性空间转换为一个新的特征空间以适应最大边值超平面的一种函数，因此允许算法使用以下方式创建模型：

线性

多项式

RBF 和

Sigmoid

在数值公差中设置允许偏离预期值。勾选迭代限制旁边的框，以设置允许的最大迭代次数。

4. 制作报告。

5. 单击应用以提交更改。如果勾选“应用”按钮左侧的框，则会自动进行更改。

3.6.2 示例

示例显示了如何使用 SVM 与散点图组合。以下工作流程对 iris 数据集进行 SVM 模型的训练，并输出支持向量，这些向量是在学习阶段用作支持向量的那些数据实例。我们可以在散点图可视化中观察这些数据实例的哪些。请注意，要使工作流程正常工作，您必须设置组件之间的链接，如下面的示例所示。



3.6-2 示例图片

3.7 逻辑回归学习法

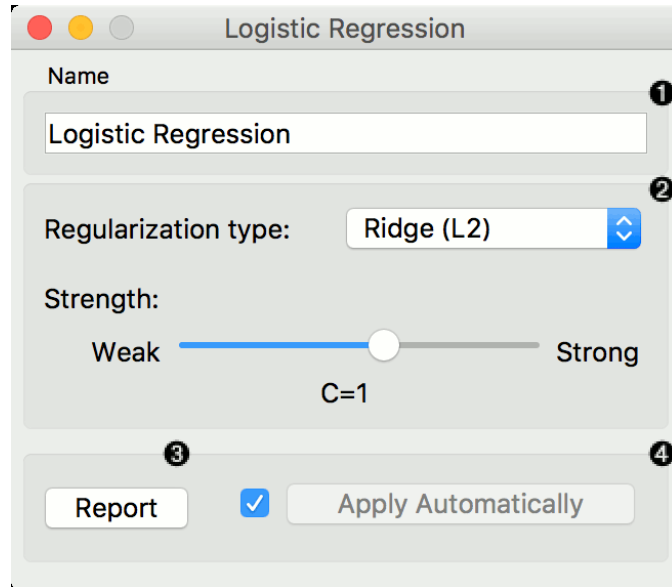


逻辑回归学习法

3.7.1 描述

Logistic Regression 从数据中学习逻辑回归模型。

它只适用于分类任务。



3.7-1 Logistic Regression 窗口

1. 可以为学习法指定名称（将在下面显示），比方说 Test Learners。默认的名称为“Logistic regression”。
2. 正规化类型（L1 或 L2）。设置成本强度（默认为 $C = 1$ ）。
3. 制作一个报告。
4. 如果选中 Send automatically，更改会自动上传，否则，点击 Send。

3.7.2 示例

该组件和其他诱导分类器的组件一样。这是一个示例，通过对 hayes-roth 数据集的逻辑回归来展示预测结果。我们首先在 File 组件中加载 hayes-roth_learn，并将数据传递给 Logistic Regression。然后我们将经过训练的模型传递给 Predictions。

现在我们要预测一个新数据集的类值。我们在第二个 File 组件中加载 hayes-roth_test，并将其连接到 Predictions。我们现在可以直接在 Predictions 中观察用 Logistic Regression 预测的类值。

The screenshot shows a workflow in a data science environment. It includes a 'Logistic Regression' component with the following settings:

- Name: Logistic Regression
- Regularization type: Ridge (L2)
- Strength: Weak (C=1)
- Buttons: Report, Apply Automatically

The 'Predictions' component displays the following data table:

	Logistic Regression	y	hobby	age	education	marital
1	0.77 : 0.17 : 0.06 → 1	1	1	1	1	2
2	0.77 : 0.17 : 0.06 → 1	1	1	1	2	1
3	0.77 : 0.17 : 0.06 → 1	1	1	2	1	1
4	0.88 : 0.04 : 0.08 → 1	1	1	1	1	3
5	0.88 : 0.04 : 0.08 → 1	1	1	1	3	1
6	0.88 : 0.04 : 0.08 → 1	1	1	3	1	1
7	0.73 : 0.15 : 0.12 → 1	1	1	1	3	3
8	0.73 : 0.15 : 0.12 → 1	1	1	3	1	3
9	0.73 : 0.15 : 0.12 → 1	1	1	3	3	1
10	0.17 : 0.77 : 0.06 → 2	2	1	2	2	1
11	0.17 : 0.77 : 0.06 → 2	2	1	2	1	2
12	0.17 : 0.77 : 0.06 → 2	2	1	1	2	2
13	0.04 : 0.88 : 0.08 → 2	2	1	2	2	3
14	0.04 : 0.88 : 0.08 → 2	2	1	2	3	2
15	0.04 : 0.88 : 0.08 → 2	2	1	3	2	2
16	0.15 : 0.73 : 0.12 → 2	2	1	2	3	3
17	0.15 : 0.73 : 0.12 → 2	2	1	3	2	3
18	0.15 : 0.73 : 0.12 → 2	2	1	3	3	2
19	0.46 : 0.46 : 0.09 → 1	1	1	1	3	2
20	0.46 : 0.46 : 0.09 → 2	2	1	3	2	1
21	0.46 : 0.46 : 0.09 → 1	1	1	2	1	3
22	0.46 : 0.46 : 0.09 → 2	2	1	2	3	1

3.7-2 示例图片

3.8 朴素贝叶斯学习法

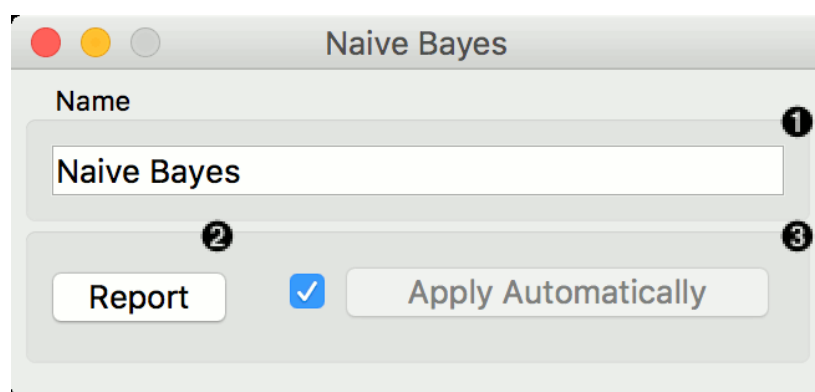


朴素贝叶斯学习法

3.8.1 描述

Naive Bayes 从数据中学习一个朴素贝叶斯模型。

它只适用于分类任务。

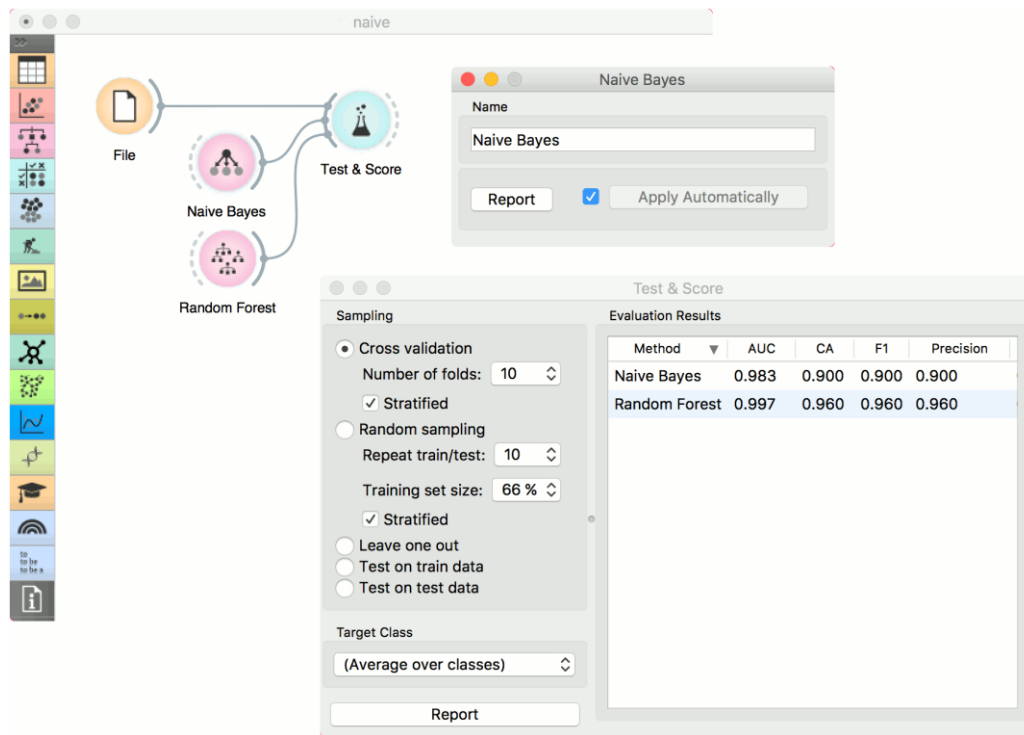


3.8-1 Naive Bayes 窗口

1. 名称（默认是 Naive Bayes）。
2. 制作一个报告。
3. 如果选中 Send automatically，更改会自动上传，否则，点击 Send。

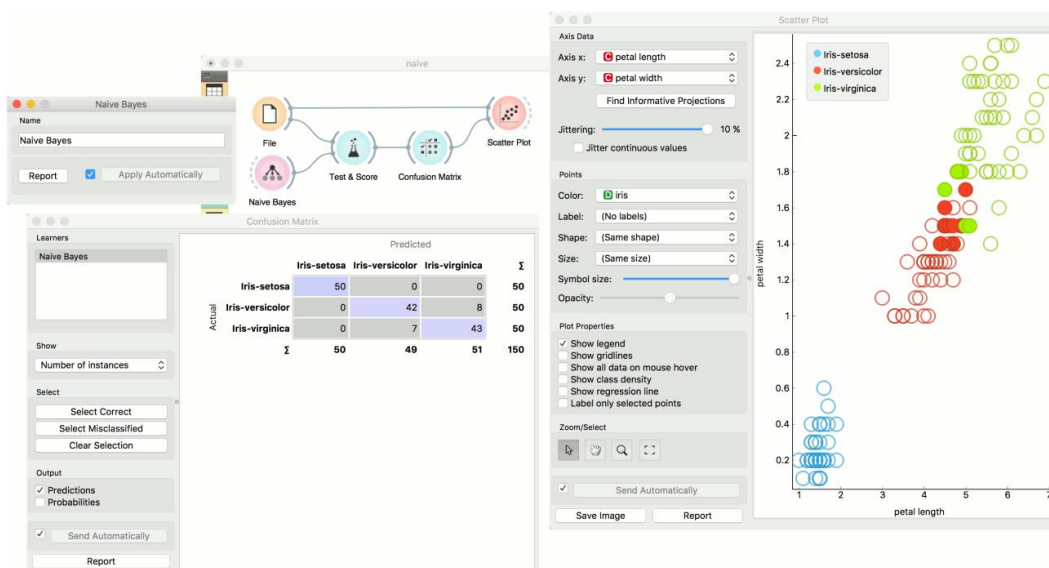
3.8.2 示例

这里，我们介绍这个组件的两个用途。首先，我们将朴素贝叶斯的结果与另一个模型“随机森林”进行比较。把 iris 数据集从 File 连接到 Test&Score。我们还连接朴素贝叶斯和随机森林到 Test&Score，并观察他们的预测分数。



3.8-2 示例图片

第二种模式显示了 Naive Bayes 做出的预测的质量。我们将 Test & Score 组件提供给 Naive Bayes 学习者，然后将数据发送到 Confusion Matrix。然后我们在 Confusion Matrix 中选择错误分类的实例，并将它们显示给 Scatterplot。散点图中的大胆点是朴素贝叶斯的错误分类实例。



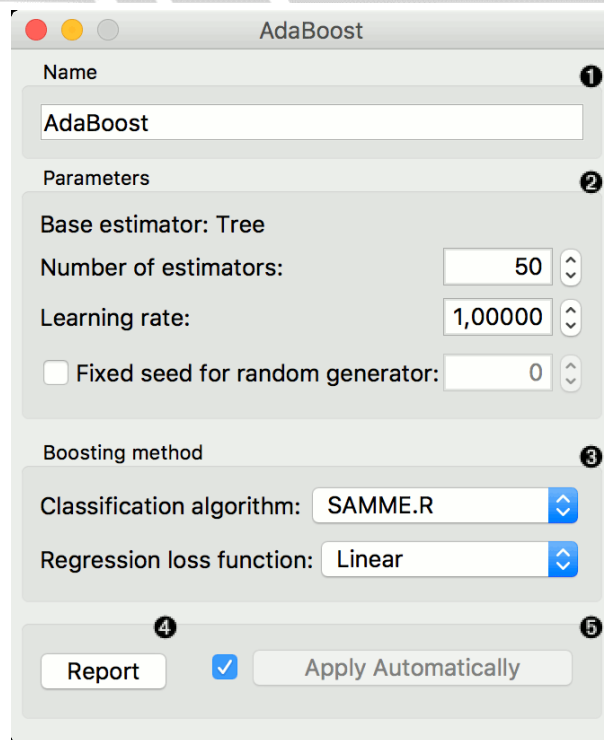
3.9 Adaboost 学习法



3.9.1 描述

AdaBoost (“自适应增强”) 小部件的简写是由 Yoav Freund 和 Robert Schapire 制定的机器学习算法。它可以与其他学习算法一起使用来提高其性能。它通过调整弱势的学习者来做到这一点。

AdaBoost 适用于分类和回归。



3.9-1 AdaBoost 窗口

1. 命名。默认名称为 “AdaBoost” 。
2. 设置参数。您可以设置：
 - a) 估计量数
 - b) 学习率：它确定新获得的信息将在多大程度上覆盖旧信息（0 =代理不会学习任何东西，1 =代理仅考虑最新信息）
3. 用于随机发生器的固定种子：设置固定的“种子”以使得能够再现结果。
升压方式。
 - a) 分类算法（如果输入分类）：SAMME（使用分类结果更新基本估计器的权重）或 SAMME.R（使用概率估计更新基本估计器的权重）。

回归损失函数（如果输入回归）：Linear（ ）， Square（ ）， Exponential（ ）。

4. 生成报告。
5. 更改设置后，单击应用。要自动通知更改自动勾选自动应用。

3.9.2 示例

对于分类，我们加载了 Iris 数据集。 我们使用 AdaBoost，分类树和逻辑回归组件，并评估模型在“测试和分数”中的表现。

The screenshot shows the Orange3 software interface. A workflow is visible with the following components: File, AdaBoost, Tree, and Logistic Regression, all connected to a Test & Score widget. The Test & Score widget is open, showing the following settings and results:

AdaBoost Settings:

- Name: AdaBoost
- Parameters:
 - Base estimator: Tree
 - Number of estimators: 50
 - Learning rate: 1,00000
 - Fixed seed for random generator: 26
- Boosting method:
 - Classification algorithm: SAMME.R
 - Regression loss function: Linear

Test & Score Settings:

- Sampling:
 - Cross validation (selected):
 - Number of folds: 10
 - Stratified (checked)
 - Random sampling
 - Repeat train/test: 10
 - Training set size: 66%
 - Stratified (checked)
 - Leave one out
 - Test on train data
 - Test on test data
- Target Class: (Average over classes)

Evaluation Results:

Method	AUC	CA	F1	Precision	Recall
AdaBoost	0.965	0.953	0.953	0.953	0.953
Tree	0.975	0.960	0.960	0.960	0.960
Logistic Regression	0.990	0.960	0.960	0.962	0.960

3.9-2 示例图片

对于回归，我们加载了 housing 数据集，将数据实例发送到两个不同的模型（AdaBoost 和 Tree），并将它们输出到“预测”组件中。

The screenshot shows a workflow in Orange3. The workflow consists of a File component, an AdaBoost component, a Tree component, and a Predictions component. The AdaBoost component is configured with 50 estimators, a learning rate of 1,00000, and the SAMME.R classification algorithm. The Tree component is also configured. The Predictions component shows a table of results for 13 instances, comparing the predictions of the AdaBoost and Tree models against the actual MEDV values and other features like CRIM, ZN, and INDUS.

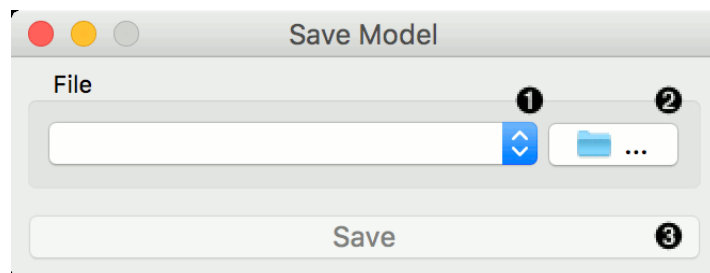
Info	AdaBoost	Tree	MEDV	CRIM	ZN	INDUS
1	24.000	26.350	24.000	0.006	18.000	2.310
2	21.600	21.867	21.600	0.027	0.000	7.070
3	34.700	34.800	34.700	0.027	0.000	7.070
4	33.400	33.200	33.400	0.032	0.000	2.180
5	36.100	37.150	36.200	0.069	0.000	2.180
6	28.700	28.900	28.700	0.030	0.000	2.180
7	22.600	22.300	22.900	0.088	12.500	7.870
8	27.100	22.100	27.100	0.145	12.500	7.870
9	16.500	15.475	16.500	0.211	12.500	7.870
10	18.900	18.350	18.900	0.170	12.500	7.870
11	15.000	15.475	15.000	0.225	12.500	7.870
12	18.900	19.167	18.900	0.117	12.500	7.870
13	21.700	22.425	21.700	0.094	12.500	7.870

3.9-3 示例图片

3.10 保存分类器



3.10.1 描述

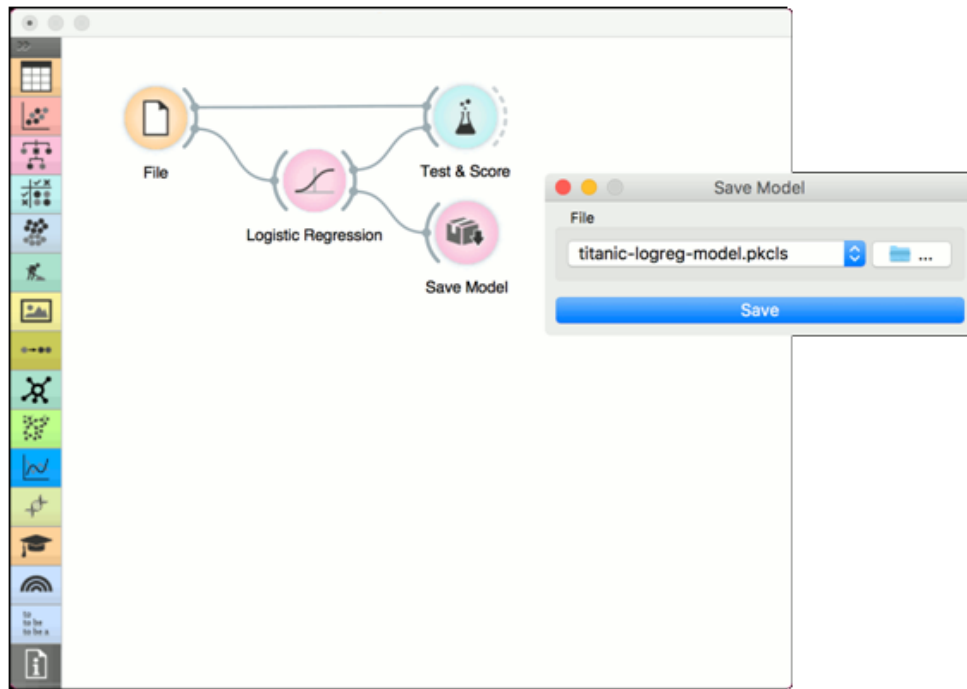


3.10-1 Save Model 窗口

1. 从先前保存的模型中选择。
2. 使用浏览图标保存创建的模型。
3. 保存模型。

3.10.2 示例：

当您保存自定义模型时，将数据提供给模型（例如逻辑回归），并将其连接到保存模型。

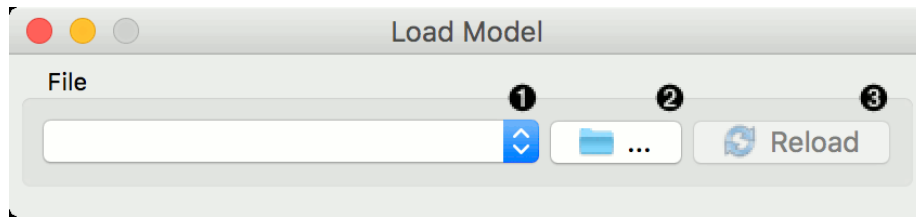


3.10-2 示例图片

3.11 加载分类器



3.11.1 描述

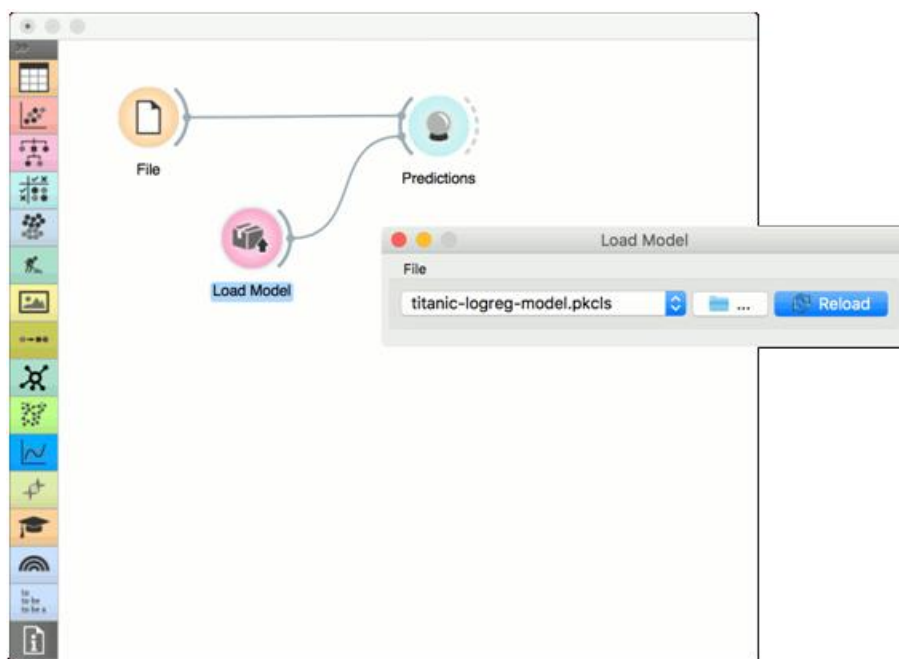


3.11-1 Load Model 窗口

1. 从以前使用的模型列表中进行选择。
2. 浏览保存的模型。
3. 重新加载所选模型。

3.11.2 示例：

当您要使用之前保存的自定义模型时，打开“加载分类器”组件，并使用“浏览”图标选择所需的文件。此组件将现有模型加载到“预测”组件中。



3.11-2 示例图片

4 回归

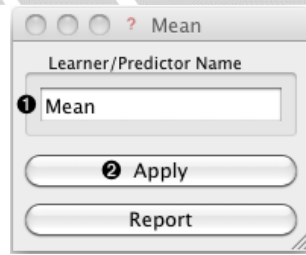
 均值学习法	 K 近邻学习法	 回归树学习法	 随机森林学习法
 支持向量机学习法	 线性回归学习法	 Adaboost 学习法	 随机梯度下降学习法
 多项式回归学习法			

4.1 均值学习法



4.1.1 描述

这是用于回归问题的最简单的学习法组件。它学习类变量的平均值并产生一个总是预测该值的预测器。



4.1-1 Mean 窗口

1. 命名。
2. 制作报告。

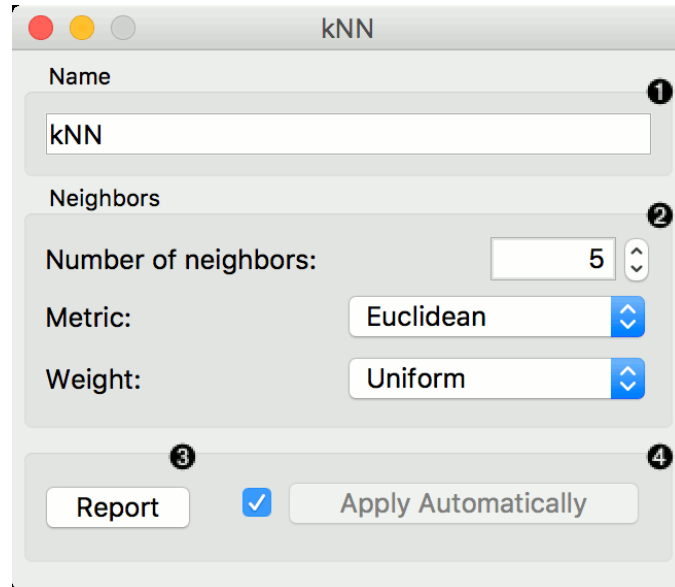
4.2 K 近邻学习法



k 最近邻(kNN)学习法

4.2.1 描述

kNN 组件使用 kNN 算法，在特征空间中搜索 k 个最接近的训练样本，并使用它们的平均值作为预测。



4.2-1 KNN 窗口

1. 命名。默认名称为“kNN”。
2. -设置最近邻居的数量
 - a) -距离参数（度量）和权重作为模型标准。
 - b) o 欧几里德（“直线”，两点之间的距离）
 - c) o 曼哈顿（所有属性的绝对差异之和）
 - d) o 最大（属性之间绝对差异最大）
 - e) o 马氏距离（点与分布之间的距离）。

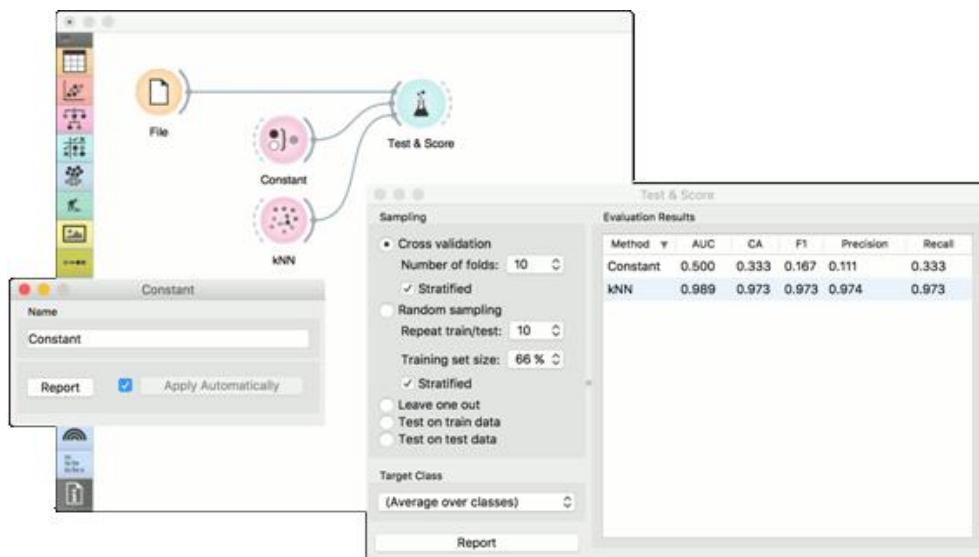
-重量

- f) o 一致：每个邻域中的所有点 均加权相等。
- g) o 距离：查询点的邻近的邻居比比其他邻居影响更大。

3. 生成报告。
4. 当您更改一个或多个设置时，您需要单击 Apply（应用），也可以通过单击应用按钮左侧的框自动应用更改。

4.2.2 示例

第一个例子是 iris 数据集的分类任务。我们将 k-最近邻居的结果与常量组件进行比较。



4.2-2 示例图片

第二个例子是回归任务。我们将 kNN 预测模型输入到预测组件中并观察预测值。

The screenshot displays a workflow in a machine learning software. A 'File' component is connected to a 'kNN' component, which is then connected to a 'Predictions' component. A configuration window for the 'kNN' model is open, showing the following settings:

- Name: kNN
- Neighbors: Number of neighbors: 5
- Metric: Euclidean
- Weight: Uniform
- Buttons: Report, Apply Automatically (checked)

Below the workflow, a 'Predictions' window shows a table of results for 13 instances. The table includes columns for the model name (kNN) and several target variables (MEDV, CRIM, ZN, INDUS, CHAS).

	kNN	MEDV	CRIM	ZN	INDUS	CHAS
1	21.780	24.000	0.006	18.000	2.310	0.000
2	22.900	21.600	0.027	0.000	7.070	0.000
3	25.360	34.700	0.027	0.000	7.070	0.000
4	26.060	33.400	0.032	0.000	2.180	0.000
5	27.100	36.200	0.069	0.000	2.180	0.000
6	27.100	28.700	0.030	0.000	2.180	0.000
7	20.880	22.900	0.088	12.500	7.870	0.000
8	19.100	27.100	0.145	12.500	7.870	0.000
9	18.400	16.500	0.211	12.500	7.870	0.000
10	19.480	18.900	0.170	12.500	7.870	0.000
11	19.280	15.000	0.225	12.500	7.870	0.000
12	22.000	18.900	0.117	12.500	7.870	0.000
13	24.340	21.700	0.094	12.500	7.870	0.000

4.2-3 示例图片

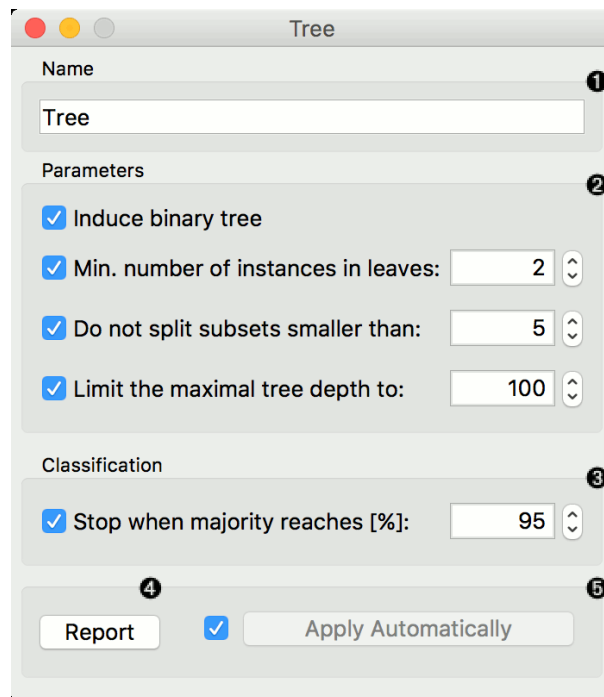
4.3 回归树



回归树

4.3.1 描述

树是一种简单的算法，它通过类纯度将数据分解成节点。它是随机森林的前身。Mining 中的树可以处理离散和连续的数据集。



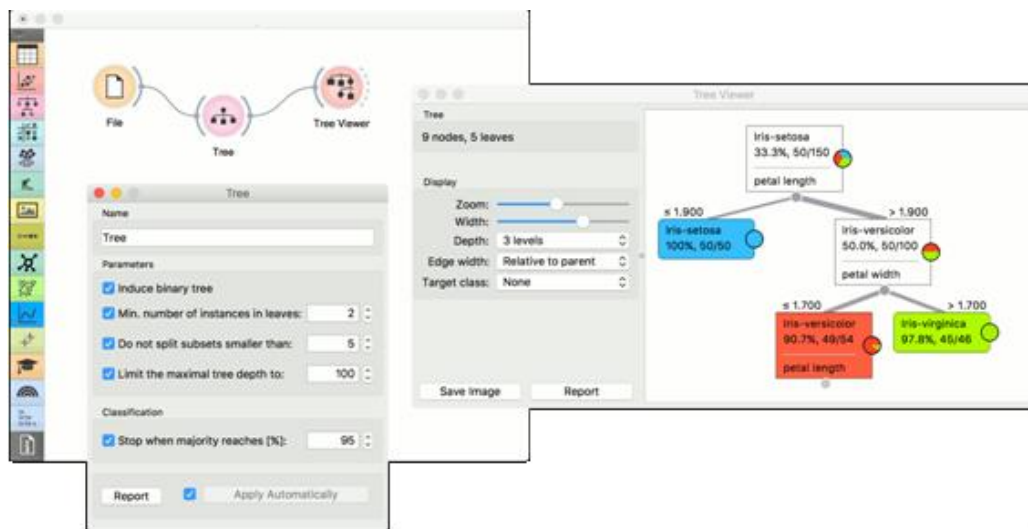
4.3-1 Tree 窗口

1. 命名。默认名称为“树”。
2. 树参数：
 - 诱导二叉树：构建一个二叉树（分为两个子节点）
 - 叶中的最小实例数：如果选中，算法将永远不会构建一个小于指定数量的训练样本的分割进入任何分支。
 - 不要拆分小于（ ）以下的子集：禁止用少于给定数量的实例分割节点的算法。

- 限制最大树深度：将分类树的深度限制为指定数量的节点级别。
3. 当多数达到[%]时停止：达到指定的多数阈值后，停止分割节点
 4. 制作报告。
 5. 更改设置后，您需要单击“应用”，或者，勾选左侧的框，更改将自动通知。

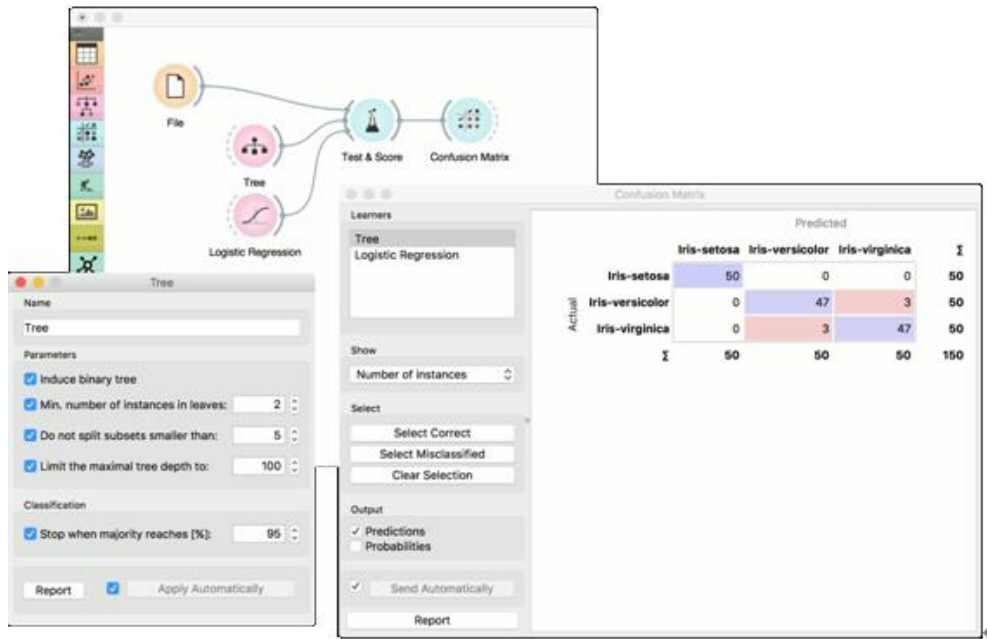
4.3.2 示例

这个小部件有两个典型的用途。首先，您可能需要引用一个模型并检查它在 Tree Viewer 中的情况。



4.3-2 示例图片

第二个模式训练一个模型，并评估其与逻辑回归性能。



4.3-3 示例图片

我们在这两个例子中都使用了 iris 数据集。然而，Tree 也用于回归任务。使用 housing 数据集并将其传递给树。来自 Tree Viewer 的选择的树节点显示在散点图中，我们可以看到所选示例具有相同的特征。



4.3-4 示例图片

4.4 随机森林回归

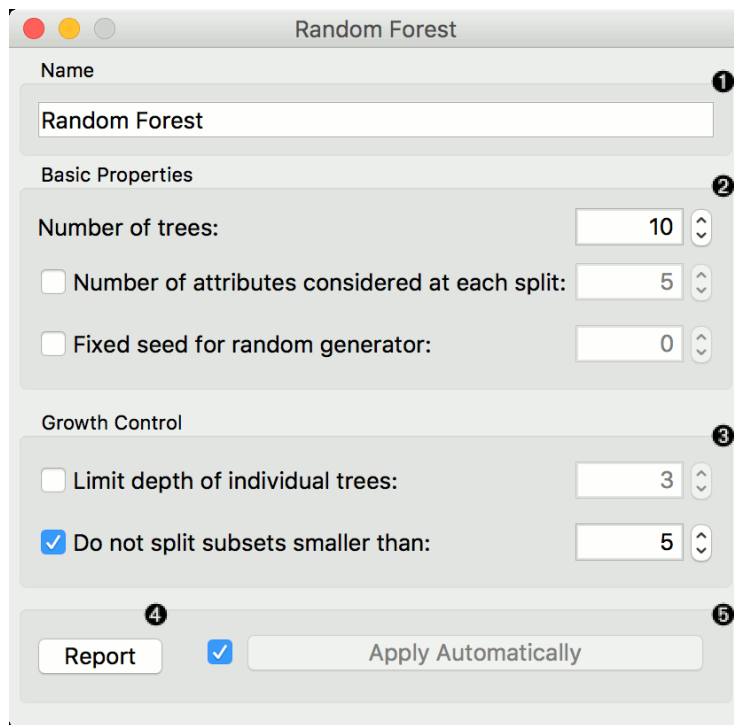


随机森林回归

4.4.1 描述

随机森林是一种用于分类、回归等任务的集成学习方法。它首先是由 Tin Kam Ho 提出并由 Leo Breiman 和 Adele Cutler 进一步发展。

随机森林建立一组决策树。每个树从引导样本的训练数据开发的。在开发单个树时，绘制任意属性子集（即“随机”），从中选择分割的最佳属性。最后的模型是基于在森林中单独开发的树木的多数结果。

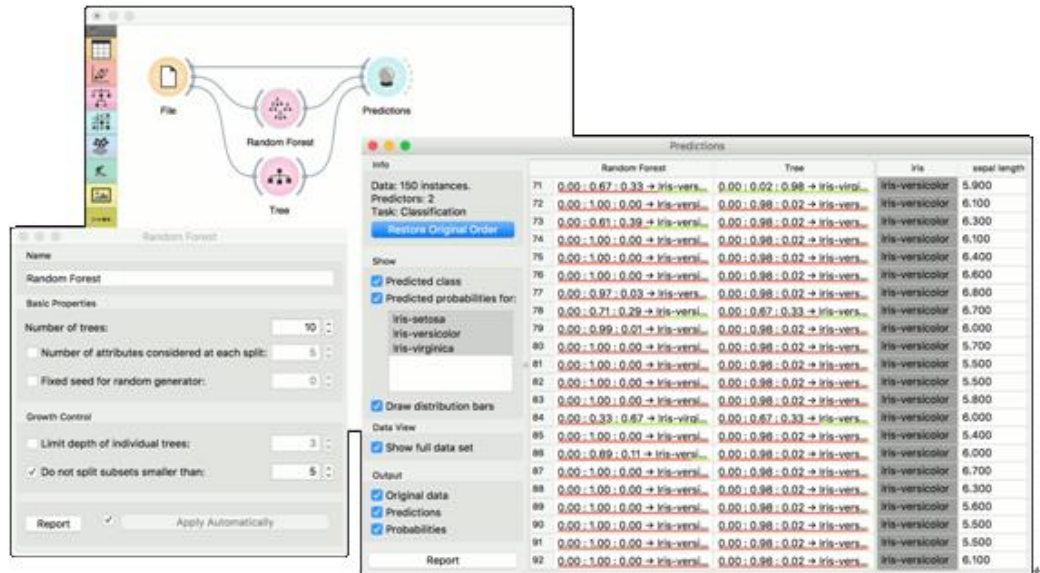


4.4-1 Random Forest 窗口

1. 命名。默认名称为“随机森林”。
2. 指定森林中将包括多少个决策树（森林中的树数），以及任意绘制多少属性以供每个节点考虑。如果未指定后者，则该数字等于数据中属性数的平方根。
3. 最初 Brieman 的建议是对决策树不做任何预先设置，但是由于预处理工作很好，速度更快，用户可以设置随机森林的深度（限制单棵树的深度）。另一个预处理是选择可以拆分的最小子集（不要拆分子集小于）。
4. 生成报告。
5. 单击应用将更改通知给其他组件。或者，勾选“应用”按钮左侧的框，更改将自动进行通信。

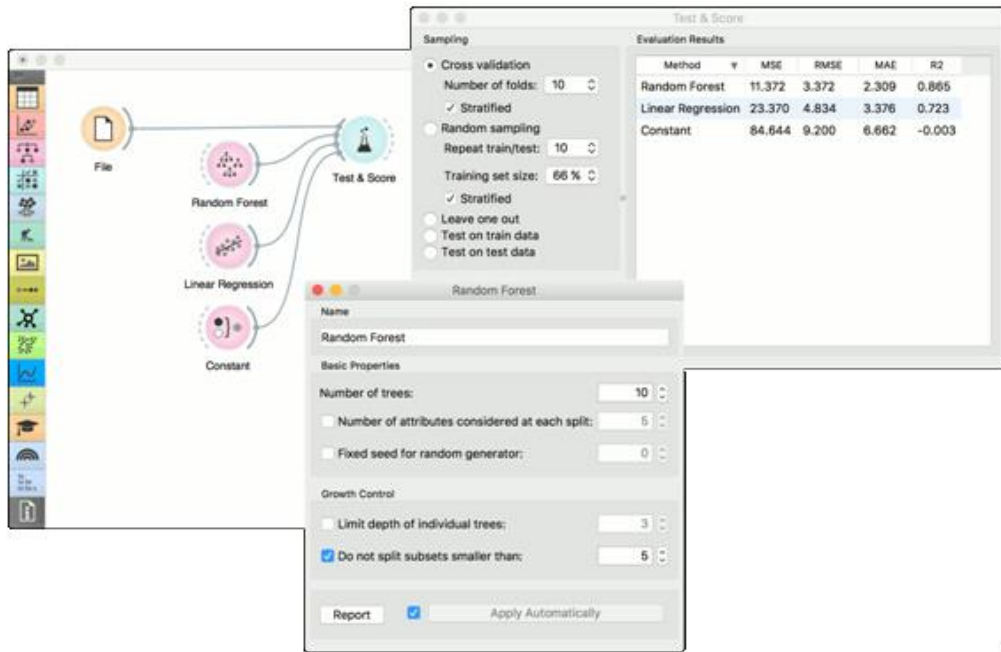
4.4.2 示例

对于分类任务，我们使用 iris 数据集。将其连接到预测组件。然后将文件连接到随机林和树，并将它们进一步连接到预测。最后，观察两个模型的预测。



4.4-2 示例图片

对于回归任务，我们将使用 housing 数据。在这里，我们将在测试和分数组件中比较不同的模型，即随机森林，线性回归和恒定组件。



4.4-3 示例图片

4.5 支持向量机

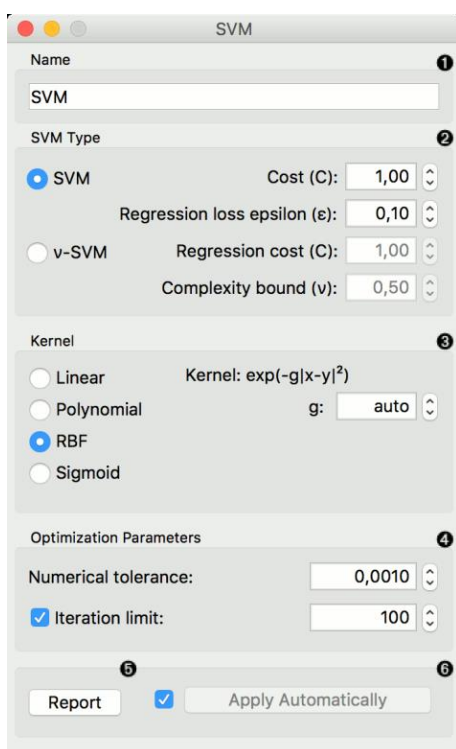


支持向量机将输入映射到更高维特征空间

4.5.1 描述

支持向量积 (SVM) 是一种流行的分类技术，它将在属性空间中构造一个分离的超平面，以最大化不同类的实例之间的边界。该技术往往会产生最高的预测性能结果。Mining 在 LIBSVM 程序包中嵌入了 SVM 的一个流行的实现，这个组件为它的功能提供了一个图形用户界面。

对于回归任务，SVM 使用 ϵ 不敏感的损失在高维特征空间中执行线性回归。其估计精度取决于 C ， ϵ 和核参数的良好设置。组件基于 SVM 回归输出类预测。该组件适用于分类和回归任务。



4.5-1 SVM 窗口

1. 命名。默认名称为“SVM”。
2. 具有测试错误设置的 SVM 类型。SVM 和 ν -SVM 基于误差函数的差异最小化。在右侧，您可以设置测试错误范围：
 - a) \emptyset SVM：
 - b) 成本：损失的罚款项，适用于分类和回归任务。
 - c) ϵ ：epsilon-SVR 模型的参数适用于回归任务。定义与真实值的距离，其中没有惩罚值与预测值相关联。
 - d) $\emptyset\nu$ -SVM：
 - e) 成本：损失的罚款项，仅适用于回归任务
 - f) ν ： ν -SVR 模型的参数，适用于分类和回归任务。训练误差分数的上限和支持向量分数的下限。
3. 核是将属性空间转换为一个新的特征空间以适应最大边值超平面的一种函数，因此允许算法使用以下方式创建模型：
 - a) 线性
 - b) 多项式
 - c) RBF 和
 - d) Sigmoid

4. 在数值公差中设置允许偏离预期值。勾选迭代限制旁边的框，以设置允许的最大迭代次数。
5. 制作报告。
6. 单击应用以提交更改。如果勾选“应用”按钮左侧的框，则会自动进行更改。

4.5.2 示例

示例显示了如何使用 SVM 与散点图组合。以下工作流程对 iris 数据集进行 SVM 模型的训练，并输出支持向量，这些向量是在学习阶段用作支持向量的那些数据实例。我们可以在散点图可视化中观察这些数据实例的哪些。请注意，要使工作流正常工作，您必须设置组件之间的链接，如下面的示例所示。

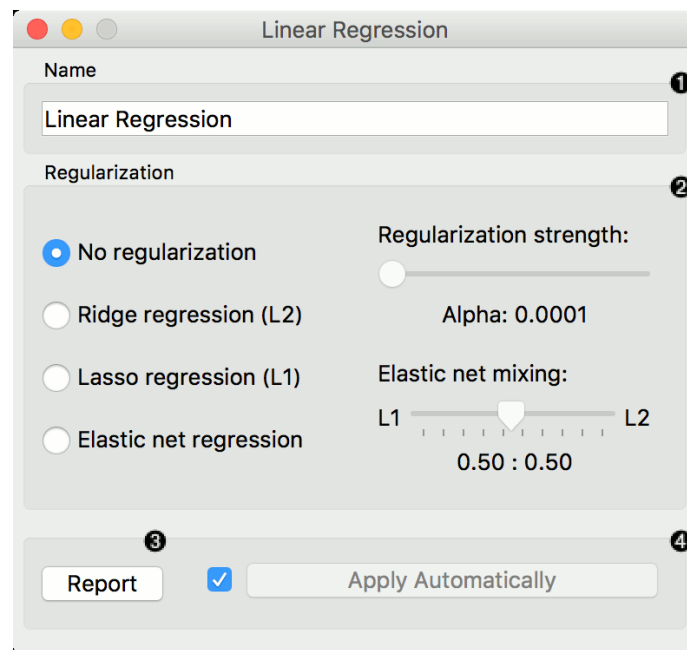
4.6 线性回归学习法



学习它的输入数据的线性函数。

4.6.1 描述

线性回归小部件构建学习者/预测器，从其输入数据中学习线性函数。该模型可以识别预测因子 x 和响应变量 y 之间的关系。另外，可以指定 Lasso 和 Ridge 正则化参数。线性方程仅适用于回归任务。



4.6-1 Linear Regression 窗口

1. 命名
2. 选择一个模型来训练：
 - a) 没有正规化
 - b) 一个脊正规化 (L2 范数惩罚)
 - c) Lasso 约束 (L1 范数惩罚)
 - d) 弹性网络正则化

3. 制作报告。
4. 按应用程序提交更改。如果自动应用勾选，更改自动提交。

4.6.2 示例

下面，是一个 housing 数据集的简单 workflow。我们训练的线性回归和随机森林和评估其性能测试及评分。



4.6-2 示例图片

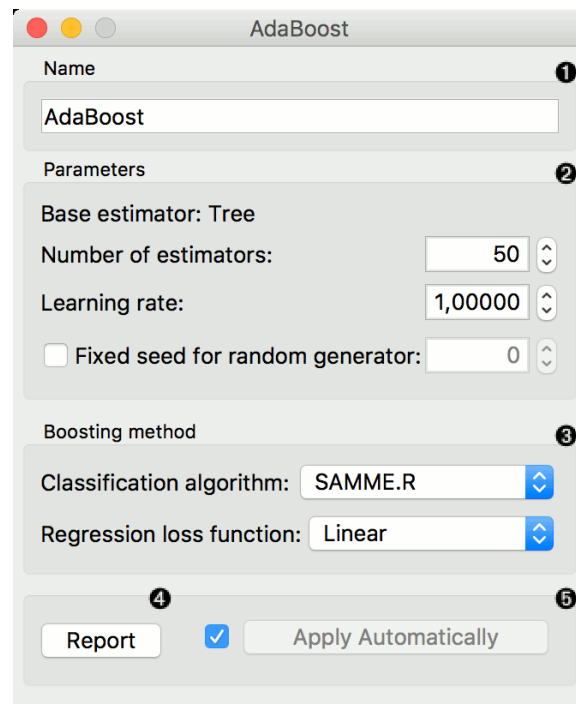
4.7 Adaboost 学习法



4.7.1 描述

AdaBoost (“自适应增强”) 小部件的简写是由 Yoav Freund 和 Robert Schapire 制定的机器学习算法。它可以与其他学习算法一起使用来提高其性能。它通过调整弱势的学习者来做到这一点。

AdaBoost 适用于分类和回归。



4.7-1 AdaBoost 窗口

1. 命名。默认名称为 “AdaBoost” 。
2. 设置参数。您可以设置：

-估计量数

学习率：它确定新获得的信息将在多大程度上覆盖旧信息（0 =代理不会学习任何东西，1 =代理仅考虑最新信息）

用于随机发生器的固定种子：设置固定的 “种子” 以使得能够再现结果。

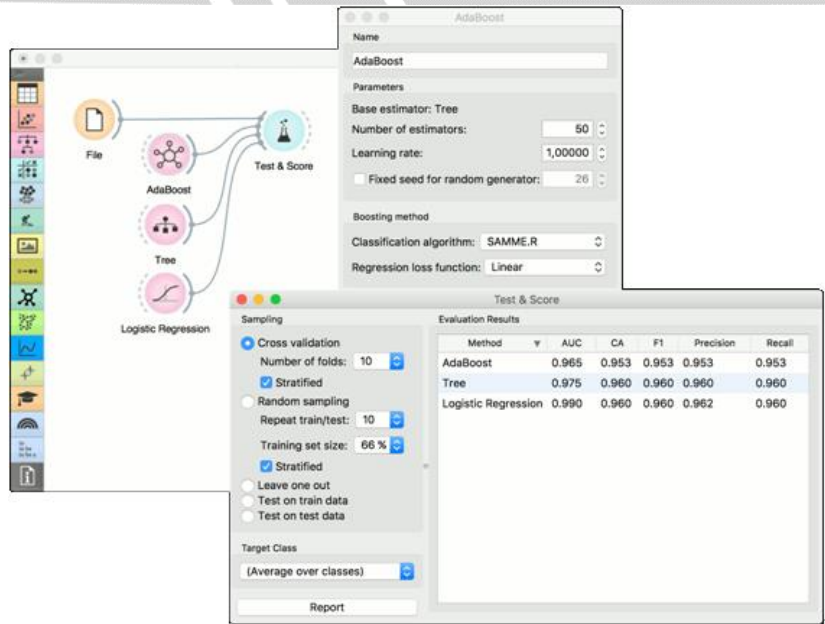
-升压方式

分类算法（如果输入分类）：SAMME（使用分类结果更新基本估计器的权重）或 SAMME.R（使用概率估计更新基本估计器的权重）。

3. 回归损失函数（如果输入回归）：Linear（ ）， Square（ ）， Exponential（ ）。
4. 生成报告。
5. 更改设置后，单击应用。要自动通知更改自动勾选自动应用。

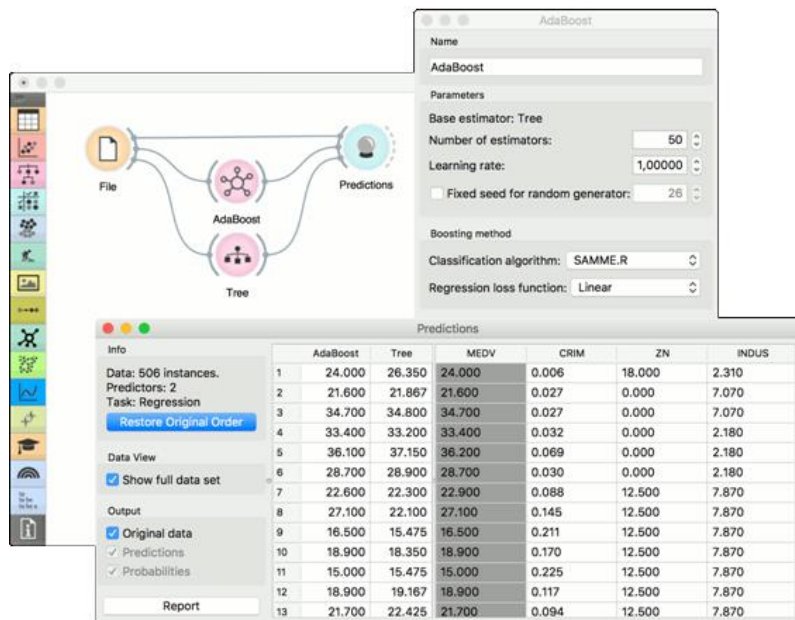
4.7.2 示例

对于分类，我们加载了 Iris 数据集。 我们使用 AdaBoost，分类树和逻辑回归组件，并评估模型在“测试和分数”中的表现。



4.7-2 示例图片

对于回归，我们加载了 housing 数据集，将数据实例发送到两个不同的模型（AdaBoost 和 Tree），并将它们输出到“预测”组件中。



4.7-3 示例图片

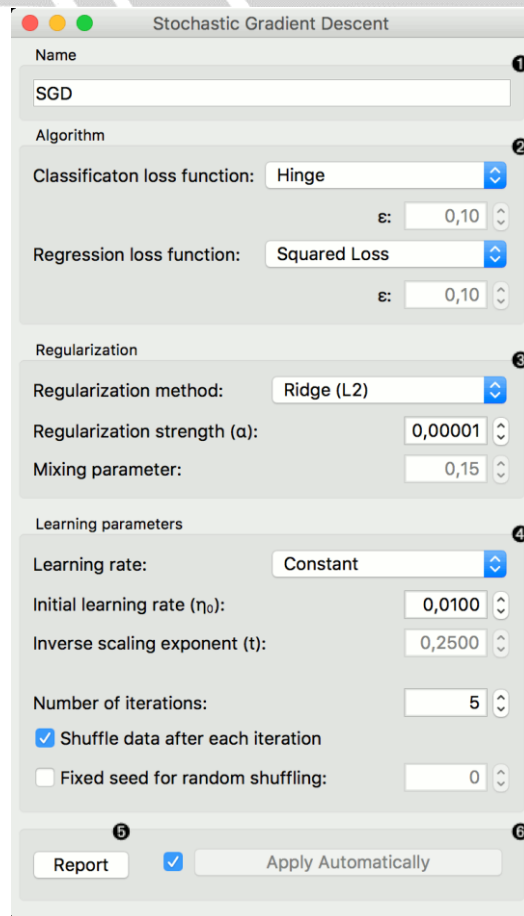
4.8 随机梯度下降学习法



使用随机下降梯度来最小化目标函数

4.8.1 描述：

随机梯度下降使用随机梯度下降，最大限度地减少选定的损失函数的线性函数。该算法通过一次考虑一个样本来逼近真实梯度，并且基于损失函数的梯度同时更新模型。对于回归，它返回预测值作为和的最小值，即 M 估计量，并且对于大规模和稀疏数据集特别有用。



4.8-1 Stochastic Gradient Descent 窗口

1. 指定模型的名称。默认名称为“SGD”。
2. 算法参数。分类损失函数：
 - a) 铰链（线性 SVM）
 - b) 逻辑回归（逻辑回归 SGD）
 - c) 改进的 Huber（平滑损失，使异常值容忍以及概率估计）
 - d) 方形铰链（二次铰链）

- e) 感知器 (感知器算法使用的线性损耗)
 - f) 平方损失 (适合普通最小二乘法)
 - g) Huber (切换到 ϵ 以外的线性损耗)
 - h) Epsilon 不敏感 (忽略 ϵ 内的错误, 线性超出它)
 - i) 平方 ϵ 不敏感 (损失平方超出 ϵ 区)。
 - j) 回归损失函数 :
 - k) 平方损失 (适合普通最小二乘法)
 - l) Huber (切换到 ϵ 以外的线性损耗)
 - m) Epsilon 不敏感 (忽略 ϵ 内的错误, 线性超出它)
 - n) 平方 ϵ 不敏感 (损失平方超出 ϵ 区)。
3. 防止过度拟合的正规化规范 :
- a) 没有
 - b) Lasso (L1) (L1, 导致稀疏解)
4. 正脊 (L2) (L2, 标准矫正器)
- a) 弹性网 (混合罚款规范)。

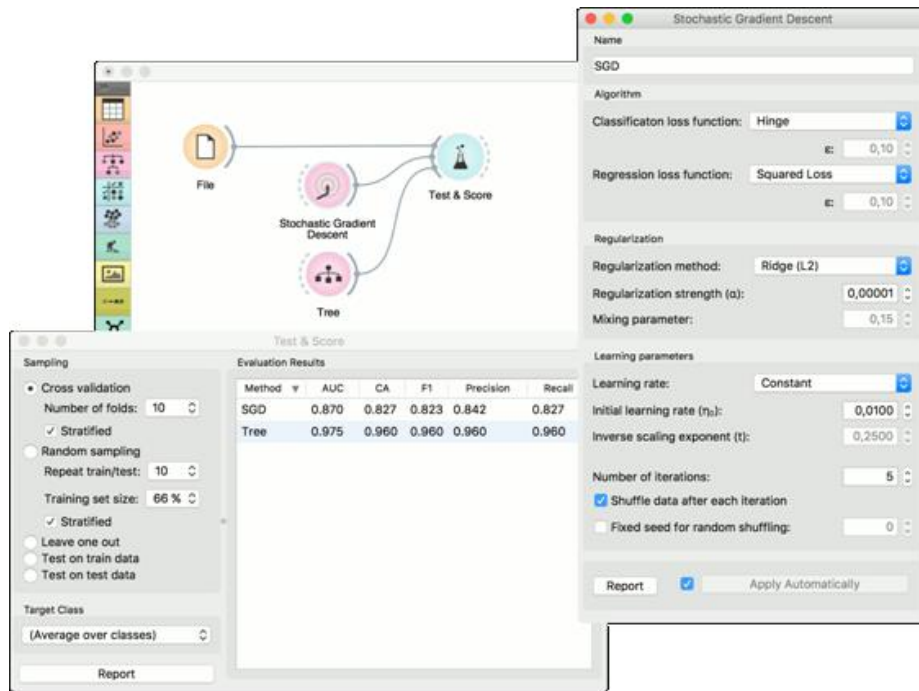
正则化强度定义将应用多少正则化 (我们正规化越少, 模型适应数据越多) 和混合参数 L1 和 L2 损耗之间的比率 (如果设置为 0, 则损耗为 L2, 如果设置为 1, 则为 L1)。

学习参数

- b) 学习率：
 - c) 常数：学习率在所有时代保持不变（通过）
 - d) 最佳：Leon Bottou 提出的启发式
 - e) 反向缩放：收益率与迭代次数成反比
 - f) 初始学习率。
 - g) 反向缩放指数：学习率衰减。
 - h) 迭代次数：通过训练数据的次数。
5. 生成报告。
6. 按应用提交更改。或者，勾选“应用”按钮左侧的框，更改将自动进行通信。

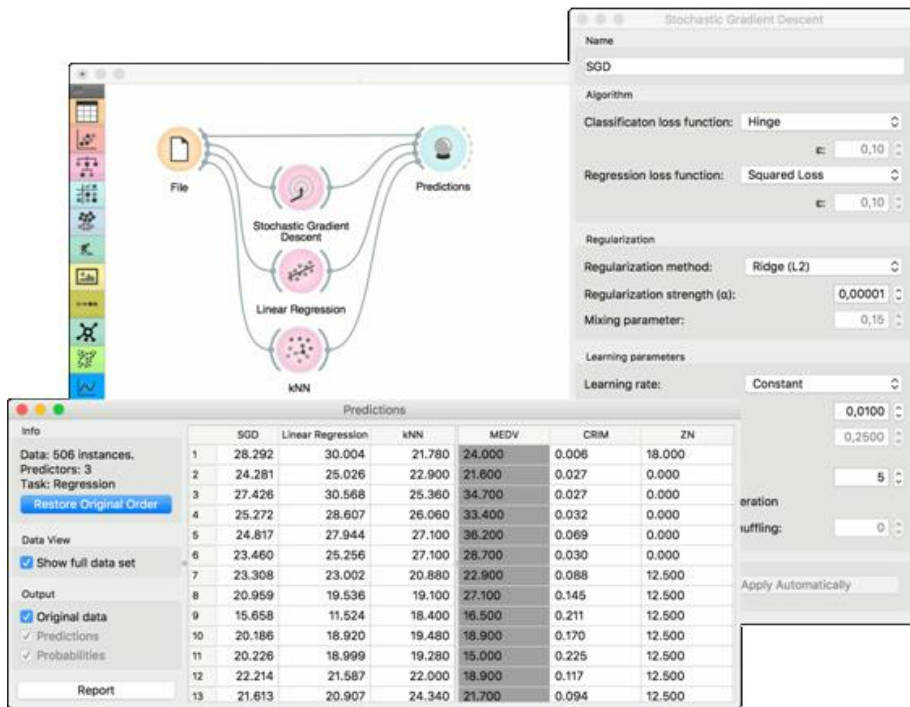
4.8.2 示例：

对于分类任务，我们将使用 Iris 数据集并对其进行两个模型测试。我们连接随机梯度下降和分类树到测试和分数组件。我们还将文件连接到测试和分数，并在组件中观察模型性能。



4.8-2 示例图片

对于回归任务，我们将比较三种不同的模型来看哪一种是什么样的结果。为了本示例的目的，使用了保留数据集。我们将“文件”组件连接到随机梯度下降，线性回归和 kNN 组件，并将所有四个连接到“预测”小部件。



4.8-3 示例图片

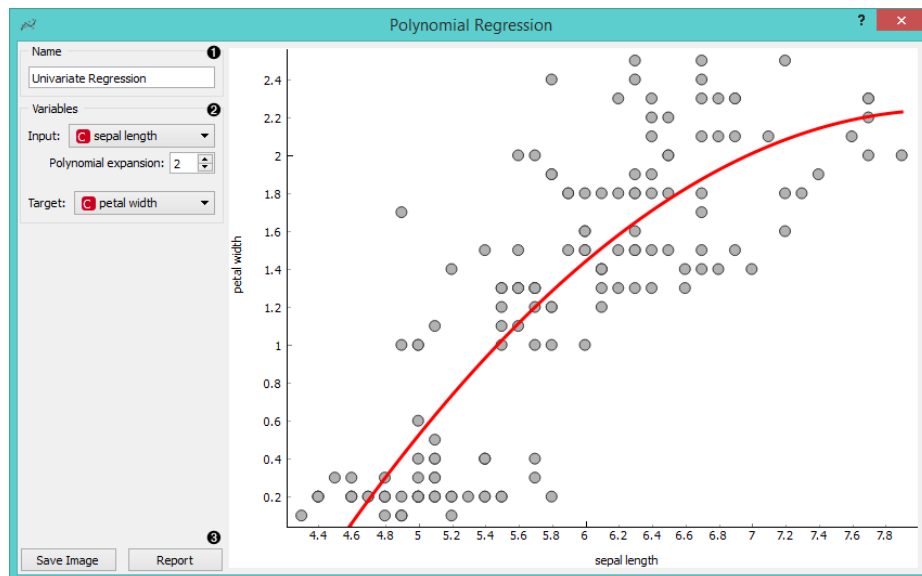
4.9 多项式回归



教育组件-交互式显示不同回归的回归线。

4.9.1 描述

在组件中，可以设置多项式展开。多项式扩展是用于变换输入数据的多项式程度的调节，并且对曲线的形状有影响。如果多项式展开设置为 1，则表示在回归中使用未转换的数据。

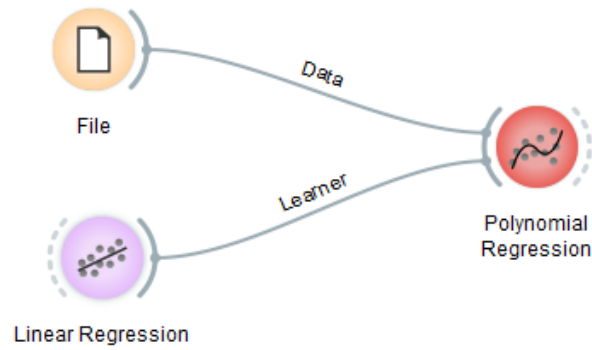


4.9-1 Polynomial Regression 窗口

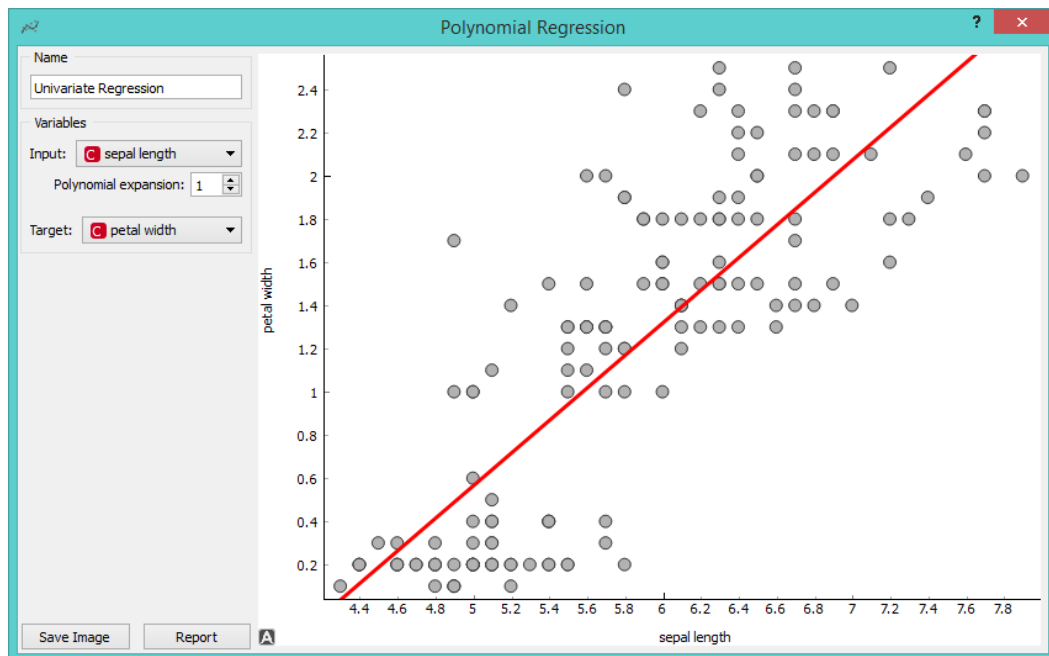
1. 回归命名。
2. 输入：
 - a) 轴 X 上的独立变量
 - b) 多项式展开：多项式展开的程度。
 - c) 目标：轴上的从属变量
3. 保存图像，将图像保存在 SVG 或 PNG 格式中。

报告包括小部件参数和可视化的报告。

4.9.2 示例：

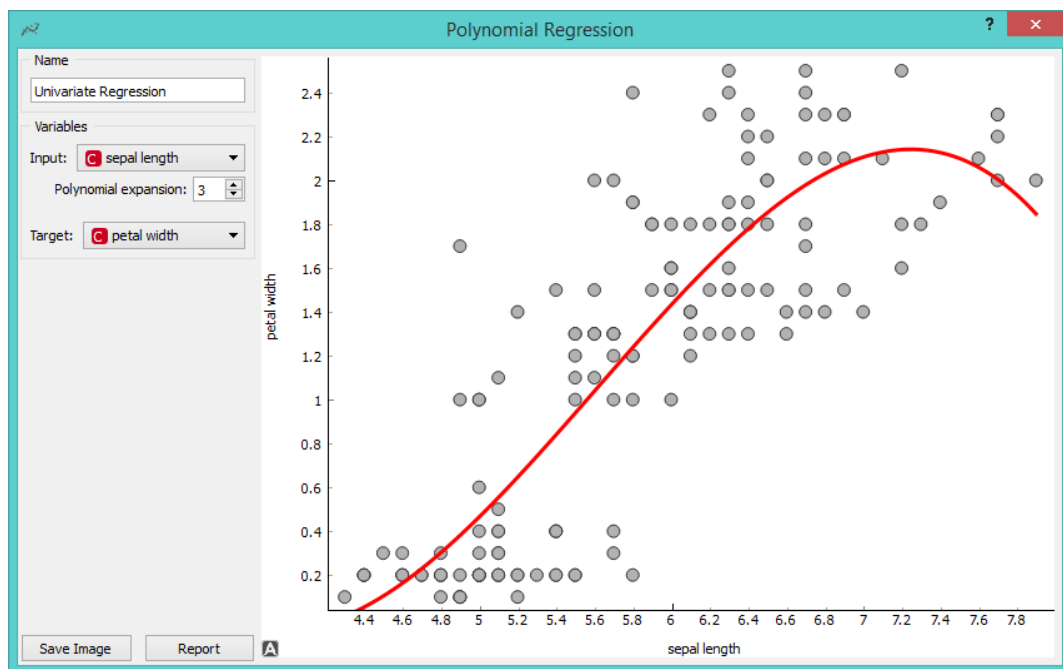


我们用文件组件加载了 Iris 数据集。然后我们将线性回归学习者连接到多项式回归组件。在组件中，我们选择花瓣长度作为我们的输入值和花瓣宽度作为我们的目标变量。我们将多项式扩展设置为 1，给出了线性回归线。结果如下图所示。



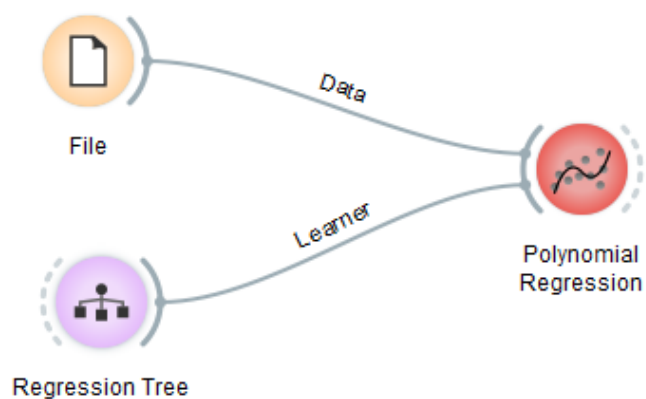
4.9-2 示例图片

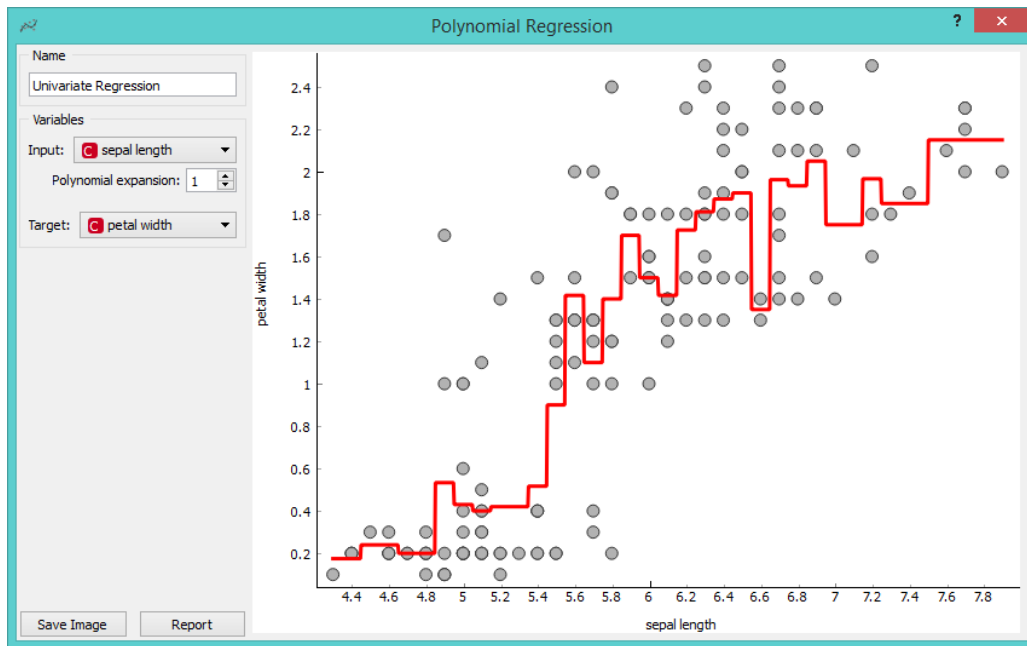
如果我们增加多项式展开参数，该线可以更好地适应。我们设置为 3。



4.9-3 示例图片

为了观察不同的结果，将线性回归改为其他回归学习者。下面的例子是用回归树学习者完成的。





4.9-4 示例图片

5 评估

 测试学习法	 预测	 混淆矩阵	 ROC 分析
 升力曲线	 校准图		

5.1 测试学习法

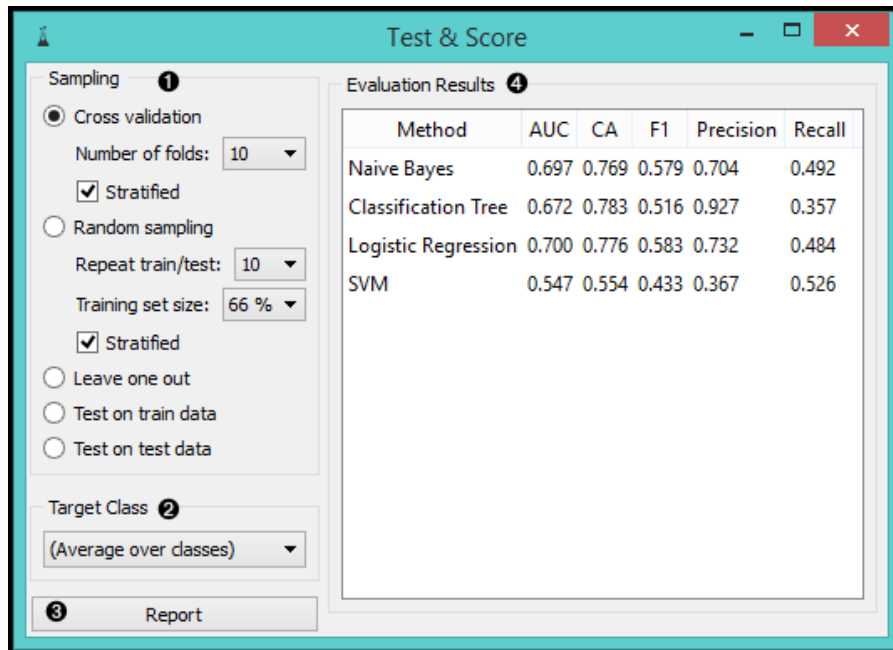


测试数据的学习算法。

5.1.1 描述

测试学习算法组件提供不同的采样方案，包括使用单独的测试数据。组件做了两件事。首先，它显示了一个具有不同分类器性能测量的表，如分类精度和曲线下面积。第二，它输出评估结果，可以由其他小部件用于分析分类器的性能，如 ROC 分析或混淆矩阵。

学习者信号具有不常见的属性：它可以连接到多个小部件，以相同的过程测试多个学习者。



5.1-1 Test & Score 窗口

1. 小部件支持各种抽样方法。

交叉验证将数据分解为给定数量的折叠（通常为 5 或 10）。该算法通过一次从一个折叠的例子进行测试；该模型是从其他折叠引起的，并且来自保持的折叠的示例被分类。对于所有的折叠，这是重复的。

与留一法相似，但是它一次性提供一个实例，从所有其他实例引入模型，然后对已有的实例进行分类。这种方法显然非常稳定，可靠但是非常慢。

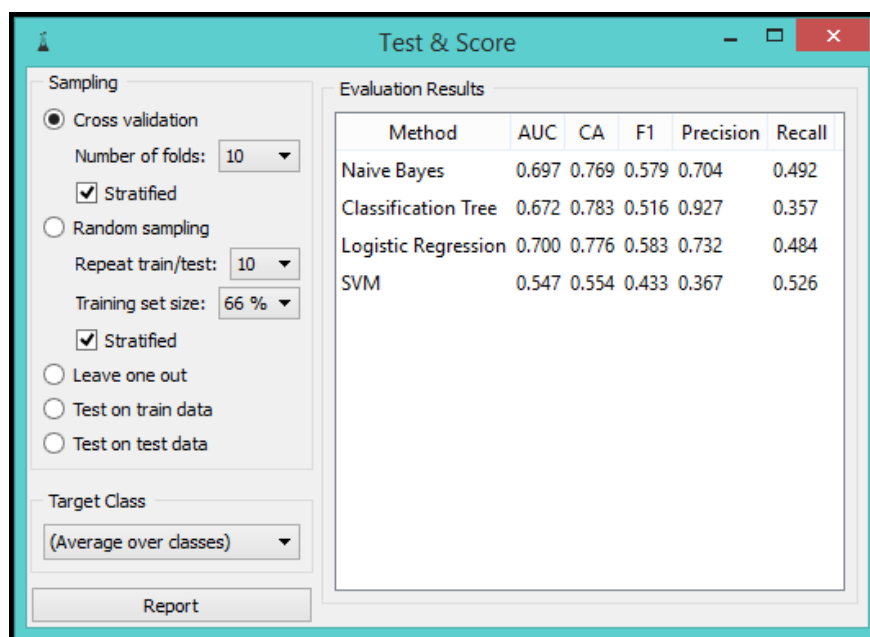
随机抽样按照给定的比例将数据随机分成培训和测试集（例如 70:30）；整个过程重复指定次数。

在训练数据上使用整个数据集进行训练，然后进行测试。这种方法实际上总是给出错误的结果。

在测试数据上测试：上述方法仅使用来自数据信号的数据。要使用测试示例（例如从另一个文件或另一个组件中选择的某些数据）输入另一个数据集，我们在通信通道中选择“单独测试数据”信号，然后选择测试测试数据。

2. 仅测试测试数据需要目标类，例如。患有这种疾病或属于亚视力 Iris 病毒。当 Target 类为 (None) 时，方法返回平均值。可以在窗口小部件的底部选择目标类。
3. 生成报告。
4. 小部件将计算许多性能统计信息：

5.1.2 分类：



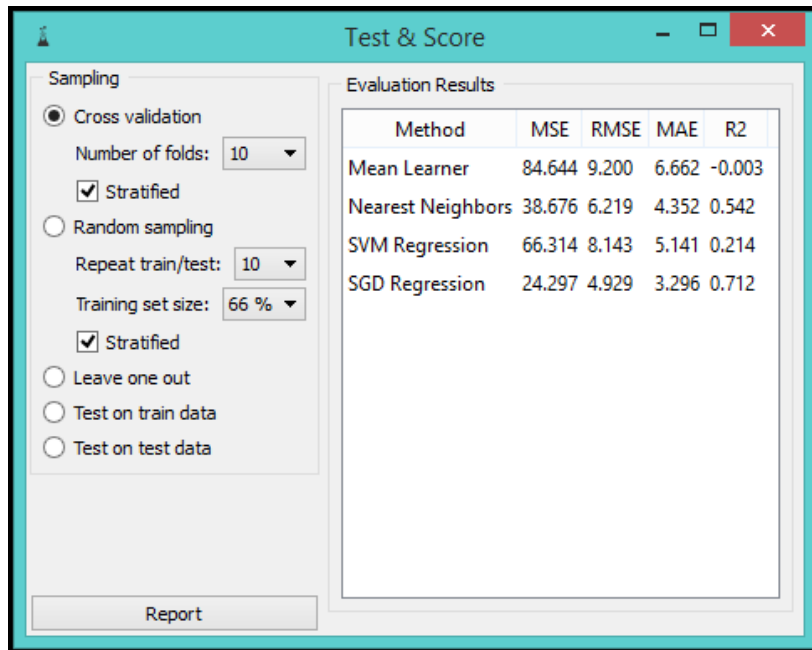
The screenshot shows a software window titled "Test & Score". On the left, there are two sections: "Sampling" and "Target Class". The "Sampling" section has radio buttons for "Cross validation" (selected), "Random sampling", "Leave one out", "Test on train data", and "Test on test data". It also includes checkboxes for "Stratified" (checked) and "Repeat train/test:" (set to 10). The "Target Class" section has a dropdown menu set to "(Average over classes)". A "Report" button is at the bottom left. On the right, the "Evaluation Results" section contains a table with columns: Method, AUC, CA, F1, Precision, and Recall. The table lists four methods: Naive Bayes, Classification Tree, Logistic Regression, and SVM, with their respective performance metrics.

Method	AUC	CA	F1	Precision	Recall
Naive Bayes	0.697	0.769	0.579	0.704	0.492
Classification Tree	0.672	0.783	0.516	0.927	0.357
Logistic Regression	0.700	0.776	0.583	0.732	0.484
SVM	0.547	0.554	0.433	0.367	0.526

5.1-2 Test & Score 窗口 (分类)

- ROC 下的区域是接收器操作曲线下的面积。
- 分类精度是正确分类实例的比例。
- F-1 是精度和召回的加权调和平均值。
- 精确度是分类为阳性的实例中真阳性的比例。
- 回想数据中所有阳性实例中真阳性的比例。

5.1.3 回归：



5.1-3 Test & Score 窗口 (回归)

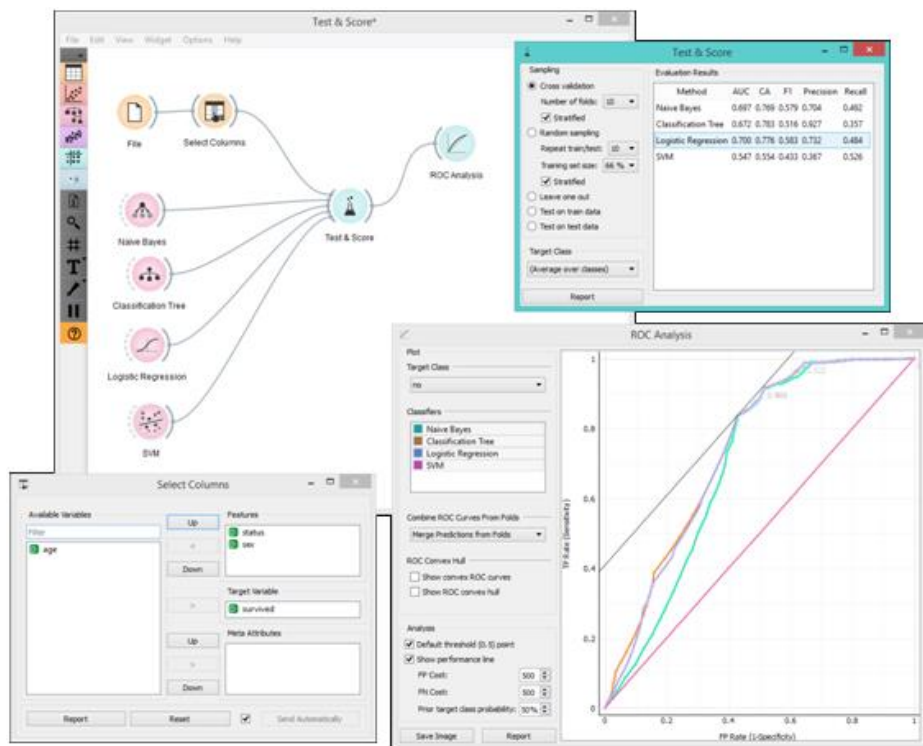
- MSE 测量误差或偏差平方的平均值 (估计量与估计值之间的差值) 。
- RMSE 是一组数字的平方的算术平均值的平方根 (估计器对数据的拟合的不完美的度量)

•MAE 用于衡量预测或预测对最终结果的接近程度。

•R2 被解释为从独立变量可预测的因变量中的方差的比例。

5.1.4 示例

在组件的使用中，我们给它一个数据集和一些学习算法，我们在 Test & Score 组件和 ROC 中的表格中观察他们的表现。数据经常在测试前进行预处理；在这种情况下，我们在 Titanic 数据集上进行了一些手动功能选择（选择列小部件），在这里我们只想知道幸存的性别和状态，并忽略年龄。



5.1-4 示例图片

5.2 预测



显示模型对数据的预测。

5.2.1 描述

组件接收数据集和一个或多个预测变量（分类器，而不是学习算法 - 请参见下面的示例）。

它输出数据和预测。

	Classification Tree	Naive Bayes	iris	sepal length	sepal width	petal length	petal width
1	Iris-setosa	Iris-setosa	Iris-setosa	4.600	3.600	1.000	0.200
2	Iris-versicolor	Iris-versicolor	Iris-versicolor	4.900	2.400	3.300	1.000
3	Iris-versicolor	Iris-versicolor	Iris-versicolor	5.700	2.800	4.100	1.300
4	Iris-setosa	Iris-setosa	Iris-setosa	5.400	3.700	1.500	0.200
5	Iris-versicolor	Iris-versicolor	Iris-versicolor	6.600	3.000	4.400	1.400
6	Iris-versicolor	Iris-versicolor	Iris-virginica	4.900	2.500	4.500	1.700
7	Iris-versicolor	Iris-versicolor	Iris-versicolor	5.500	2.500	4.000	1.300
8	Iris-virginica	Iris-virginica	Iris-virginica	5.700	2.500	5.000	2.000
9	Iris-setosa	Iris-setosa	Iris-setosa	4.600	3.100	1.500	0.200
10	Iris-versicolor	Iris-versicolor	Iris-versicolor	5.600	3.000	4.100	1.300
11	Iris-setosa	Iris-setosa	Iris-setosa	4.800	3.000	1.400	0.300
12	Iris-virginica	Iris-virginica	Iris-virginica	6.700	3.300	5.700	2.100
13	Iris-virginica	Iris-virginica	Iris-virginica	7.600	3.000	6.600	2.100
14	Iris-setosa	Iris-setosa	Iris-setosa	4.500	2.300	1.300	0.300
15	Iris-setosa	Iris-setosa	Iris-setosa	5.000	3.300	1.400	0.200
16	Iris-versicolor	Iris-virginica	Iris-virginica	7.200	3.000	5.800	1.600
17	Iris-virginica	Iris-versicolor	Iris-virginica	6.300	2.700	4.900	1.800
18	Iris-virginica	Iris-virginica	Iris-virginica	5.800	2.700	5.100	1.900
19	Iris-setosa	Iris-setosa	Iris-setosa	5.100	3.300	1.700	0.500
20	Iris-virginica	Iris-virginica	Iris-virginica	6.700	3.100	5.600	2.400

5.2-1 Predictions 窗口

1.输入信息

2. 用户可以选择分类选项。如果显示预测类被勾选,附加的数据表提供有关预测类的信息。

如果显示预测概率被勾选,则附加的数据表提供关于分类器预测的概率的信息。用户还可以选择他或她想要在附加数据表中显示的预测类。选项“绘制分配栏”提供了一个很好的可视化预测。

3. 通过勾选“显示完整数据集”,用户可以将整个数据表附加到“预测”窗口小部件。

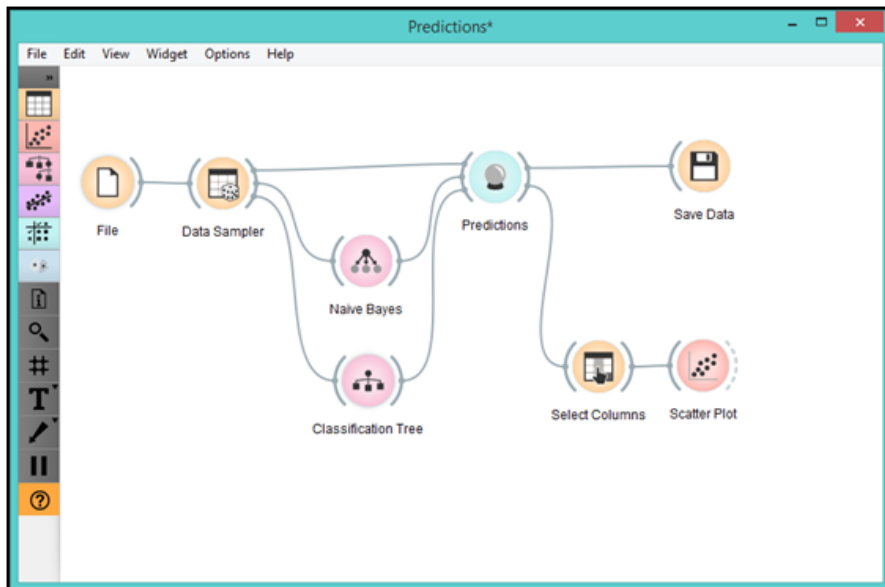
4. 选择所需的输出。

5. 附加的数据表

6. 生成报告。

尽管它很简单,但是小部件允许对预测性模型的决策进行非常有趣的分析;在页面底部有一个简单的演示。组件的输出是另一个数据集,其中的预测被追加为新的元属性。您可以选择希望输出哪些特性(原始数据、预测、概率)。所得到的数据集可以附加到小部件中,但是您仍然可以选择在一个单独的数据表中显示它。

5.2.2 示例

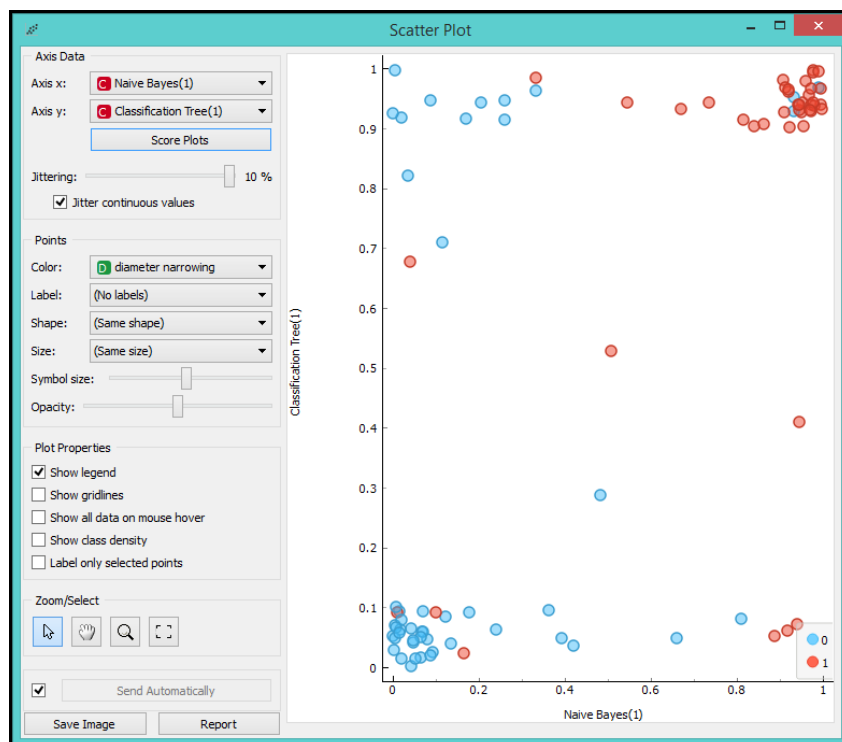


5.2-2 示例图片

我们将 iris 数据随机分为两个子集。包含 70% 的数据实例的较大的子集被发送到朴素贝叶斯和决策树，因此可以生成相应的模型。然后将其余 30% 的数据模型发送到预测中。预测显示这些例子是如何分类的。

要保存预测，我们只需将“保存”窗口小部件附加到“预测”。最终文件是一个数据表，可以保存为.tab 或.tsv 格式。

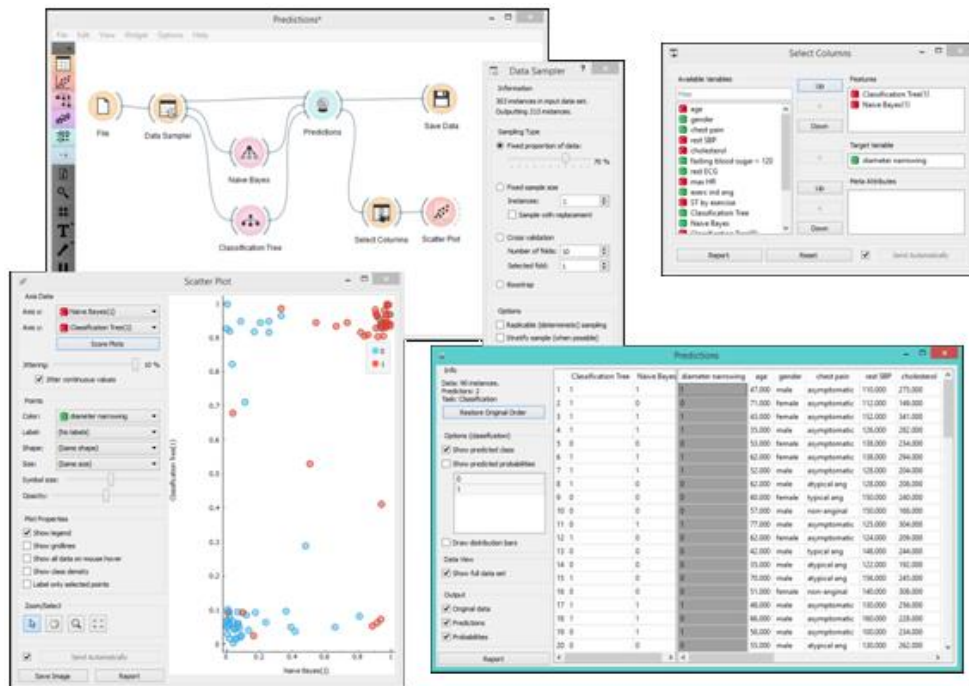
最后，我们可以分析模型的预测。为此，我们首先采用选择列，我们用特征预测将元属性移动到特征。然后将转换的数据提供给 Scatterplot，我们将其设置为将概率的属性用作 x 和 y 轴，而类（已经是默认）用于对数据点进行着色。



5.2-3 示例图片

为了获得上述的情况，我们选择抖动连续值，因为决策树给出了几个不同的概率。在左下角的蓝色点代表人的血管没有直径变窄，这是正确分类的两种模式。右上角红点代表狭窄血管的患者，两者均正确分类。

请注意，这种分析是在一个相当小的样本，所以这些结论可能是不接地。这里是整个 workflow：



5.2-4 示例图片

另一个使用这个小部件的例子是在部件混淆矩阵的文档中给出的。

5.3 混淆矩阵

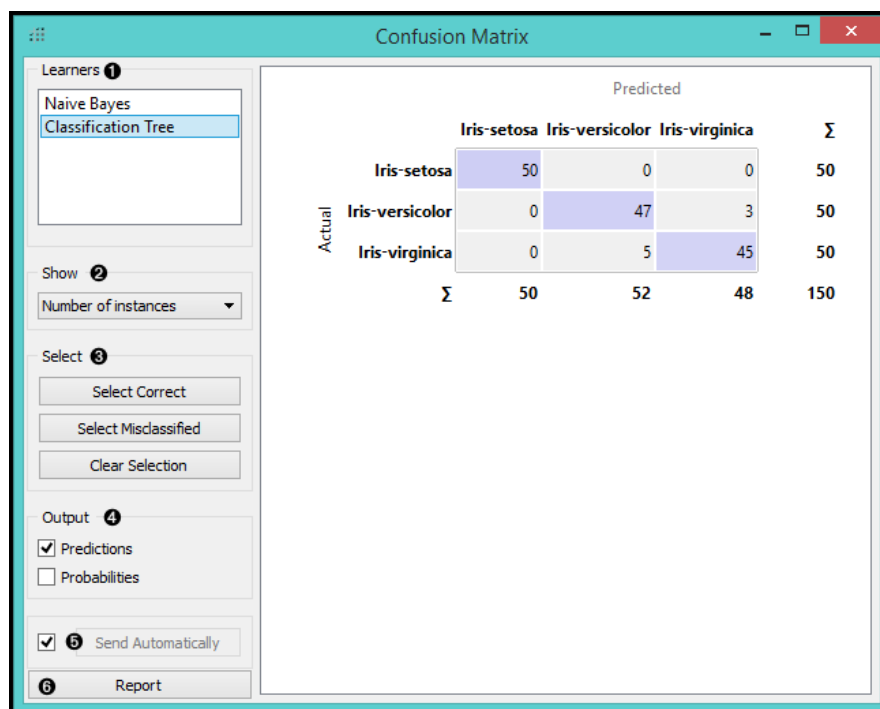


显示混淆矩阵。

5.3.1 描述

混淆矩阵 (Confusion Matrix) 给出一个类 (被分类到其他 (或相同) 的类) 中的示例数量 / 比例。除此之外, 选择矩阵的元素会将相应的示例馈送到输出信号上。这样, 用户便可以观察到哪些特定的示例被采用某种方式错误分类。

这个组件通常从 Test Learners 中获得评估结果; 下面显示了此方案的一个示例。



5.3-1 Confusion Matrix 窗口

1. 评估结果包含多种学习算法的数据时, 必须在“学习者”框中选择一种。

示例显示了在 Iris 数据上训练和测试的 Tree 和朴素贝叶斯模型的混淆矩阵。组件的右侧包含朴素的贝叶斯模型的矩阵 (因为该模型在左侧被选中)。每行对应一个正确的类, 而列表示预测类。

在显示中, 我们选择了我们希望在矩阵中看到的数据。实例的数目显示正确和错误分类的数字。

		Predicted			Σ
		Iris-setosa	Iris-versicolor	Iris-virginica	
Actual	Iris-setosa	100.0 %	0.0 %	0.0 %	50
	Iris-versicolor	0.0 %	88.7 %	6.4 %	50
	Iris-virginica	0.0 %	11.3 %	93.6 %	50
Σ		50	53	47	150

2.在选择中，您可以选择所需的输出。

- 通过选择矩阵的对角线将正确分类的实例发送到输出。
- 错误分类选择错误分类的实例。
- 没有声明选择。

如前所述，还可以选择表格中的单个单元格来选择特定类型的错误分类的实例。

		Predicted			Σ
		Iris-setosa	Iris-versicolor	Iris-virginica	
Actual	Iris-setosa	100.0 %	0.0 %	0.0 %	50
	Iris-versicolor	0.0 %	88.7 %	6.4 %	50
	Iris-virginica	0.0 %	11.3 %	93.6 %	50
Σ		50	53	47	150

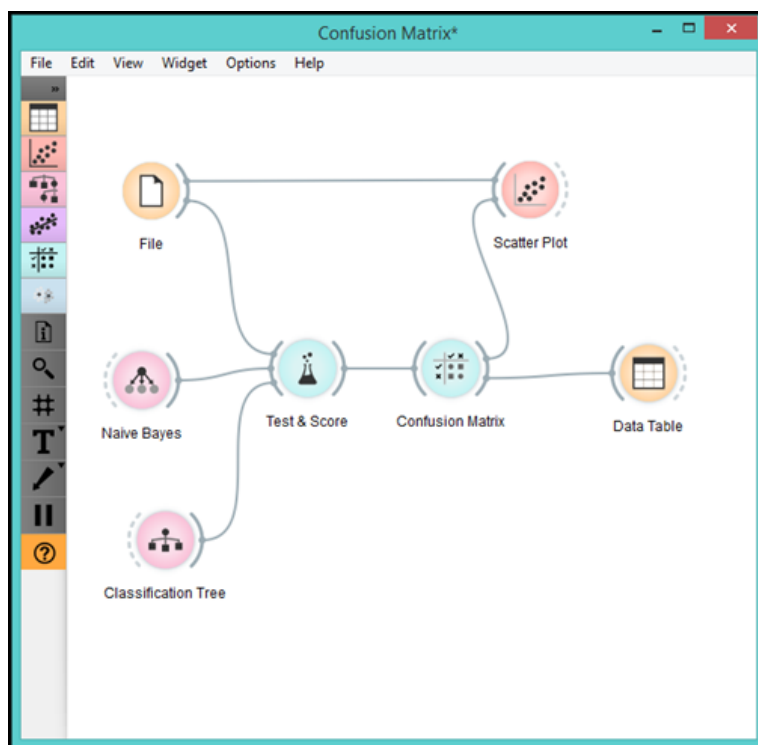
3.发送所选实例时，如果检查相应的选项预测和/或概率，组件可以添加新属性。

4.如果勾选“自动发送”，该小部件会输出每个更改。如果没有，用户将需要单击“发送选定”以提交更改。

5.生成报告。

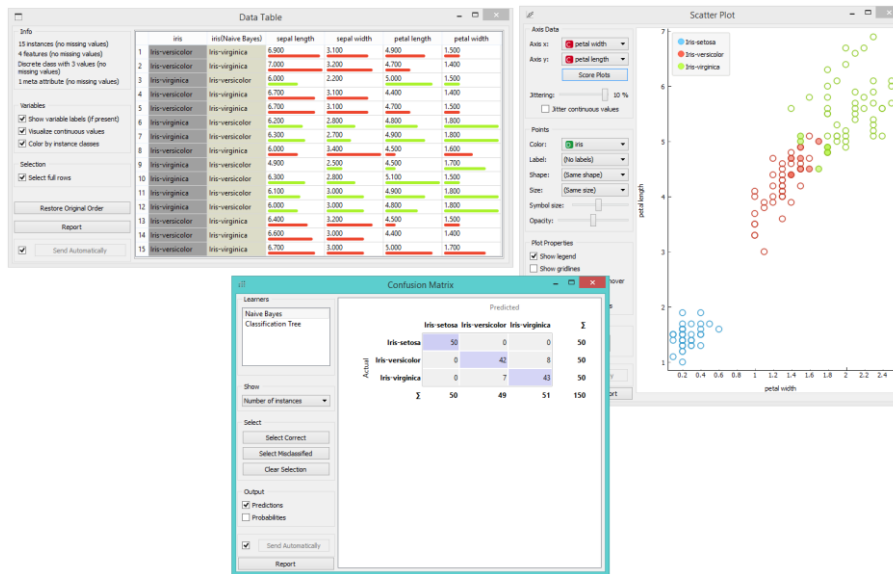
5.3.2 示例

以下工作流程演示了这个组件可以使用的内容。



5.3-2 示例图片

测试和分数从文件获取数据，并从 Naive Bayes 和 Tree 获取两种学习算法。它执行交叉验证或一些其他训练和测试程序，以获得所有（或一些）数据实例的两种算法的类预测。测试结果被输入到混淆矩阵中，在这里我们可以看到有多少个实例被错误分类，以及哪个方式。在输出中，我们使用数据表来显示我们在混淆矩阵中选择的实例。例如，如果我们点击 Misclassified，表将包含所选方法被错误分类的所有实例。Scatterplot 获取两组数据。从文件小部件得到完整的数据，而混淆矩阵只发送所选数据，例如错误分类。散点图将显示所有数据，粗体符号表示所选数据。



5.3-3 示例图片

5.4 ROC 分析

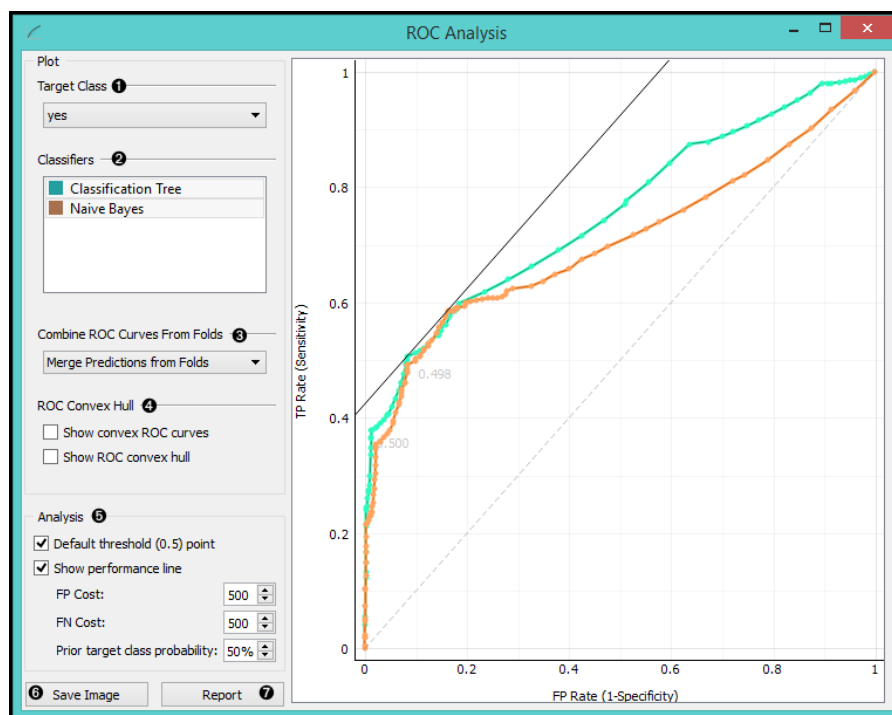


显示 ROC 曲线并对它们进行分析。

5.4.1 描述

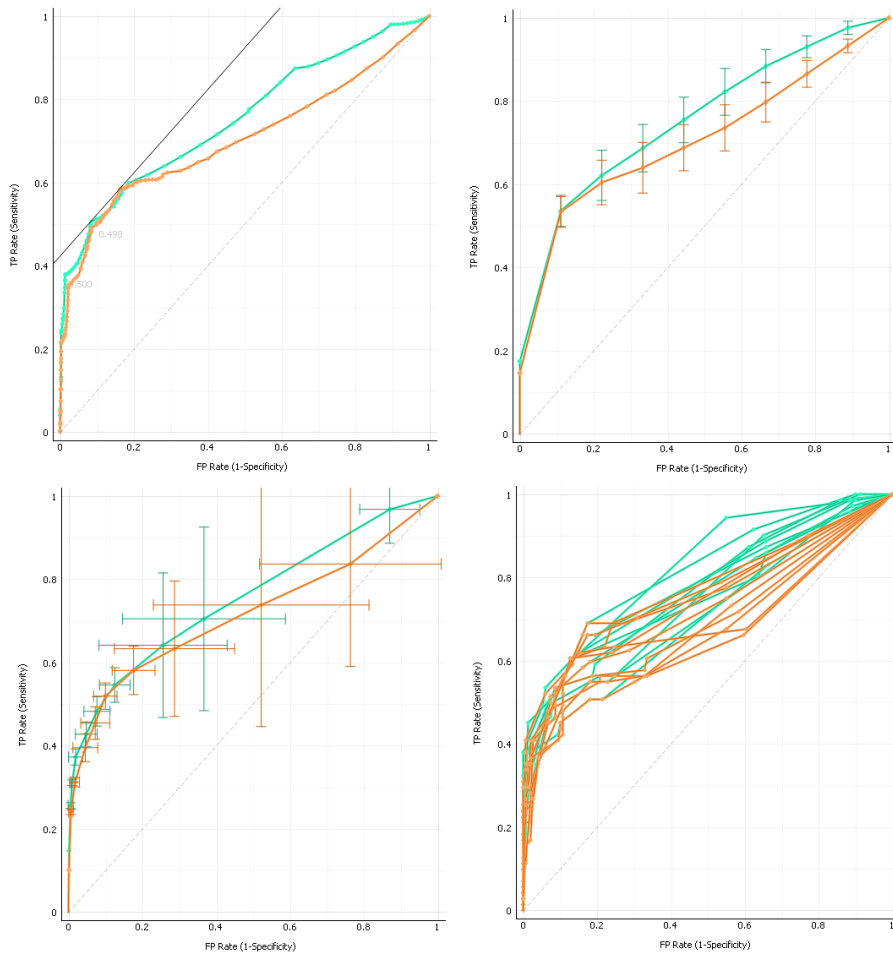
小部件显示测试的模型和相应的凸包的 ROC 曲线。它是分类模型之间的一种比较。曲线在 x 轴上绘制假阳性率(1 特异性;当真值 = 0 时的目标 = 1 的概率)与 y 轴上的真正阳性率(灵

敏度;当真值= 1 时的目标= 1 的概率) 1)。曲线越靠近左边框,然后是 ROC 空间的上边界,分类器越准确。鉴于假阳性和假阴性的成本,窗口小部件还可以确定最佳分类器和阈值。



5.4-1 ROC Analysis 窗口

- 1.选择所需的目标类。默认类按字母顺序排列。
2. 如果测试结果包含多个分类器,用户可以选择要查看的曲线。点击分类器选择或取消选择。
- 3.当数据来自多次迭代的训练和测试时,例如 k 倍交叉验证,结果可以(并且通常是)平均。

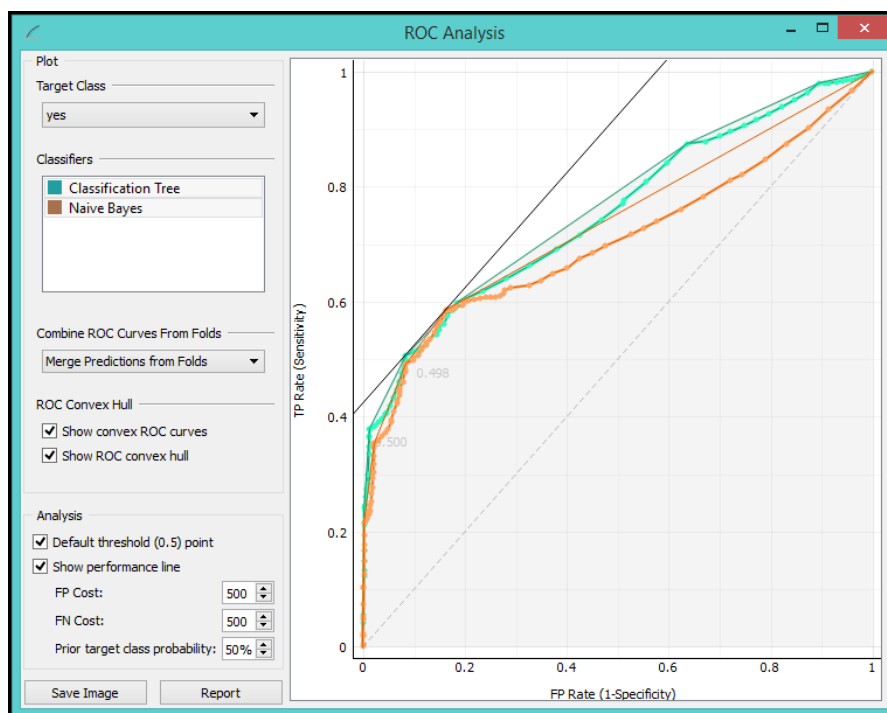


平均的选项是:

- 从折叠(左上角)合并预测，将所有的测试数据视为来自单个迭代
- 平均 TP 率(右上)是垂直的，显示出相应的置信区间
- 的平均 TP 和 FP 的阈值(左下方)遍历阈值，平均曲线的位置，显示水平和垂直置信区间
- 显示单个曲线(右下方)并不平均，而是描画所有的曲线

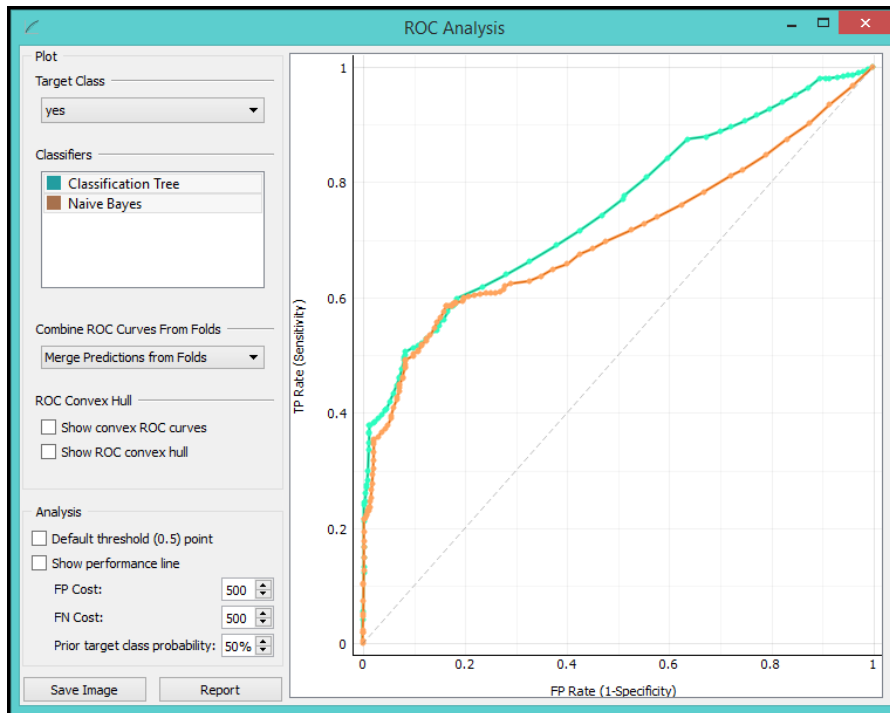
4.选项显示凸 ROC 曲线是指在每个单独的分类器上的凸曲线(曲线上的细线)。显示 ROC 凸包图绘制的一个组合的所有分类器(曲线下的灰色区域)。绘制这两种类型的凸曲线是有

道理的，因为选择一个阈值的凹部的曲线无视成本矩阵不能产生最佳的结果。此外，它是可能以达到任何点上的凸曲线相结合的分类器所表示的点的边界上的凹区域。



对角虚线表示随机分类器的行为。完整的对角线代表 iso 性能。图形底部的黑色“A”符号按比例重新调整图形。

5 最后一个专门用来分析曲线的。用户可以指定成本的误报 (FP) 和假阴性 (FN)，和前目标类概率。

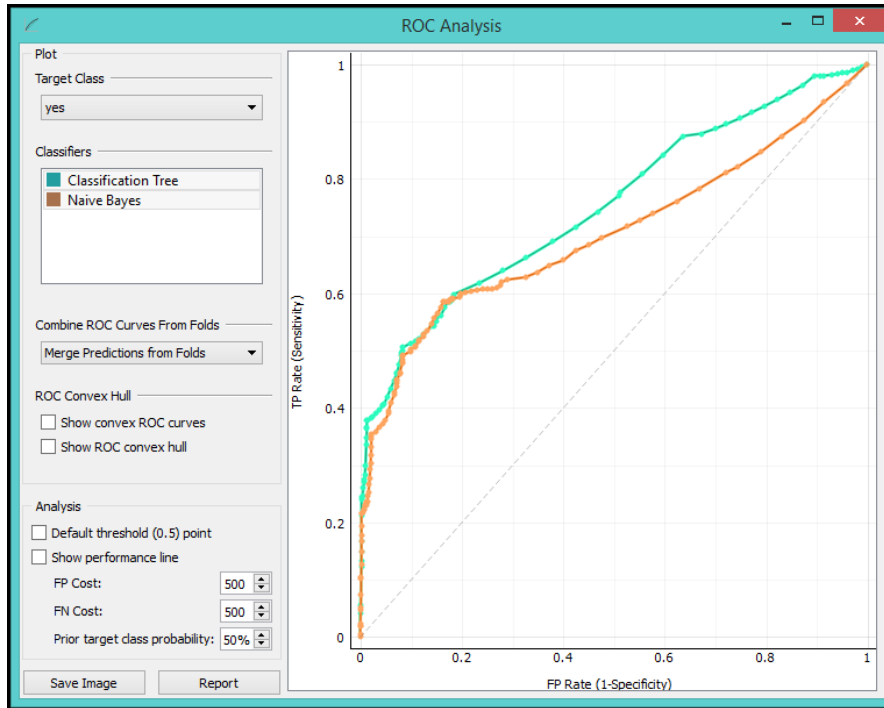


默认阈值 (0.5) 点：显示分类器实现的 ROC 曲线上的点，如果它的概率等于或等于 0.5，则预测目标类。

显示绩效线：显示在 ROC 空间中的异常表现，使得线上的所有点都具有相同的利润/损失。左上方的线优于右下方。线的方向取决于成本和概率。这给出了描述给定成本的最佳阈值的方法：这是具有给定倾斜度的切线接触曲线并且在图中标记的点。如果我们将等效性提高到更高或更高，则等级性能线上的点不能由学习者达到。向下或向右移动，降低性能。

该小部件允许将成本从 1 到 1000 设置。单位不重要，因为不是大小。这两个成本之间的关系是重要的，所以将它们设置为 100 和 200 将得到与 400 和 800 相同的结果。

默认值：成本相等 (500)，前目标类概率 50% (从数据) 假阳性成本：830，假阴性成本 650，前目标类概率为 73%。



6、如果要将创建的图像保存到 SVG 或 PNG 格式的计算机，请按“保存图像”。

7、制作报告。

示例

目前，唯一提供 ROC 分析所需的正确信号类型的部件就是“测试与分数”。下面我们比较两个分类器，即 Tree 和 Naive Bayes，并在“测试与分数”中进行比较，然后比较其在 ROC 分析，生命曲线和校准曲线中的表现。



5.4-2 示例图片

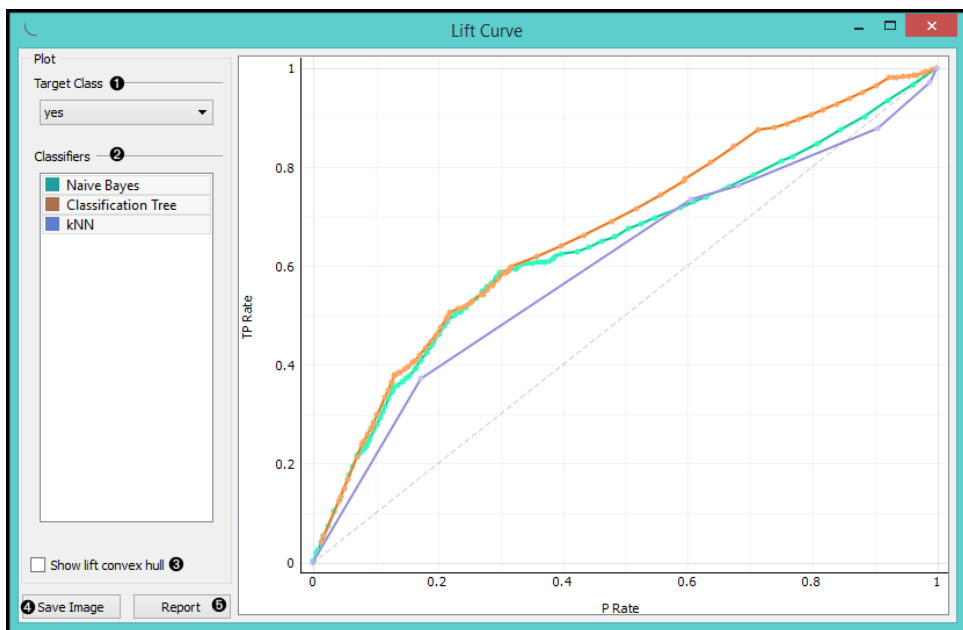
5.5 生命曲线



显示生命曲线并对其进行分析。

5.5.1 描述

生命曲线显示了预测为阳性的实例数与确实为正的实例数之间的关系,从而测量所选分类器对随机分类器的性能。该图以 x 轴上累积数(以概率降序排列)和 y 轴上真实阳性的累积数构成。升降曲线通常用于分割人口,例如,绘制响应客户的数量与所联系的所有客户的数量。您还可以从图中确定最优分类器及其阈值。



5.5-1 Lift Curve 窗口

- 1.选择所需的 Target 类。默认类按字母顺序排列。
- 2.如果测试结果包含多个分类器,用户可以选择要查看的曲线。点击分类器选择或取消选择曲线。
- 3.提升凸包在所有分类器的生命曲线上绘制凸包(黄色曲线)。该曲线显示了对于每个期望的 TP / P 速率的最佳分类器(或其组合)。

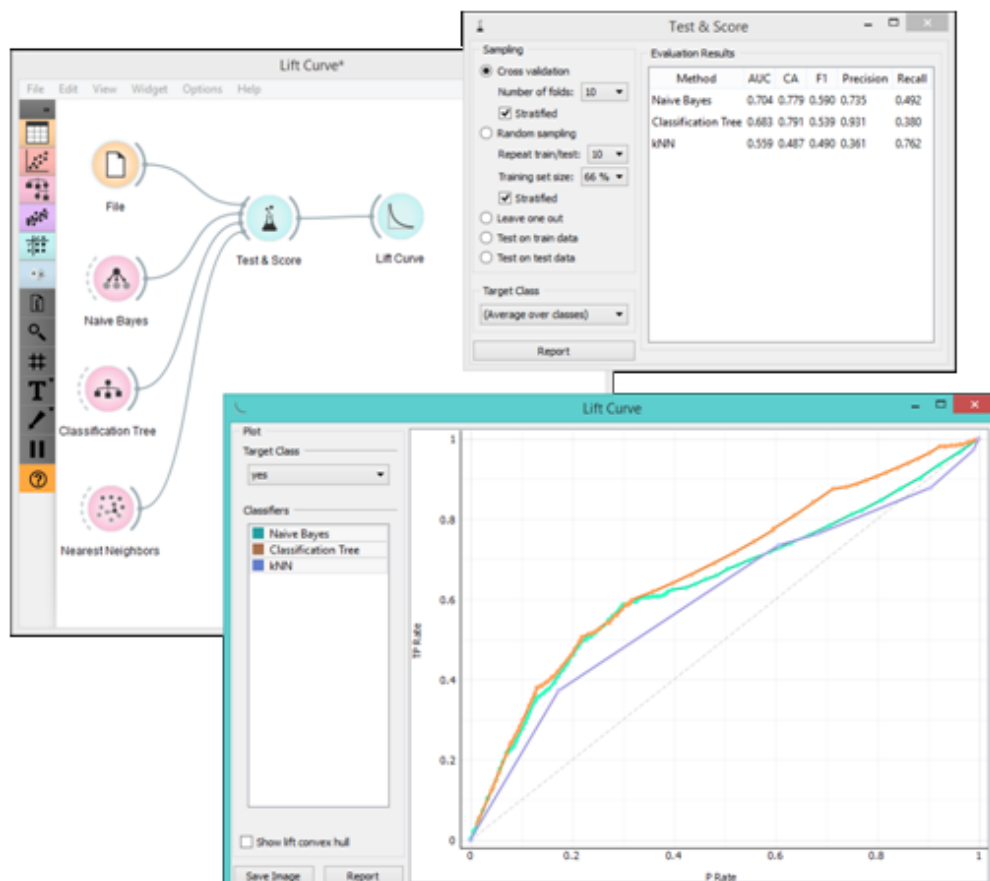
4.如果要以.svg 或.png 格式将创建的图像保存到计算机，请按保存图像。

5.制作报告。

5.5.2 示例

目前，唯一提供生命曲线所需信号类型的组件是“测试与分数”。

在下面的例子中，我们尝试看到 Titanic 数据集上的类“预存”的预测质量。我们在“测试学习者”小部件中比较了三种不同的分类器，并将它们发送到“提升曲线”，以根据随机模型查看其性能。我们看到树分类器是三个中最好的，因为它最好与升降凸包对齐。我们还看到，其表现对于前 30%的人口（按降序概率）来说是最好的，我们可以将其设置为最优分类的阈值。



5.5-2 示例图片

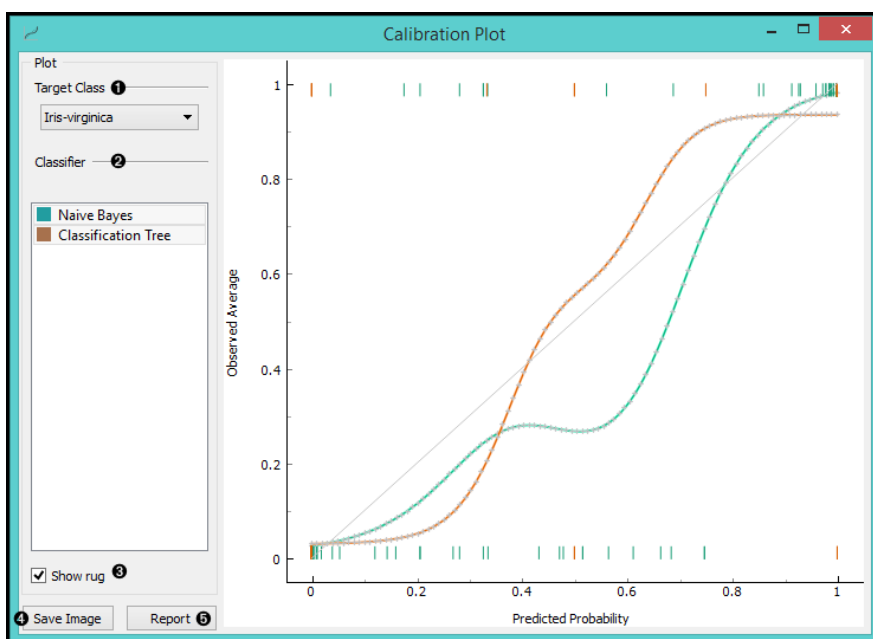
5.6 校准图



显示分类器的概率预测以及实际类概率之间的匹配。

5.6.1 描述

校准图 (Calibration plot) 绘制类概率与分类器预测的类概率进行比较的图。



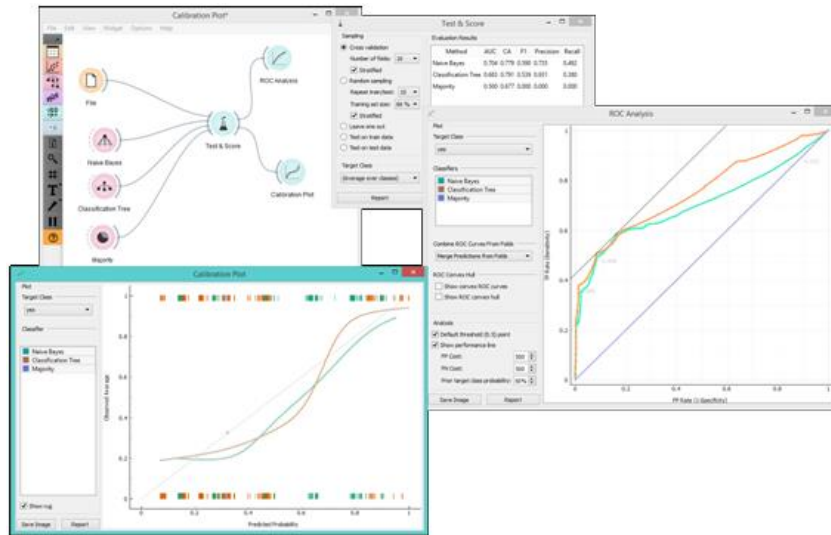
5.6-1 Calibration Plot 窗口

- 1.从下拉菜单中选择所需的目标类。
- 2.选择要绘制的分类器。对角线表示最佳行为;分类器曲线越接近,其预测概率越准确。因此,我们将使用这个小部件来查看分类器是否过于乐观(主要是积极的结果)或者消极(主要是负面的结果)。
- 3.如果显示“show rug”已勾选,刻度将显示在图的底部和顶部,分别表示负面和正面的示例。它们的位置对应于分类器的概率预测,颜色显示分类器。在图的底部,左侧的点是(正确地)分配了目标类的低概率的点,而右边的点被错误地分配给高概率。在图的顶部,右侧的实例被正确地分配给高概率,反之亦然。
- 4.如果要以.svg 或.png 格式将创建的图像保存到计算机,请按保存图像。
- 5.制作报告。

5.6.2 示例

目前,唯一提供校准图所需信号类型的组件是“测试与分数”。因此,校准曲线始终遵循测试和分数,由于没有输出,因此没有其他组件跟随。

这里是一个典型的例子,我们比较三个分类器(即 Naive Bayes, Tree 和 Constant),并将它们输入到 Test&Score 中。我们使用了 Titanic 数据集。Test&Score 然后显示每个分类器的评估结果。然后我们从测试和分数中绘制校准图和 ROC 分析窗口,以进一步分析分类器的性能。校准图使您能够看到图中类概率的预测精度。



5.6-2 示例图片

6 无监督

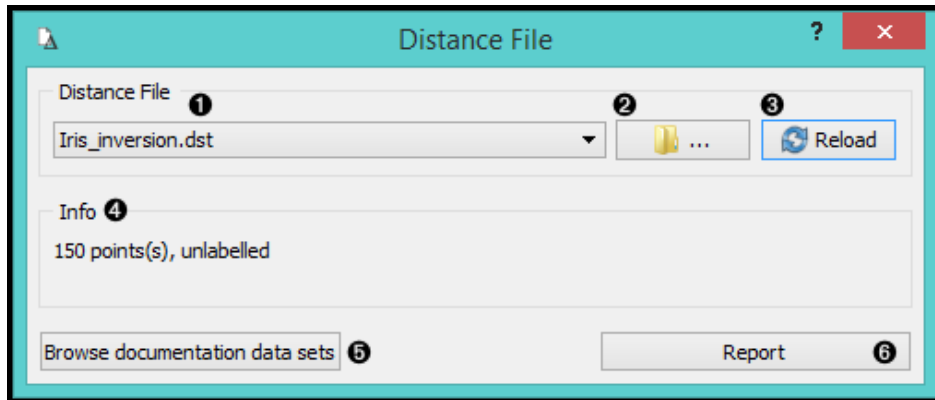
 距离文件	 距离矩阵	 距离图	 层次聚类
 K 均值聚类	 流行学习	 主成分分析	 一致性分析
 属性距离	 距离变换	 MDS	 保存距离矩阵

6.1 距离文件



计算数据集中示例之间的距离

6.1.1 描述

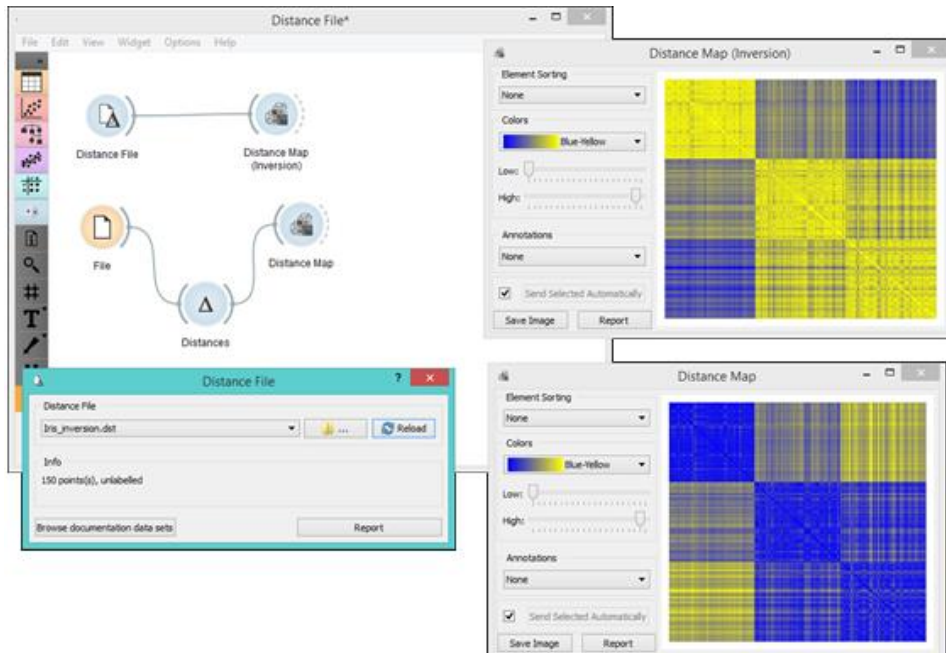


6.1-1 Distance File 窗口

- 1.从先前保存的距离文件列表中选择。
- 2.浏览保存的距离文件。
- 3.重新加载所选的距离文件。
- 4.关于距离文件的信息（点数，标记/未标记）
- 5.浏览文档数据集。
- 6.生成报告。

6.1.2 示例

当您想要使用之前保存的自定义距离文件时，打开“距离文件”组件，并使用“浏览表”选择所需的文件。此组件加载现有的距离文件。在下面的示例中，我们从保存距离矩阵示例中加载了转换后的 Iris 距离矩阵。我们在 Distance Map 组件中显示已转换的数据矩阵。我们还决定显示原始 Iris 数据集的距离图以进行比较。



6.1-2 示例图片

6.2 距离矩阵



可视化距离矩阵中的距离度量

6.2.1 描述：

“距离矩阵”组件创建一个距离矩阵，该距离矩阵是一个二维数组，它包含成对集合中元素之间的距离。数据集中的元素数量定义矩阵的大小。数据矩阵对于层次聚类是至关重要的，它们在生物信息学中也是非常有用的，它们用于以不依赖于坐标的方式表示蛋白质结构。

	Iris-setosa	Iris-setosa	Iris-setosa	Iris-setosa	Iris-setosa	Iris-versicolor	Iris-versicolor	Iris-versicolor	Iris-versicolor
Iris-versicolor	2.955	2.948	3.092	2.951	2.982	1.526	1.030	1.536	0.43
Iris-versicolor	2.152	2.406	2.285	2.435	2.291	2.632	2.112	2.657	0.91
Iris-versicolor	3.094	3.071	3.209	3.097	3.126	1.572	1.010	1.543	0.45
Iris-versicolor	3.076	2.960	3.176	2.990	3.069	1.421	0.843	1.425	0.76
Iris-versicolor	3.108	3.023	3.217	3.050	3.114	1.428	0.843	1.418	0.66
Iris-versicolor	3.373	3.243	3.503	3.240	3.350	0.949	0.458	0.964	0.97
Iris-versicolor	1.881	2.112	2.027	2.131	2.005	2.661	2.142	2.715	1.11
Iris-versicolor	3.023	2.970	3.142	2.990	3.040	1.490	0.922	1.487	0.54
Iris-virginica	5.324	5.132	5.418	5.167	5.305	1.844	1.808	1.616	2.66
Iris-virginica	4.164	4.104	4.274	4.135	4.193	1.449	1.063	1.253	1.34
Iris-virginica	5.365	5.171	5.491	5.167	5.325	1.407	1.688	1.187	2.70
Iris-virginica	4.706	4.562	4.815	4.584	4.696	1.245	1.183	0.990	1.95
Iris-virginica	5.085	4.923	5.197	4.942	5.070	1.463	1.493	1.212	2.35
Iris-virginica	6.174	5.958	6.300	5.950	6.124	2.121	2.500	1.936	3.50

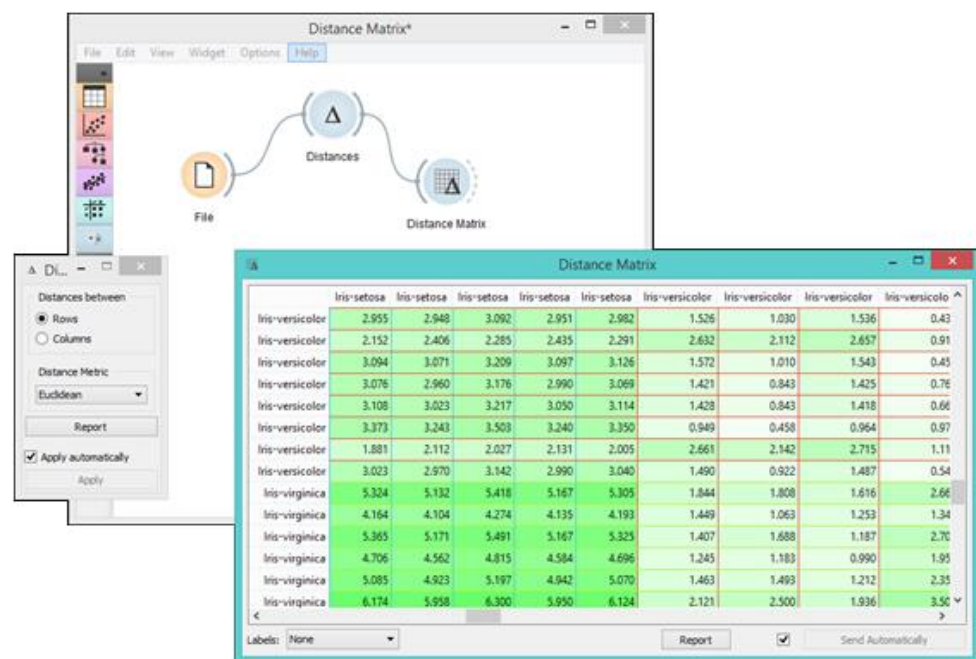
6.2-1 Distance Matrix 窗口

- 1.数据集中的元素以及它们之间的距离
- 2.标记表。 选项是：none，枚举，根据变量。
- 3.生成报告。
- 4.单击发送将更改通知给其他小部件。 或者，勾选“发送”按钮前面的框，更改将自动发送（自动发送）。

距离矩阵的唯一两个合适的输入是 Distanceswidget 和 Distance Transformation 组件。组件的输出是包含距离矩阵的数据表。用户可以决定如何标记表格，然后将距离矩阵（或距离矩阵中的实例）可视化或显示在单独的数据表中。

6.2.2 示例：

下面的示例显示了 Distance Matrix widget 的非常标准的用法。我们计算样本中数据集之间的距离，并将它们输出到距离矩阵中。Iris Setosa 和 Iris Setosa 是最遥远的，这一点也不奇怪。



6.2-2 示例图片

6.3 距离图



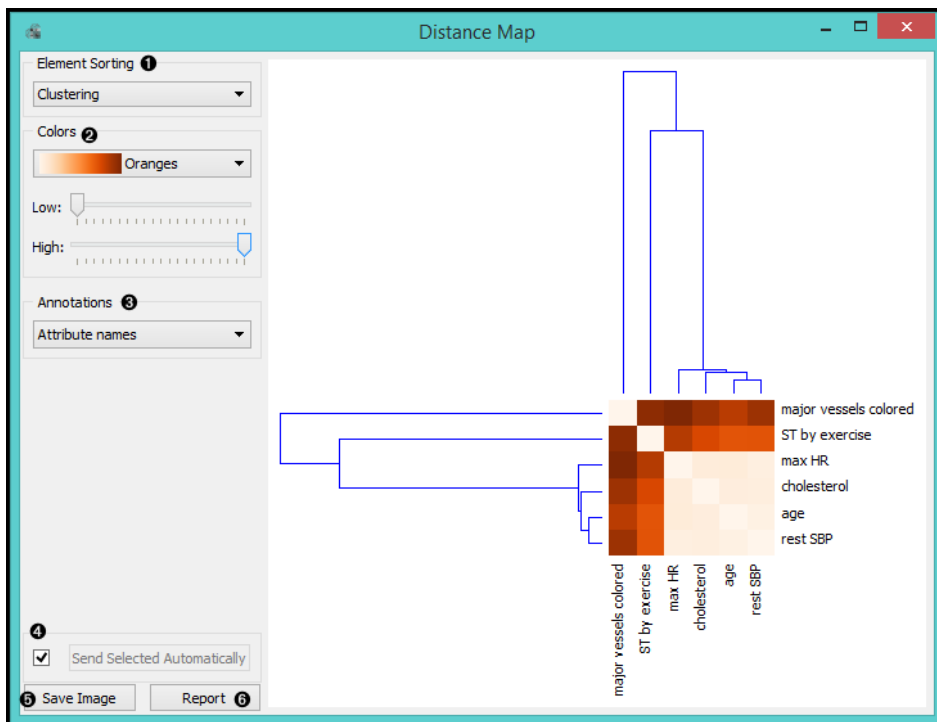
可视化项目之间的距离

6.3.1 描述

距离映射可视化对象之间的距离。这个可视化和我们打印一个数字表格是一样的，只不过数字被颜色的点代替了。

距离通常是在实例之间(距离小部件中的“行”)或属性(距离小部件中的“列”)之间的距离。远程地图的唯一合适的输入是距离小部件。对于输出，用户可以选择映射的一个区域，而小部件将输出相应的实例或属性。还要注意，Distanceswidget 忽略了离散的值，只计算连续数据的距离。

下面示例显示了心脏疾病数据中的列之间的距离，其中较小的距离用浅而较大的深橙色表示。矩阵是对称的，对角线是橙色的淡色 - 没有属性与本身不同。假定总是对称性，而对角线也可以是非零。



6.3-1 Distance Map 窗口

1.元素排序在地图中排列元素

None(列出在数据集中找到的实例)

o 集群(通过相似的集群数据)

o 用有序的叶子进行集群(最大化相邻元素的相似性的总和)

2.颜色

o 颜色(选择你的距离地图的调色板)

低的和高的颜色调色的阈值(对于实例或属性的低距离和高的实例或高距离的属性)。

3 所示.选择注释。

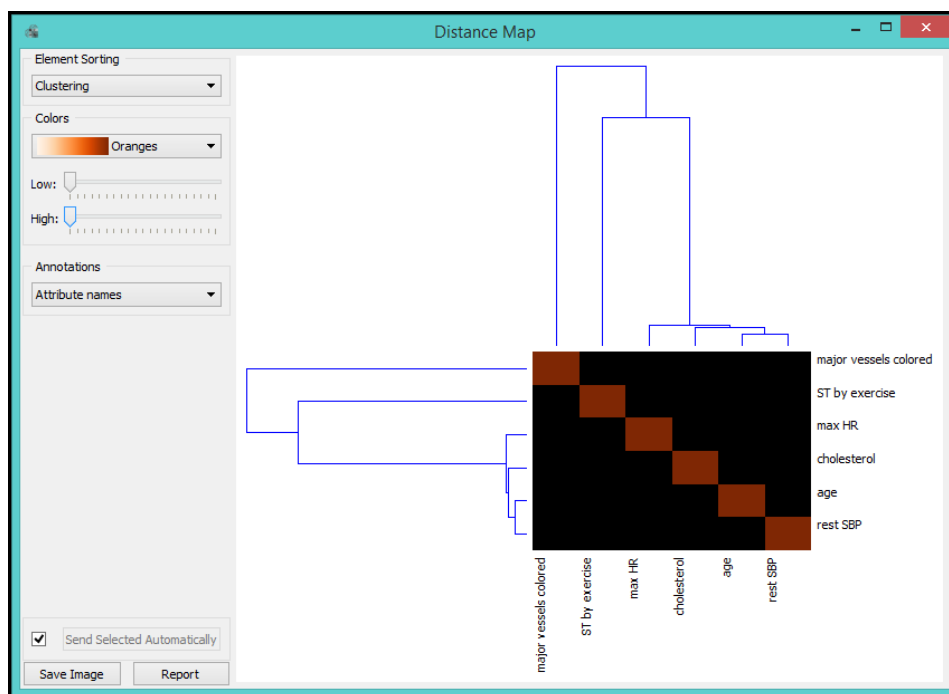
4 所示.如果自动选择发送，则数据子集自动通信，否则您需要按选择的发送。

5.如果您想将所创建的图像保存到您的计算机上，请按保存图像。

6.生成报告。

通常，调色板用于显示矩阵中出现的整个距离范围。这可以通过设置低和高阈值来更改。以这种方式，我们忽略了该间隔之外的距离差异，并且可视化分布有趣的部分。在下面，我们通过将高距离的颜色阈值设置为最小来显示心脏疾病数据集中最相关的属性(按列的距离)。我们得到一个主要是黑色的方块，其中具有最低距离分数的属性由所选颜色模式的较浅的阴

影表示（在我们的例子中是橙色）。在对角线旁边，我们看到，在我们的例子中，ST 运动和主要血管是两个最接近的属性。

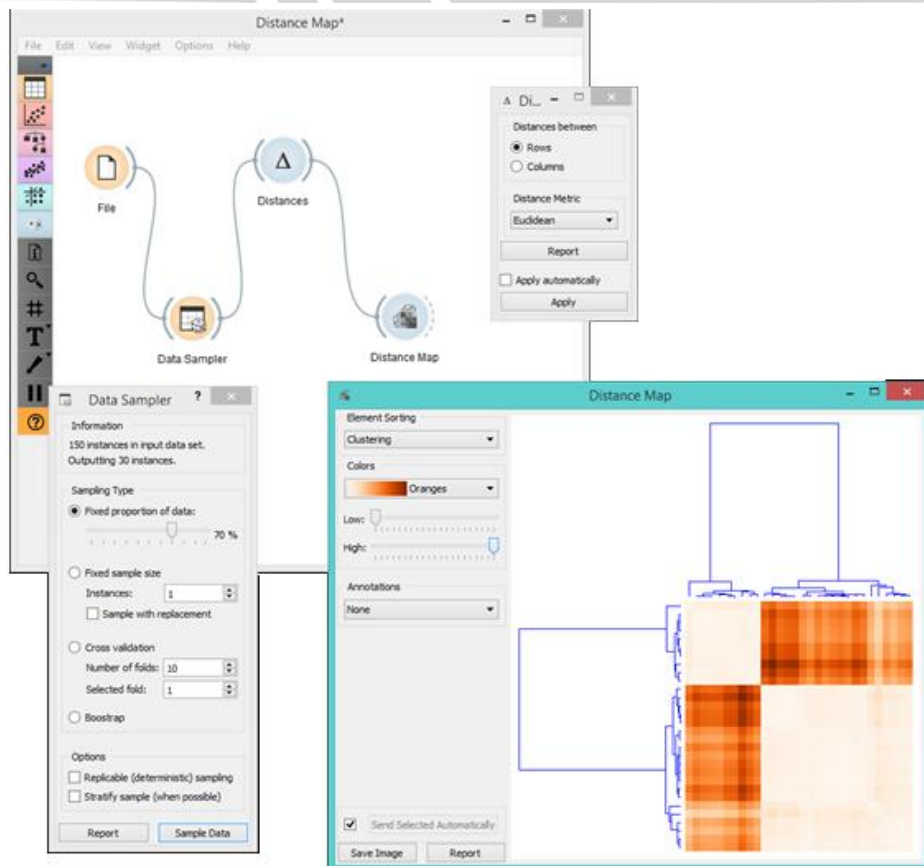


6.3-2 示例图片

用户可以通过光标点击或拖动来选择图中的区域。当选择图的一部分时，窗口组件将从所选单元格输出所有项目。

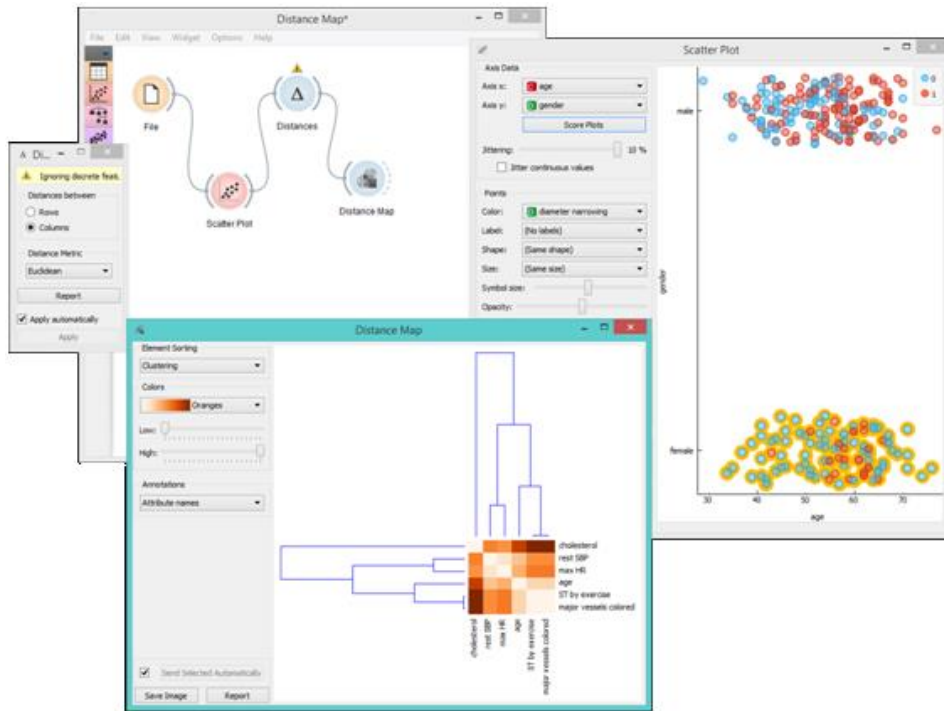
6.3.2 示例

第一个工作流程显示了 Distance Mapwidget 的标准的使用。我们选择 70% 的原始 Iris 数据作为我们的样本，并查看距离图中的行之间的距离。



6.3-3 示例图片

在第二个例子中，我们再次使用心脏病数据，并从散点图中选择女性子集。然后，我们可以看出距离图中的列之间的距离。由于子集还包含一些离散数据，“距离”组件会警告我们将忽略离散特征，因此我们将仅在地图中看到连续的实例/属性。



6.3-4 示例图片

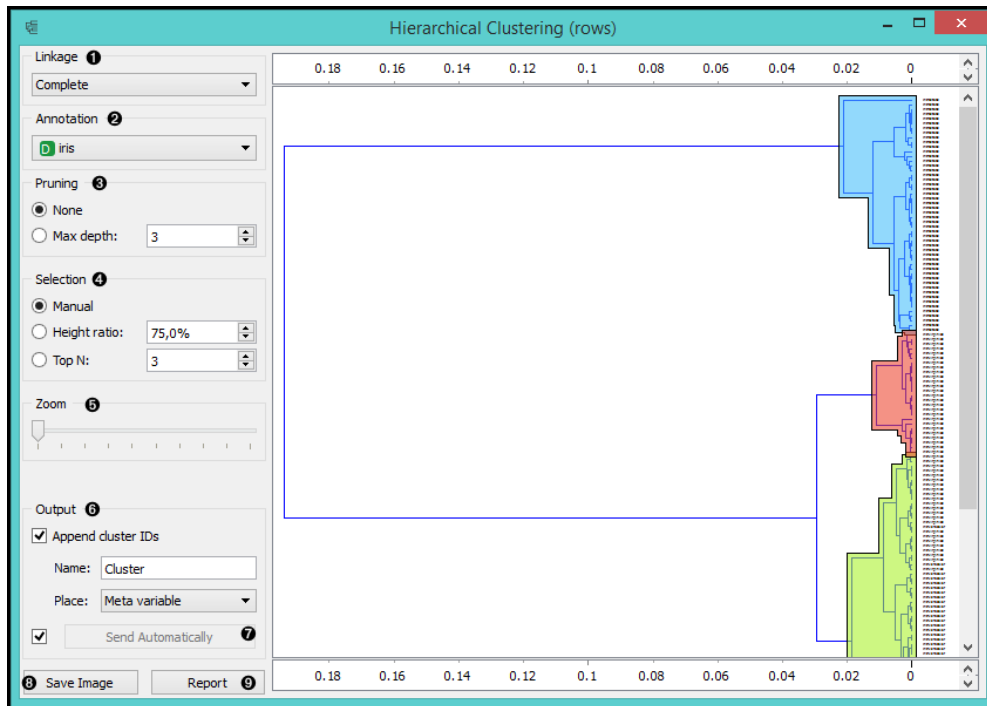
6.4 层次聚类



使用层次聚类算法对项目进行分组。

6.4.1 描述

层次聚类组件从距离矩阵计算任意类型的对象的层次聚类，并显示相应的树形图。



6.4-1 Hierarchical Clustering 窗口

- 1.小部件支持四种测量集群之间距离的方法：
 - 单个链接计算两个簇的最近元素之间的距离
 - 平均链接计算两个群集的元素之间的平均距离
 - 加权联动使用 WPGMA 方法
 - 完整的连接计算集群最远的元素之间的距离
- 2.可以在“注释”框中选择树形图中的节点标签。
- 3.通过选择树形图的最大深度，可以在修剪框中修剪巨大的树形图。这只会影响显示，而不影响实际的聚类。
- 4.小部件提供三种不同的选择方法：
 - 手动（单击树形图内部将选择一个群集，可以通过按住 Ctrl / Cmd 来选择多个群集，每

个选定的群集以不同的颜色显示，并在输出中被视为单独的群集。

o 高度比（点击树形图的底部或顶部标尺在图中放置一个截止线，选择行右侧的项）。

o Top N（选择顶级节点数）

5.使用缩放并滚动来放大或缩小。

如果要聚集的项目是实例，则可以添加一个集群索引（附加集群 ID）。该 ID 可以显示为一个普通属性，类属性或 Meta 属性。在第二种情况下，如果数据已经具有类属性，则原始类被放置在元属性之间。

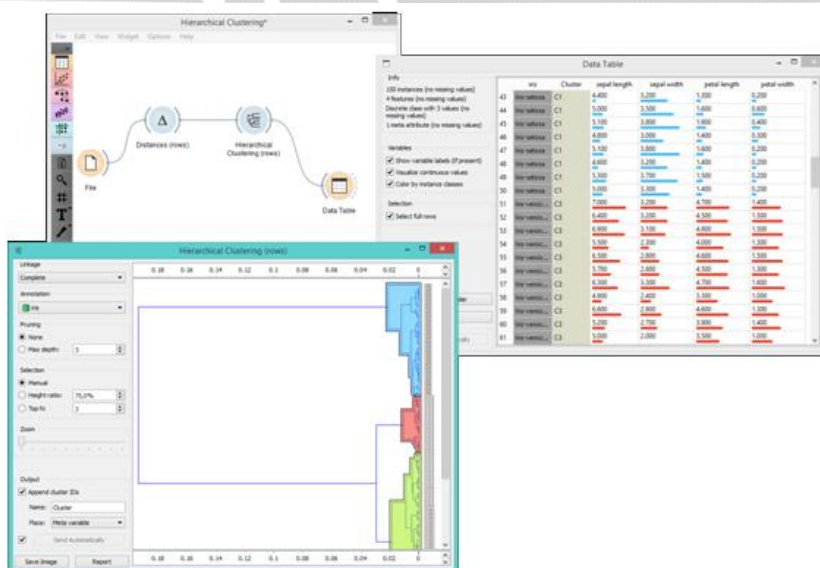
7.数据可以在任何更改（自动发送打开）上自动输出，或者如果未勾选该方框，请按“发送数据”。

8.单击此按钮可生成可保存的图像。

9.生成报告。

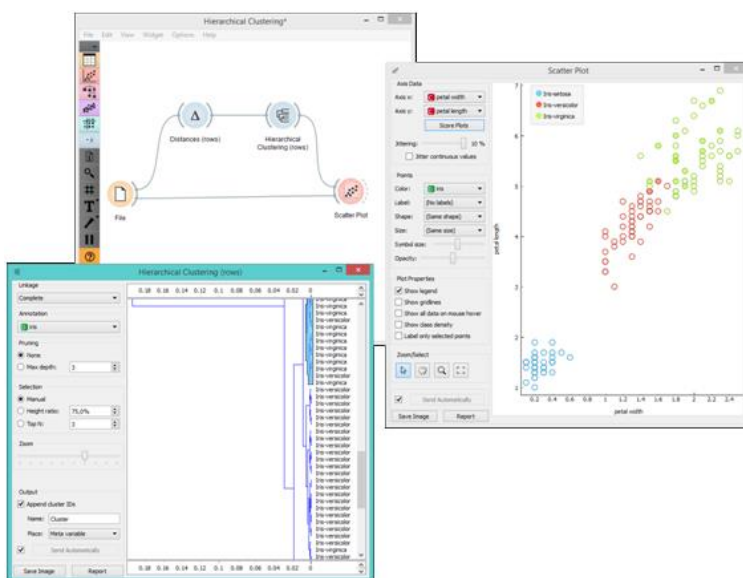
6.4.2 示例

下面的工作流显示了数据表控件中的 Iris 数据集的层次聚类输出。我们看到，如果在层次聚类中选择附加群集 IDs，我们可以在数据表中看到一个名为簇的附加列。这是一种检查层次聚类如何聚集单个实例的方法。



6.4-2 示例图片

在第二个例子中，我们再次加载了 Iris 数据集，但是这次我们添加了散点图，显示了文件中的所有实例，同时从层次聚类中接收所选的实例信号。这样我们可以观察到所选择的聚类在投影中的位置。



6.4-3 示例图片

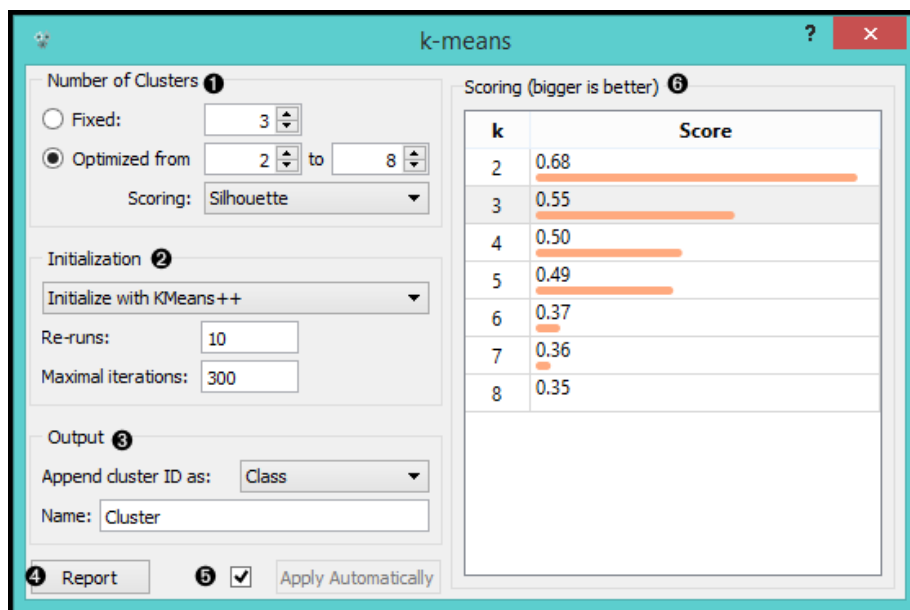
6.5 K 均值聚类



使用 K 均值聚类算法对示例进行分组。

6.5.1 描述

这个组件对输入中的数据应用 K 均值聚类算法并输出一个新的数据集。在这个数据集中使用簇索引作为类属性。原始的分类属性（如果存在）被移动到元属性。这个组件中还显示了有关聚类结果的基本信息。



6.5-1 K-means 窗口

1.选择群集数。

- 固定：算法将数据集中在指定数量的集群中。
- 优化：小部件显示所选群集范围的群集分数。
- 剪影（与同一个群集中的元素的平均距离与其他群集中的元素的平均距离形成对比）
- 集群间距离（测量集群之间的距离，通常在质心之间）
- 与质心的距离（测量到集群算术平均值的距离）

2.选择初始化方法（算法开始聚类的方法）：

- k-Means ++（第一个中心是随机选择的，随后是从与最近中心的平方距离成比例的剩余点中选择）
- 随机初始化（簇首先随机分配，然后进一步迭代更新）

重新运行（运行算法的次数）和最大迭代次数（每个算法运行中的最大迭代次数）可以手动设置。

3.小部件输出具有附加集群信息的新数据集。选择如何附加群集信息（作为类，功能或元属性）并命名列。

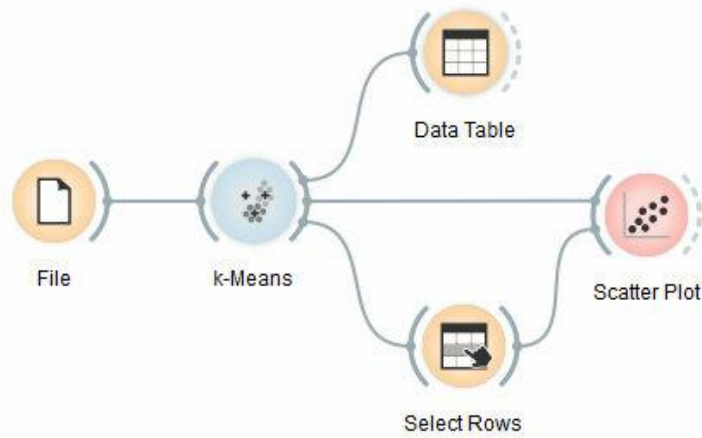
4.如果勾选“自动应用”，窗口小部件将自动提交更改。或者，单击应用。

5.制作报告。

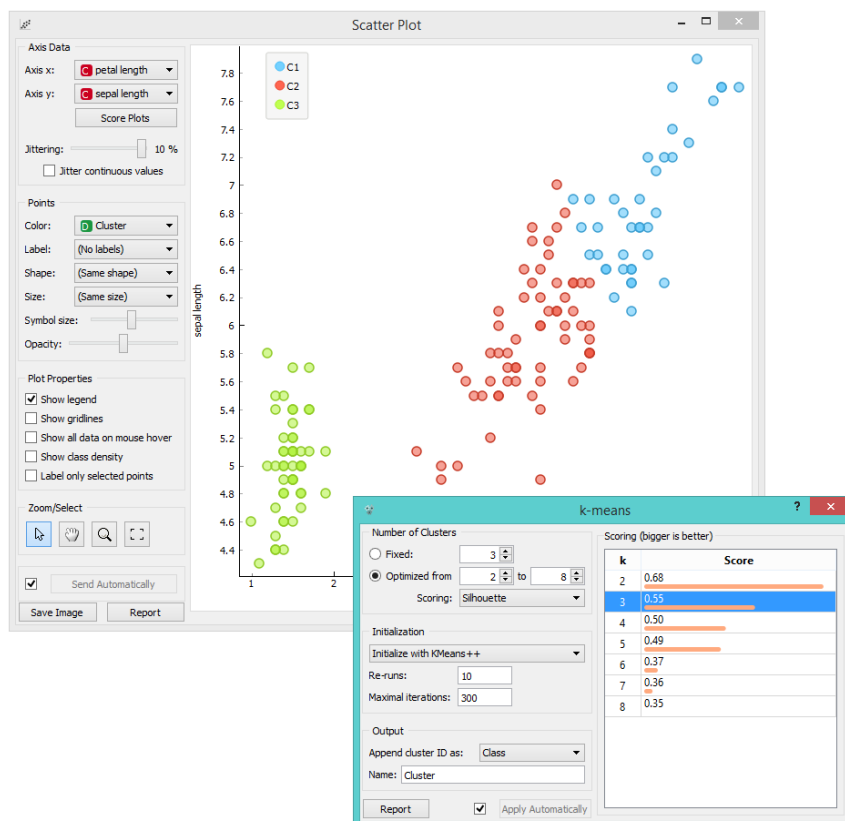
6.检查各种 k 的聚类结果的分数。

6.5.2 示例

我们将使用以下工作流来进行探索。



首先，我们加载 Iris 数据集，将其分为三个集群，并显示在数据表中，我们可以在其中查看哪个实例进入哪个集群。有趣的部分是散点图和选择行。

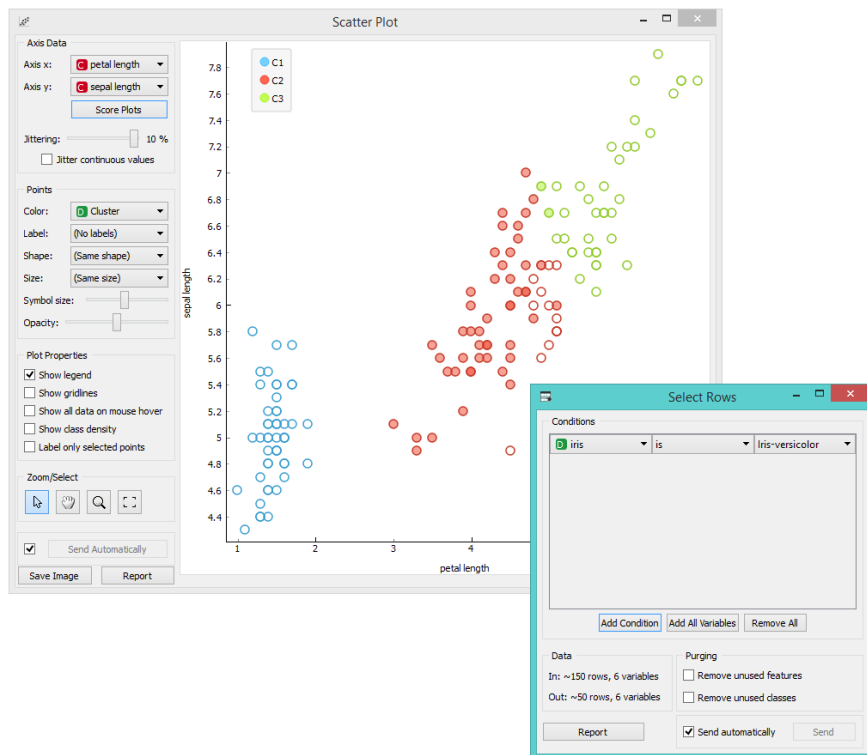


6.5-2 示例图片

由于 k-Means 将集群索引添加为类属性，所以散点图将根据它们所处的集群对点进行着色。

我们真正感兴趣的是(无监督)聚类算法引发的聚类与数据中的实际类匹配的程度如何。因此，我们选择“选择行”小部件，我们可以在其中选择单个类并在散点图中标记相应的点。

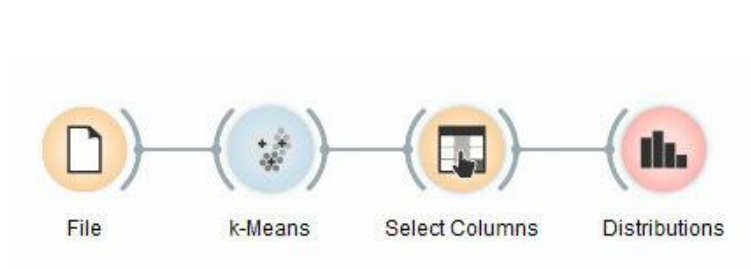
这场测试对 setosa 来说是完美的，对于其他两个来说也非常棒。



6.5-3 示例图片

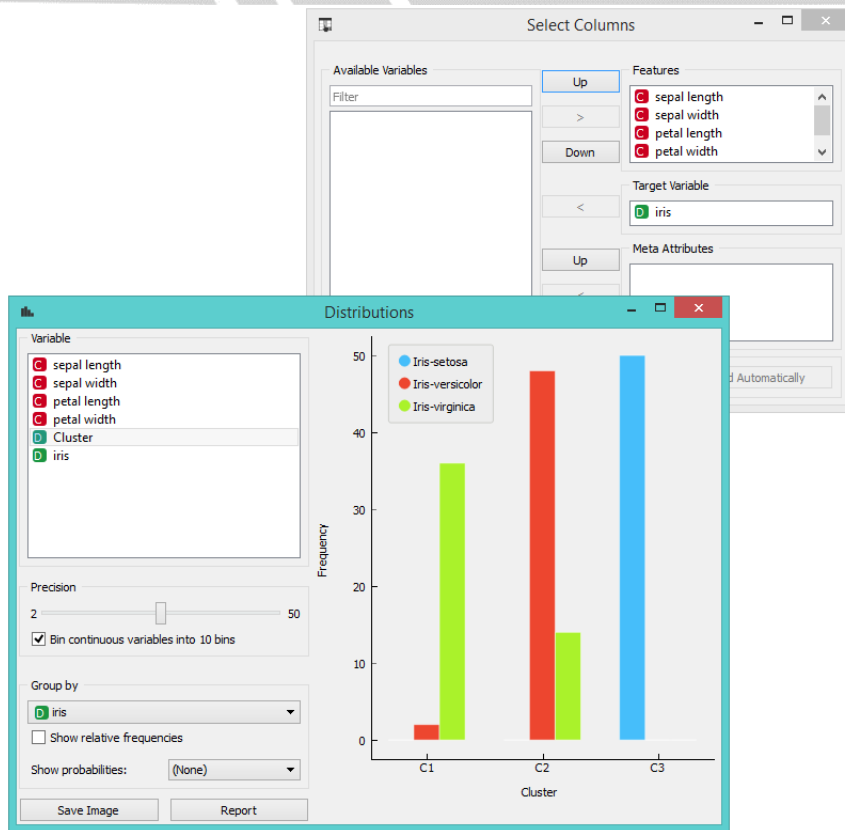
您可能已经注意到，我们保留了未使用的值/属性，并在选中的行中删除未使用的类。这很重要:如果小部件修改属性，它将输出修改过的实例的列表，而散点图不能将它们与原始数据进行比较。

测试集群和原始类之间的匹配的一种更简单的方法是使用分布组件。



这里唯一的（次要的）问题是这个小部件仅可视化正常（而不是元）属性。我们通过使用选择列来解决这个问题：我们将原来的类 Iris 恢复为类，并将集群索引放在属性之间。

这个测试对 setosa 是完美的 setosa 的所有实例都在第三个集群（蓝色）。48 个 versicolors 在第二个集群（红色），而两个最后在第一个。对于 virginicae，36 个在第一个集群，14 个在第二个集群。



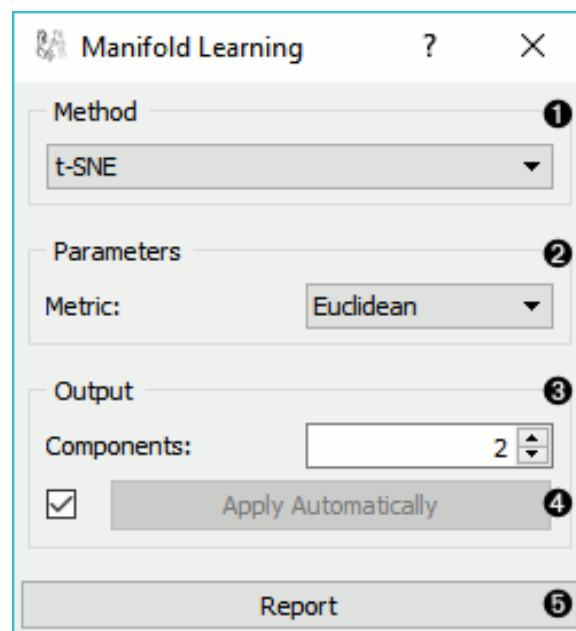
6.5-4 示例图片

6.6 流形学习



6.6.1 描述

流形学习是一种在高维空间中发现非线性流形的技术。然后，该部件输出对应于二维空间的新坐标。这些数据可以随后使用可视化与散点图或其他可视化部件。



6.6-1 Manifold Learning 窗口

1. 流式学习方法:

- t-SNE
- MDS, 参考 MDS 组件
- Isomap
- 局部线性嵌入
- 光谱嵌入

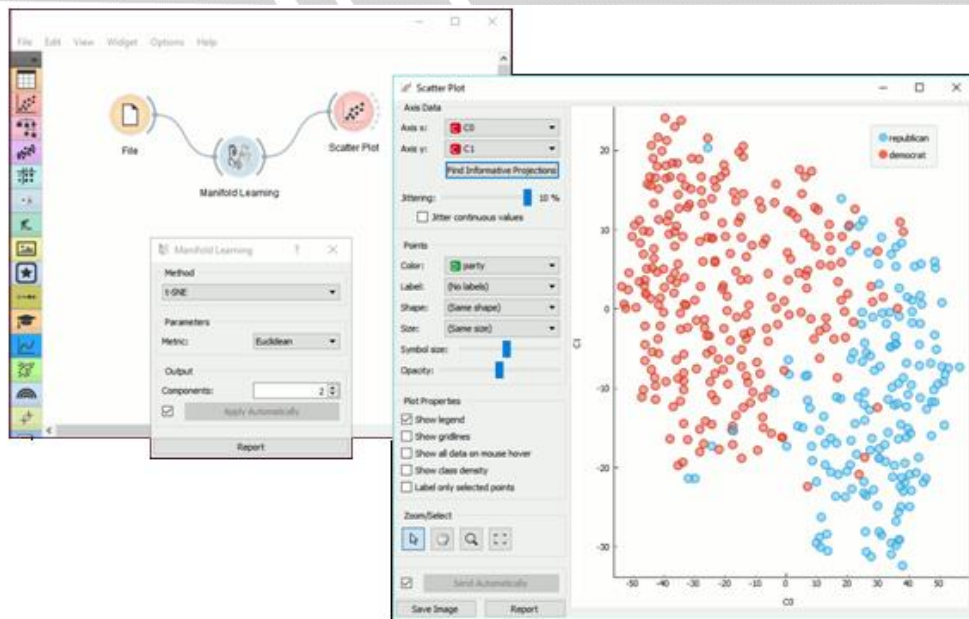
2. 设置方法的参数:

- t-SNE (距离测量):
 - 欧几里得距离
 - 曼哈顿
 - 切比雪夫
 - 杰卡德
 - 马氏
 - 余弦
- MDS (迭代和初始化):
 - 最大间隔:最大化优化间隔数
 - 初始化: 初始化算法的方法 (PCA 或随机)
- Isomap:
 - 邻居数量
- 本地线性嵌入:
 - 方法
 - 标准
 - 修改

- hessian 特征图
 - 本地
 - 邻居数量
 - 最大迭代数量
 - 光谱嵌入:
 - 亲和度:
 - 最近邻
 - RFB 内核
3. 输出: 减少功能 (组件) 的数量。
 4. 如果自动应用勾选, 更改将自动传播。或者, 单击应用。
 5. 生成报告。

6.6.2 示例 :

流形学习小部件将高维数据转换为较低维近似。这使得它可以很好地显示具有许多功能的数据集。我们使用 voting.tab 将 16 维数据映射到 2D 图上。然后我们使用散点图绘制嵌入。



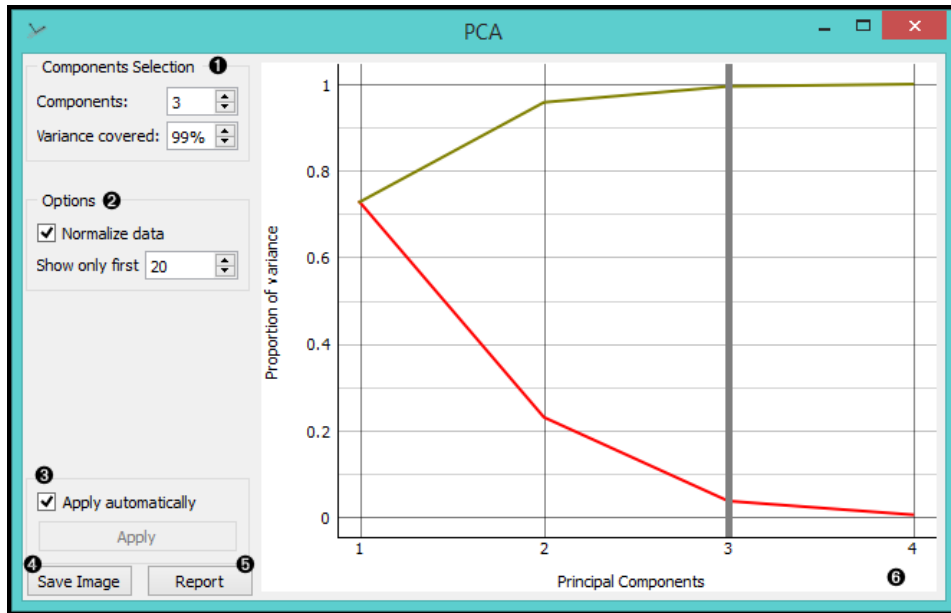
6.6-2 示例图片

6.7 主成分分析



6.7.1 描述

主成分分析 (PCA) 计算输入数据的 PCA 线性变换。 它输出具有单独实例的权重或主要组件权重的转换数据集。



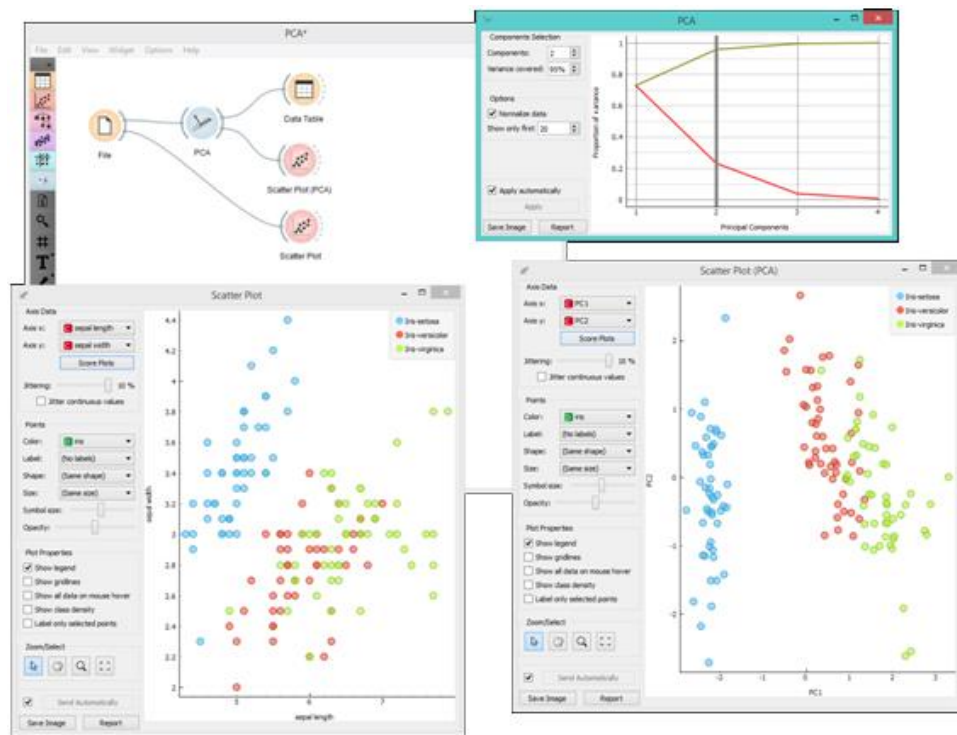
6.7-1 PCA 窗口

1. 在输出中选择您希望的主要组件数量。您还可以设置要用主要组件覆盖多少方差。
2. 您可以规范化数据以将值调整为常规。
3. 勾选自动应用程序时，窗口小部件将自动传送所有更改。 或者，单击应用。
4. 如果要将创建的图像保存到计算机，请按保存图像。
5. 制作报告。
6. 主要成分图，其中红色（下）线为每个成分所覆盖的方差，绿色（上）线为部件所涵盖的累积方差。

可以在“组件选择”输入框中或通过拖动图形中的垂直截止线来选择转换的组件数量。

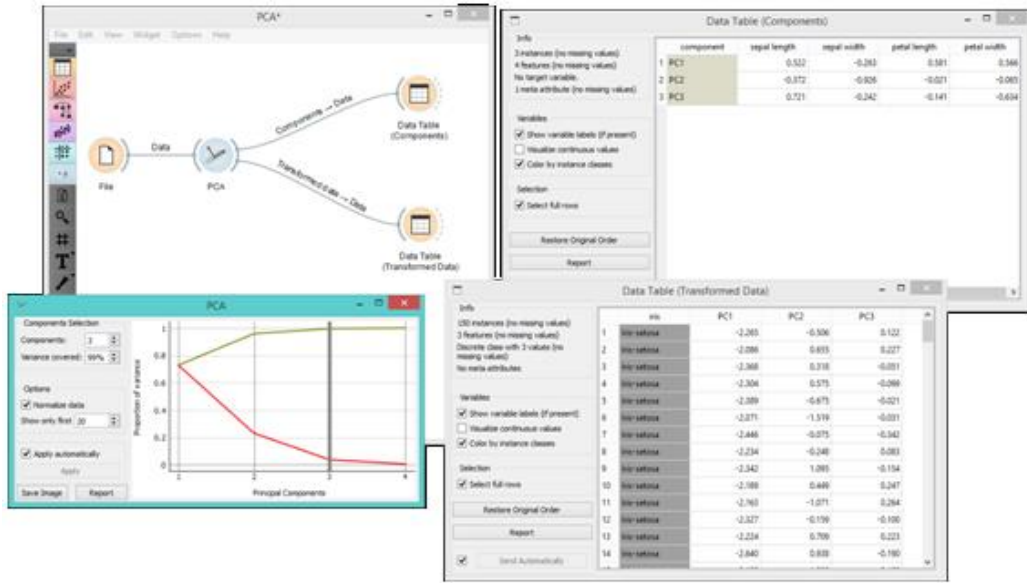
6.7.2 示例

PCA 可用于简化大数据集的可视化。下面我们使用 Iris 数据集显示如何使用 PCA 改进数据集的可视化。Scatter Plot 中的变换数据显示了类之间比默认设置更明确的区别。



6.7-2 示例图片

小部件提供两个输出：变换数据和主要组件。变换后的数据是新坐标系中各个实例的权重，而组件是系统描述符（公制组件的权重）。当输入数据表时，我们可以看到两个输出的数字形式。我们使用两个数据表，以便为工作流提供更清晰的可视化，但您也可以选择以仅在一个数据表中显示数据的方式编辑链接。您只需创建两个链接，并将“变换的数据和组件”输入连接到“数据”输出。



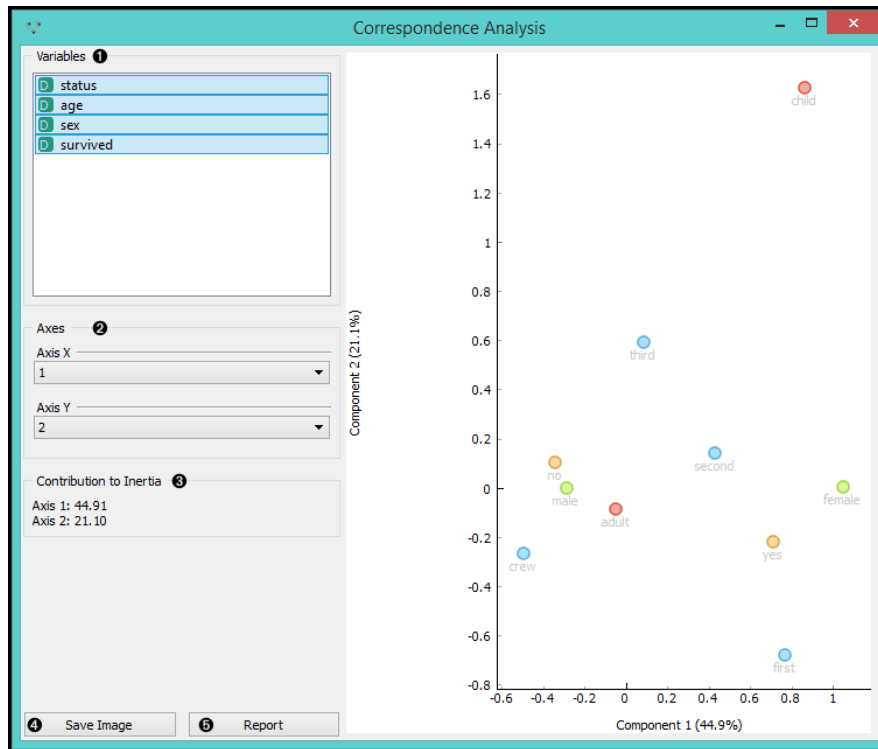
6.7-3 示例图片

6.8 一致性分析



6.8.1 描述

对应分析 (CA) 计算输入数据的 CA 线性变换。虽然它类似于 PCA，但 CA 计算离散的线性变换，而不是连续数据。

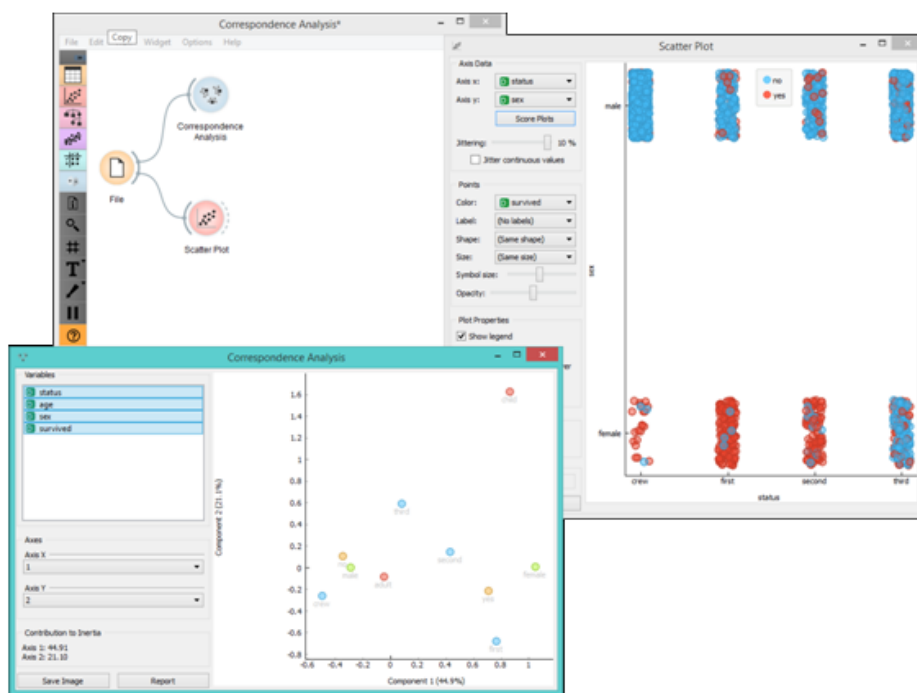


6.8-1 Correspondence Analysis 窗口

1. 选择查看的变量。
2. 选择每个轴的组件。
3. 惯性值（独立于转换的百分比，即变量在同一个维度）。
4. 生成报告。

6.8.2 示例：

下面是 Titanic 数据集上的对应分析和散点图小部件之间的简单比较。虽然散点图显示“类”和“性别”有良好的生存率。对应分析可以绘制一个二维图中的几个变量，从而很容易看到变量值之间的关系。从图中可以看出，“不”，“男”和“船员”都相互关联。“是”，“女”和“第一”也是一样。



6.8-2 示例图片

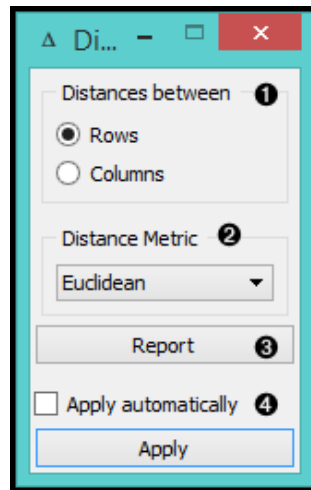
6.9 示例距离



计算数据集中示例之间的距离

6.9.1 描述

距离 (Distances) 组件计算数据集中行或列之间的距离。



6.9-1 Distances 窗口

1. 选择测量行或列之间的距离。
2. 选择距离度量：
 - 欧几里德 (“直线” , 两点之间的距离)
 - 曼哈顿 (所有属性的绝对差异之和)
 - 余弦 (内积空间的两个矢量之间的角度的余弦)
 - 杰卡德 (交集的大小除以样本集合的大小)

- 斯皮尔曼 (值之间的线性相关性, 以[0,1]间隔重新映射为距离)
- 斯皮尔曼绝对值 (绝对值的秩之间的线性相关性, 在[0,1]间隔中重新映射为距离)
- 皮尔逊 (值之间的线性相关性, 在[0,1]间隔中重新映射为距离)
- 皮尔逊绝对值 (绝对值之间的线性相关, 在[0,1]间隔中重新映射为距离)

在缺少值的情况下, 组件会自动计算行或列的平均值。

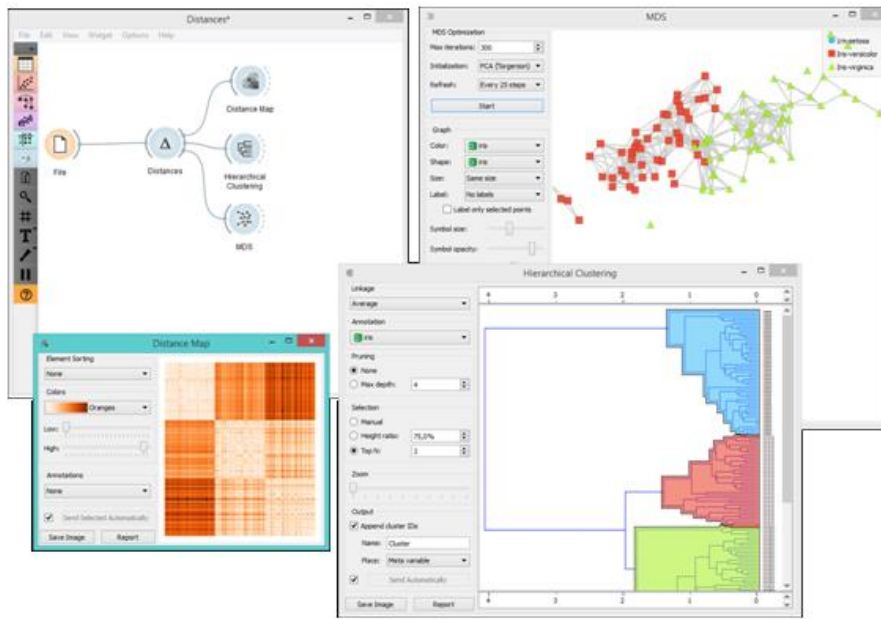
由于组件无法计算离散和连续属性之间的距离, 因此它仅使用连续属性并忽略离散属性。如果要使用离散属性, 请先使用“连续化”组件将它们连续化。

3. 制作报告。

4. 勾选 Apply Automatically , 自动将更改提交给其他小部件。或者, 按 “Apply” 。

6.9.2 示例

这个组件需要连接到另一个组件以显示结果, 例如连接到 Distance Map 来可视化距离, Hierarchical Clustering 以集中属性, 或 MDS 来可视化平面中的距离。



6.9-2 示例图片

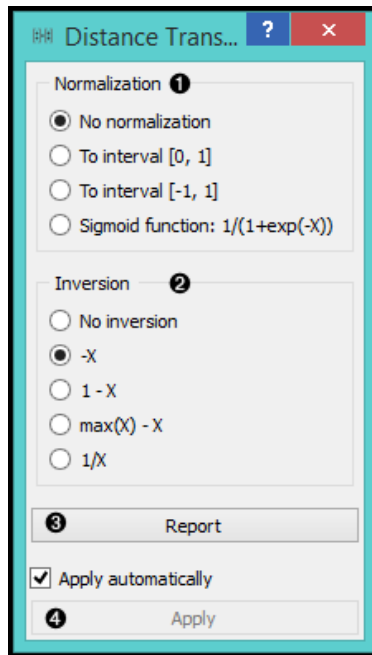
6.10 距离转换



转换数据集中的距离

6.10.1 描述

“距离转换”小部件用于距离矩阵的归一化和反演。数据的归一化是使用所有变量相互成比例的必要条件。



6.10-1 Distance Transformation 窗口

1. 选择标准化类型：

- 没有规范化
- 间隔[0, 1]
- 间隔 [-1, 1]
- S 形函数: $1/(1+\exp(-X))$

2. 选择反转类型:

- 无反转
- -X
- 1 - X

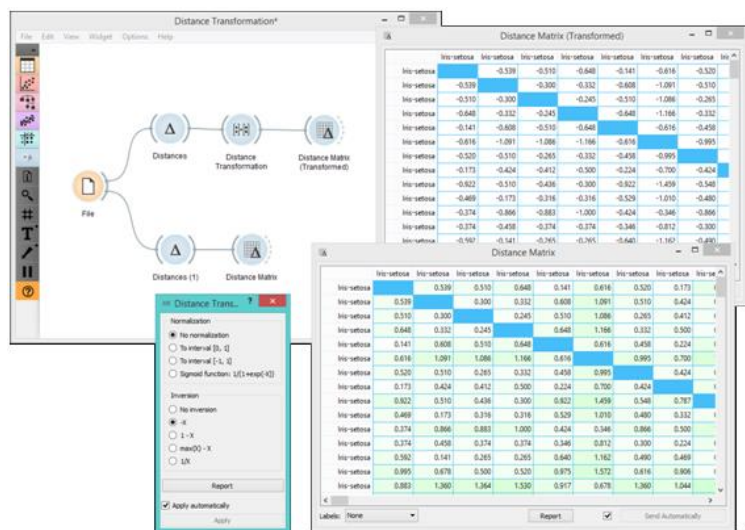
- $\max(X) - X$
- $1/X$

3. 生成报告。

4. 更改设置后，您需要单击应用以将更改提交到其他小部件。 或者，勾选自动应用。

6.10.2 示例：

在下面的示例中，您可以看到转换如何影响距离矩阵。 我们加载了 Iris 数据集，并在距离小组件的帮助下计算了行间的距离。 为了演示距离变换如何影响距离矩阵，我们创建了下面的工作流程，并将变换的距离矩阵与“原始”矩阵进行了比较。



6.10-2 示例图片

6.11 MDS



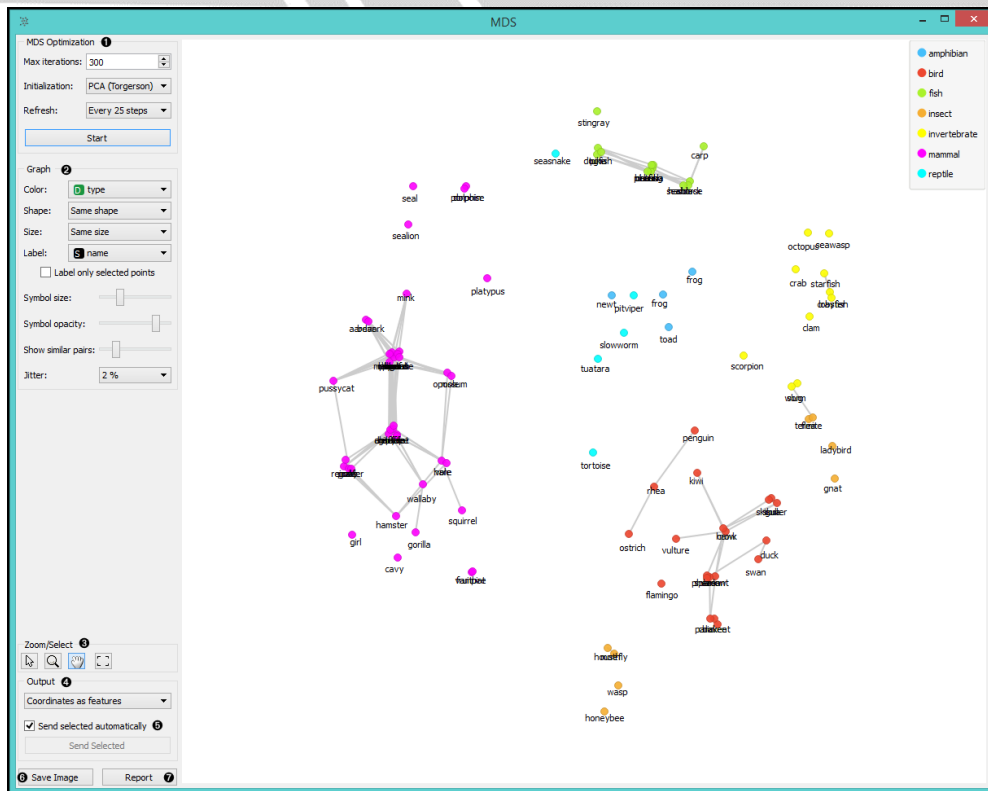
多维尺度 (Multidimensional scaling , MDS) - 预测到与点之间的给定距离拟合的平面
中

6.11.1 描述

多维尺度 (Multidimensional scaling) 是一种查找点的低维 (在我们的示例中 , 为二维) 预测的方法 , 该方法试图尽可能拟合给定的点之间的距离。完美拟合通常不可能实现 , 因为数据是高维数据或者距离不是欧几里得距离。

在输入中 , 窗口小部件需要数据集或距离矩阵。 当可视化行之间的距离时 , 您还可以调整点的颜色 , 更改其形状 , 标记它们 , 并在选择时输出它们。

这个算法在一种物理模型的模拟中迭代地左右移动这些点 : 如果两个点彼此距离太近 (或太远) , 会有一个力将它们分开 (合拢) 。 点在每个时间间隔的位置更改与作用于此上的力的总和相对应。



6.11-1 MDS 窗口

1. 在优化过程中，小部件重新绘制投影。优化在一开始就自动运行或之后通过按开始运行。

- 最大迭代：当投影在最后一次迭代时最小化或者达到最大迭代次数时，优化将停止
- 初始化：PCA (Torgerson) 沿着主坐标轴定位初始点。随机将初始点设置为随机位置，然后重新调整它们。

- 刷新：设置要刷新可视化的频率。它可以在每一个迭代，每 5/10/25/50 步骤或从不（无）。设置较低的刷新间隔使得动画更具视觉吸引力，但如果点数较高，则可能较慢。

2. 定义“点”如何可视化。这些选项仅在访问行间距离（在“数组”窗口小部件中选定）时可用。

- 颜色：按属性分的颜色（灰色连续，彩色为离散）。
- 形状：按属性点的形状（仅适用于离散）。
- 大小：设置点的大小（相同大小或选择属性），或者让大小取决于点表示的连续属性的值（Stress）。
- 标签：离散属性可以作为标签。
- 符号大小：调整点的大小。
- 符号不透明度：调整点的透明度级别。
- 显示类似的点：调整网线的强度。
- 抖动：设置抖动以防止点重叠。

3. 使用缩放/选择调整图形。该箭头可用于选择数据实例。放大镜可以进行缩放，也可以通过滚动进出。“手”图标允许您移动图形。矩形按比例重新调整图形。

4. 选择所需的输出：

- 仅原始特征（输入数据集）
- 仅坐标（MDS 坐标）

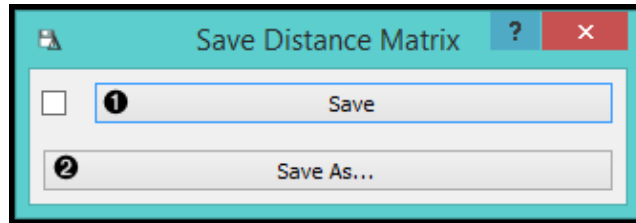
- 坐标作为特征（输入数据集+ MDS 坐标作为常规属性）
 - 坐标作为元属性（输入数据集+ MDS 坐标作为元属性）
5. 发送实例可以将“自动发送”选中勾选。或者，单击发送选择。
 6. 保存图像允许您将创建的图像以.svg 或.png 文件保存到设备中。
 7. 生成报告。

MDS 图表执行许多可视化部件的功能。这是在许多方面类似的散点图小部件，所以我们建议阅读该部件的描述。

6.11.2 示例

使用以下简单模式绘制上述图形。我们使用了 iris.tab 数据集。使用距离窗口小部件，我们将距离矩阵输入到 MDS 窗口小部件中，我们看到 Iris 数据显示在二维平面中。我们可以在“数据表”窗口小部件中看到附加的坐标。

6.12.1 描述



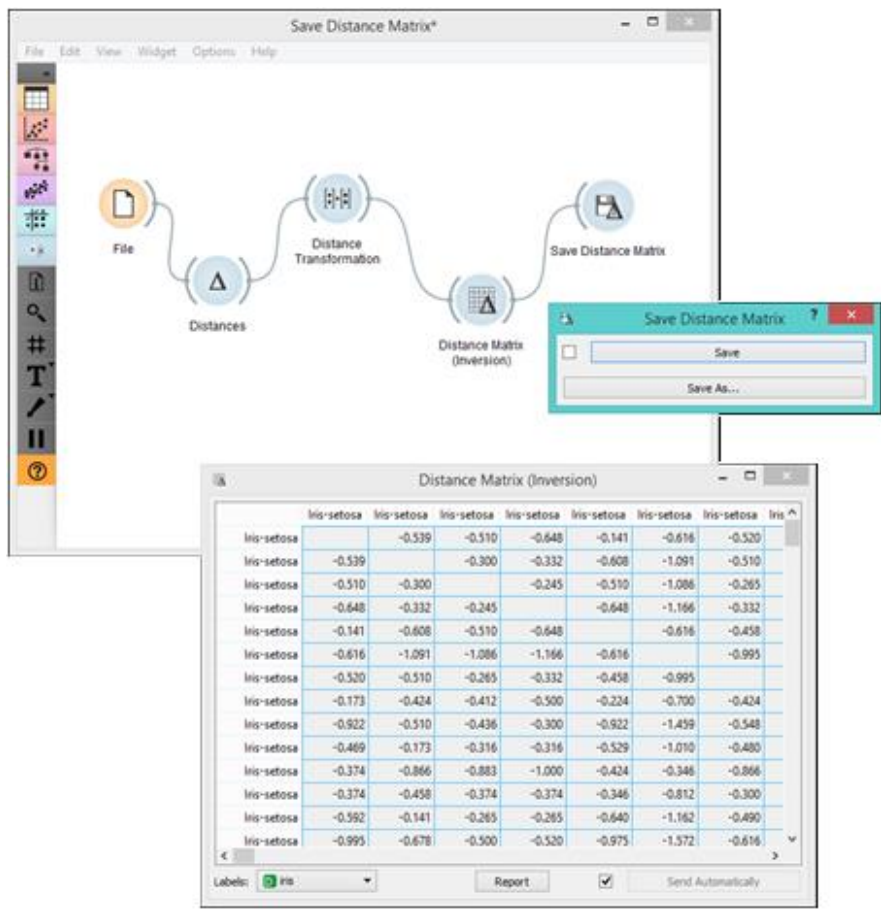
6.12-1 Save Distance Matrix 窗口

1、单击保存，您可以从先前保存的距离矩阵中选择。或者，单击保存按钮左侧的框，自动更改。

2、单击“保存为”，将“距离矩阵”保存到计算机中，只需输入文件名并单击“保存”。距离矩阵将被保存为.dst 类型。

6.12.2 示例：

在下面的图片中，我们使用“距离变换”小部件来转换 Iris 数据集中的距离。然后，我们选择将转换后的版本保存到计算机上以便稍后再使用。我们决定输出所有的数据实例。您可以选择仅输出数据矩阵的次要子集。如果您想知道我们更改的文件发生了什么，请到这里。



6.12-2 示例图片



曙光瑞翼教育合作中心

地址：北京市海淀区万柳亿城中心C2座1104室
电话：010-58815892